



DOAG Regionaltreffen Bremen, 02.02.16

Partitioning für OLTP –
Applikationstuning mal anders

Bremen, 02.02.2016, M. Giesel

- 60 Mitarbeiter an den Hauptstandorten Friedrichsdorf und München
- 11 Mio. Umsatz
- Branchen: Einzelhandel, Logistik, Öffentliche Hand, Pharma, Banken
- Flache Hierarchien, offene Kommunikation, Flexibilität
- Umfassende Beratung mit Praxis-Perspektive
Individuell optimierte Lösungen
- Starkes Partnernetzwerk

Hugendubel

Die Welt der Bücher



PARAGON DATA

Umsatz: 11 Mio €

Mitarbeiter: 60

Starkes Partner-Netzwerk



Fresh. Fair. Fullservice!

tolino  media 

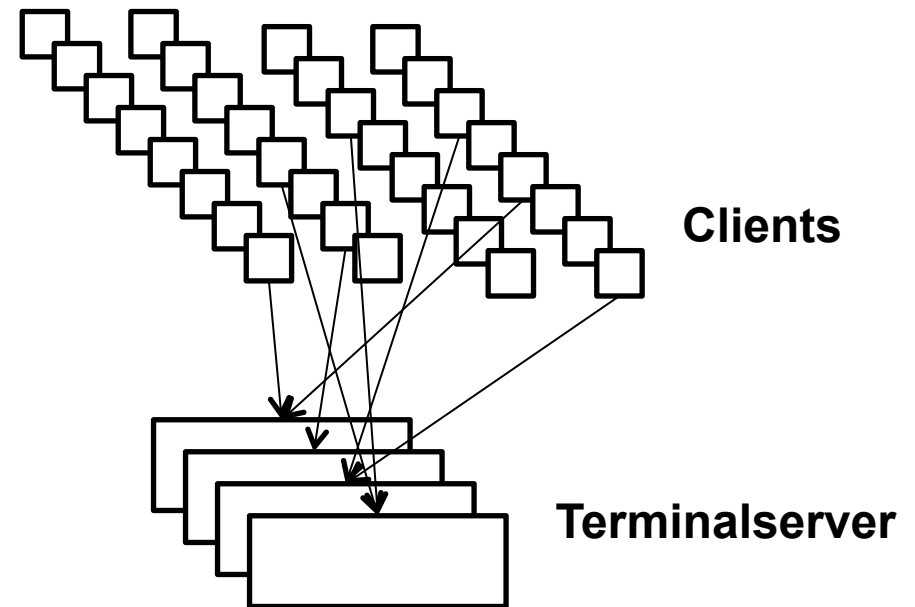


Weltbild *plus*

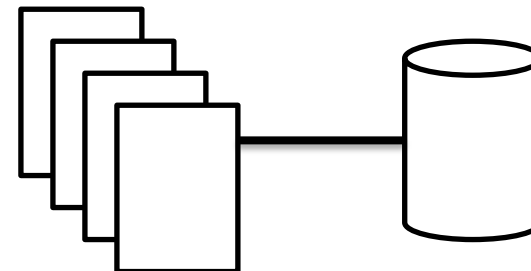


Die Ausgangssituation

- **Warenwirtschaftssystem**
- **buchhandelsspezifisch**
- **schlechte Performance**
- **blocking/ waiting Locks**
- **wenig/ keine Unterstützung des Herstellers**
- **Eigenentwicklungen**
- **Datenbankkonsolidierung, 8 Datenbanken auf einem System**
- **Partitioning Option durch Datawarehouse vorhanden**

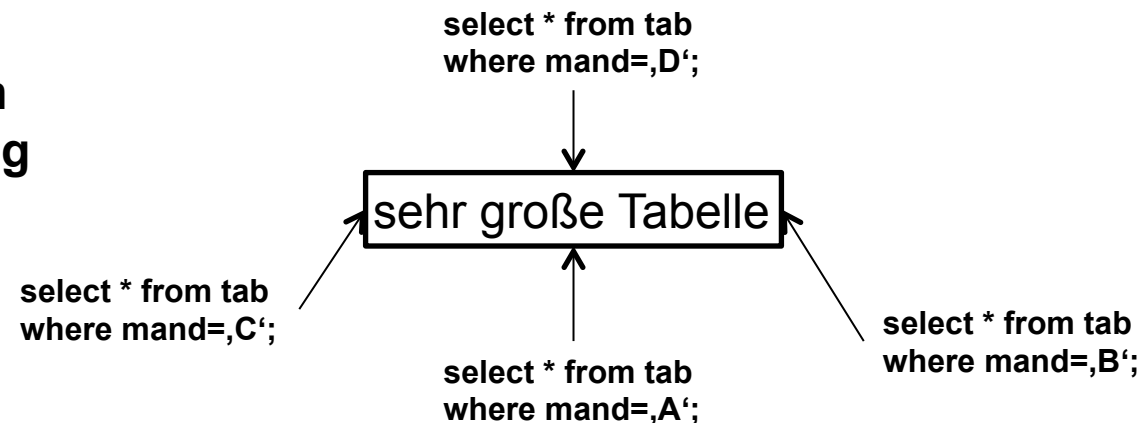


RAC/ Exadata mit Storage



- keine bzw. keine gute Zusammenarbeit mit dem Hersteller
- Applikation nicht optimal
- sehr kritische Performance, Gefährdung für das Unternehmen
- riesige Lockbäume mit mehr als 200 Sessions beteiligt
- „Sehr große“ Tabellen werden von allen in Teilen/ mandantenabhängig gelesen
- Sehr stark gelesene mandantenunabhängige Tabellen

Session 1: select * from ... for update;
→ Session 2: select * from ... for update;
→ Session 3: select * from ... for update;
...
→ Session n: select * from ... for update;



Fasten your transaction!

- ~~• Applicationtuning durch den Hersteller~~
- Hardwareupgrade
 - Memoryupgrade
 - CPU
 - Storage
- Applicationtuning durch applikationstransparente Änderung des Datenmodells
 - Partitioning
 - möglich, da DB Konsolidierung auf System mit Partitioning Option
 - Mit Partitioning besteht die Möglichkeit von effektiver Nutzung von Parallel Query

Partitioning, aber wie?

Verschiedene Möglichkeiten des Partitioning:

- 1) Range
- 2) Hash
- 3) Composite

Keep it simple!

Indexpartitioning → nur local oder global

Partitioningkey?

Key muss in den SQL-Statements benutzt werden, sonst kein Nutzen!

Mandantengestütztes System, Mandantenquadrupel (FirmaNr, MandantNr, GeschäftsNr, FilialNr) zieht sich durch das komplette DB-Modell!

Zugriffe per SQL-Trace und SGA (V\$SQL) ermitteln

**langwierige und
aufwendige Analysen**

Vorgehensweise

Große und häufig zugegriffene Tabellen und Indizes werden partitioniert!

→ 15 Tabellen identifiziert!

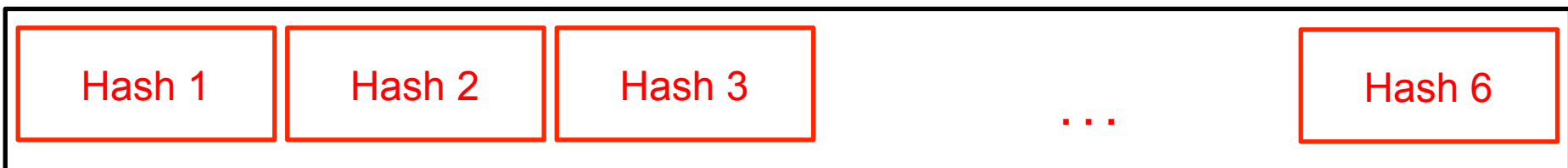
→ Partitionkey wird das Mandatenquadrupel → 1 Partition pro Filiale

Tabelle A



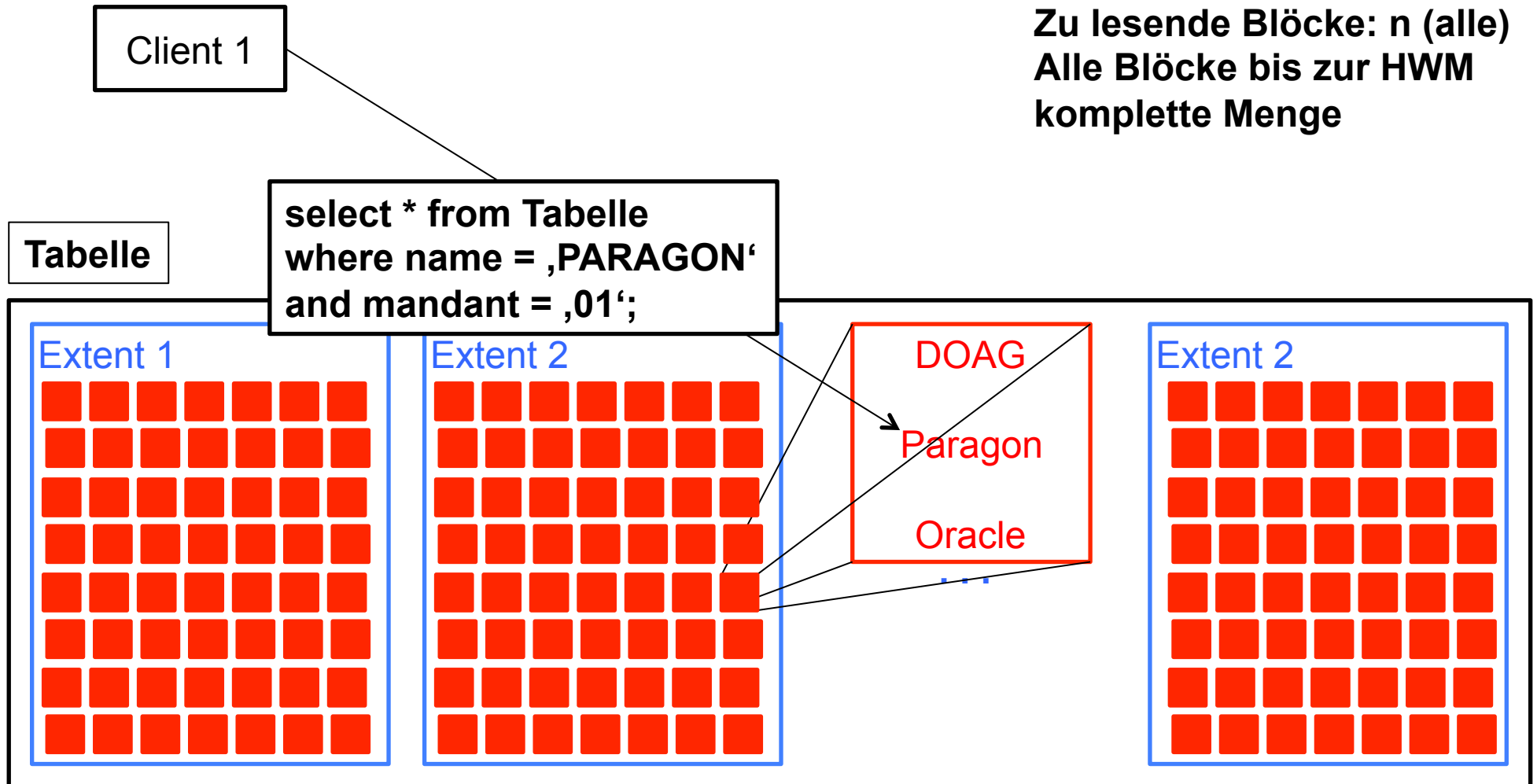
→ Tabellen ohne Partitionkey (Sortiment ist für alle Filialen gleich, da zentral bewirtschaftet) → Hashpartitioning auf Primary Key mit PQ

Tabelle S

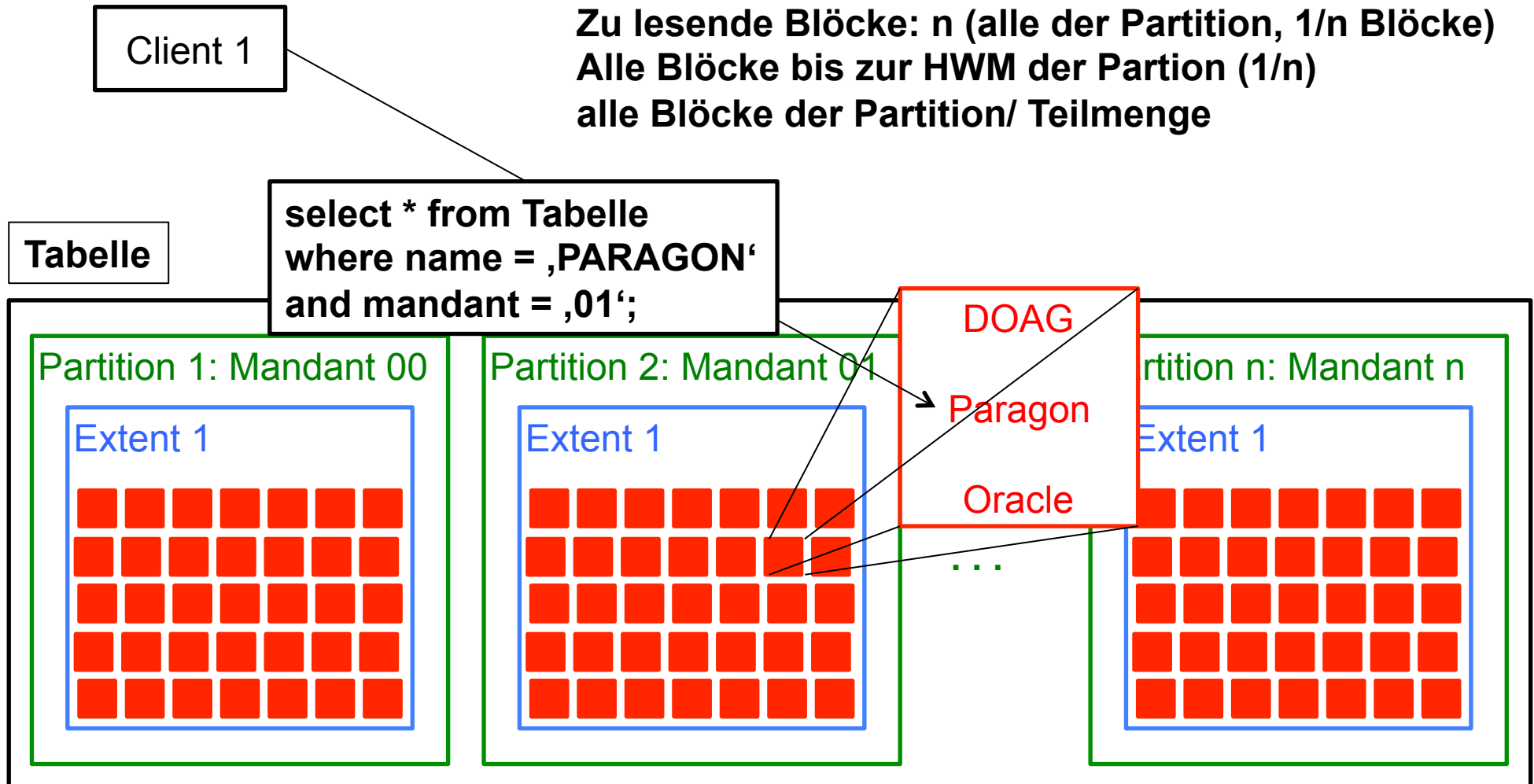


→ keine Reduzierung der Treffermenge, aber schnellerer Zugriff per PQ!

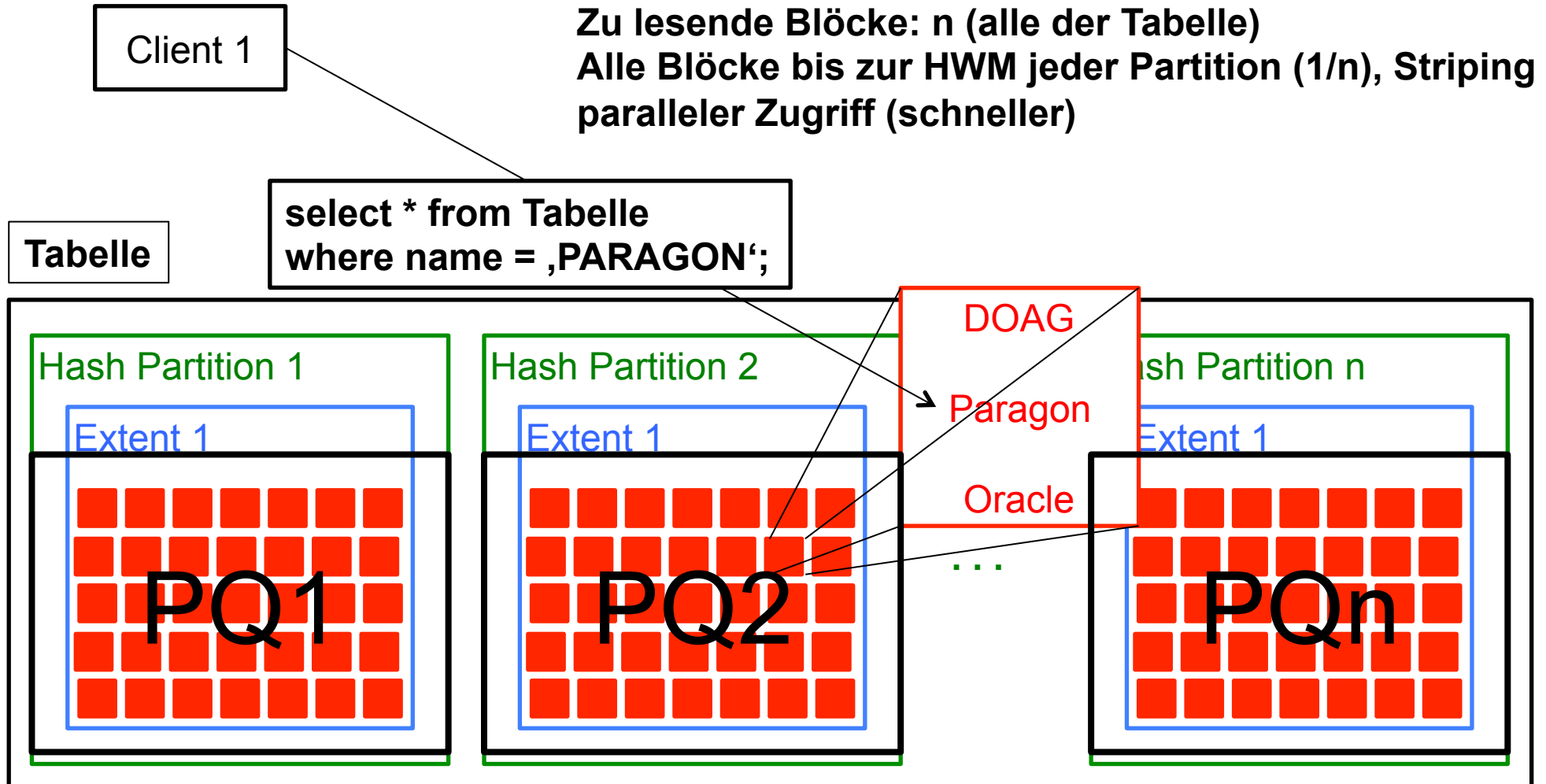
Zugriff auf Tabellen ohne Partitioning



Zugriff auf eine range partitionierte Tabelle (mandantengeführt, eindeutiger Partitionkey)



Zugriff auf eine hash partitionierte Tabelle (kein Mandant, kein eindeutiger Partitionkey)

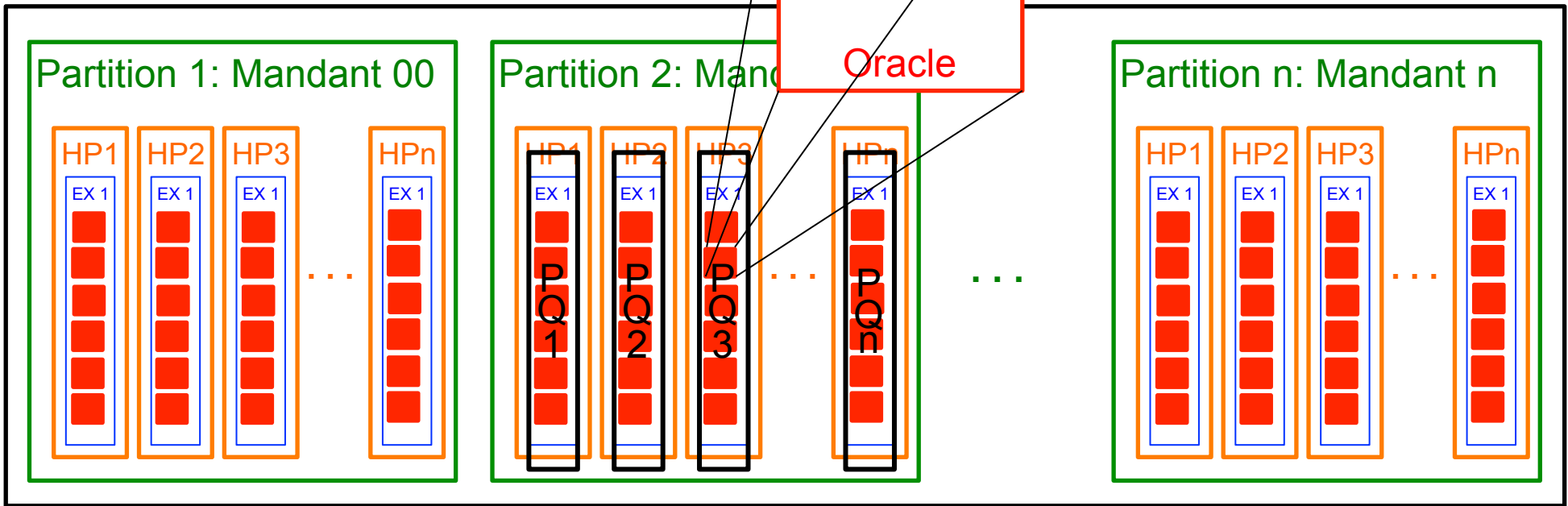


Zugriff auf eine composite partitionierte Tabelle (mandantengeführt, eindeutiger Partitionkey, sehr groß)

Client 1

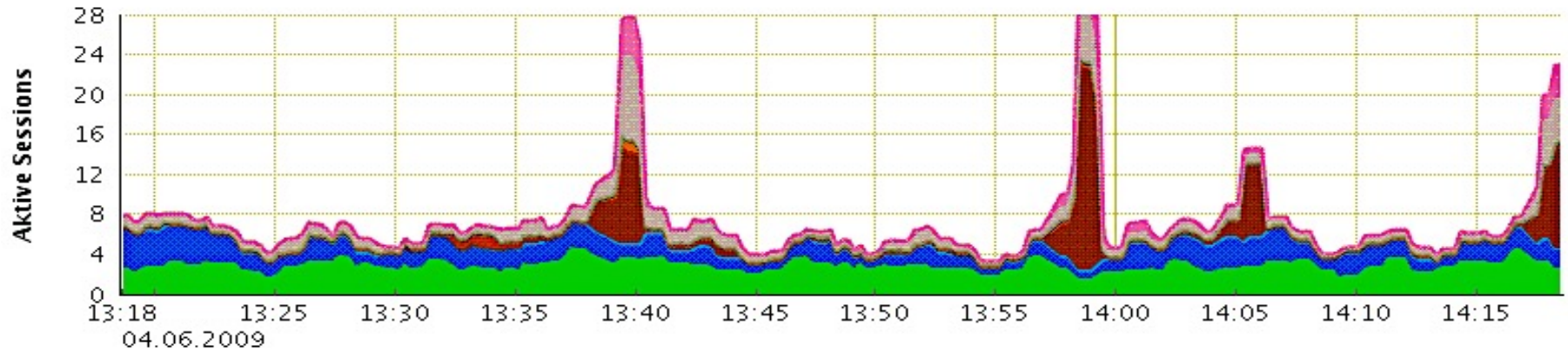
```
select * from Tabelle
where name = 'PARAGON'
and mandant = '01';
```

Zu lesende Blöcke: n (alle der Rangepartition, m Hashpartitionen, 1/n Blöcke)
 Alle Blöcke bis zur HWM der Hashpartitionen (1/m)
 PQ zur Beschleunigung

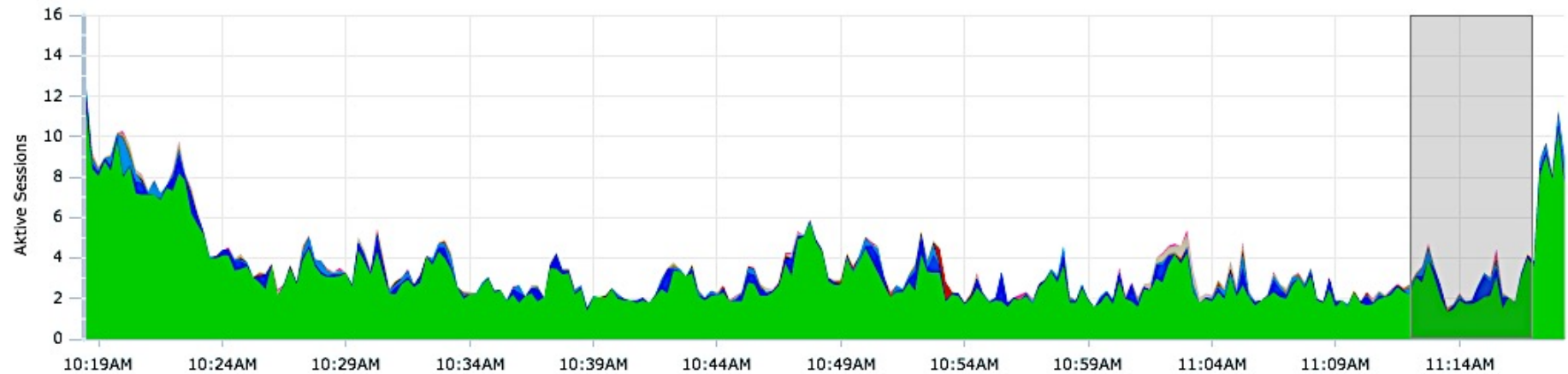


Darstellung der Probleme im Grid Control bzw. Cloud Control

Vorher:



Danach:



Ergebnisse des Partitionings

- **deutlich schnellere Zugriffe auf die Tabellen und Daten**
 - **zwischen 4 und 20 mal schneller**
- **schnellere applikative Prozesse**
 - **Beispiel Dispoliste (wichtiger interner Prozess)**
 - **Laufzeit ohne Partitioning: ca. 10 Minuten**
 - **Laufzeit mit Partitioning: 50 Sekunden**
- **Locking/ Blocking Locks mit Partitioning so gut wie nicht mehr vorhanden**
- **Exporte/ Importe via Datapump sehr viel schneller, da parallel gearbeitet werden kann**
- **Vorsicht bei Parallel Query (PQ), Overhead mit einberechnen, skaliert nicht unbedingt linear**

Zukünftige Änderung der Partitionierung zur Heiß-/ Kaltdataauslagerung

Tabelle T (mandantengestützt)

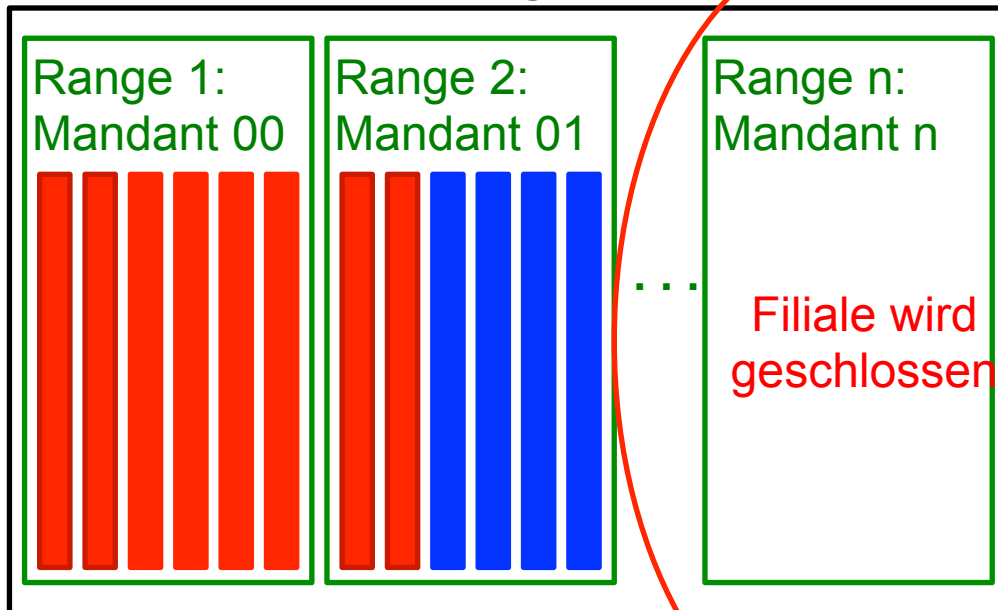
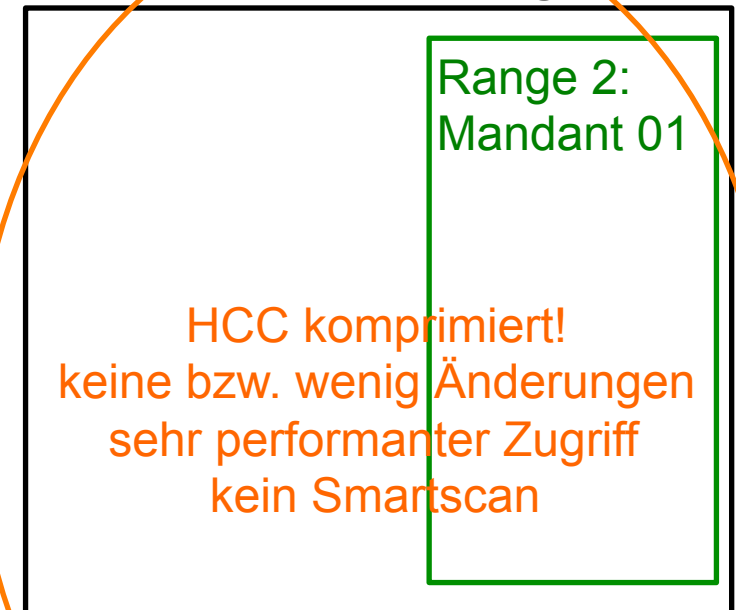


Tabelle T (mandantengestützt)



Infiniband 40 Gbit/s

Q & A

Vielen Dank!

Malthe Griesel, M.Griesel@paragon-data.de