

**performing  
databases**

# Oracle Einsteigerstream

## Das Transaktionskonzept und seine Umsetzung

Martin Klier

Performing Databases GmbH  
Mitterteich



**ORACLE**<sup>®</sup>  
ACE Director

```
SQL> SELECT compressed(know_how)
      2 FROM database_community
      3 FOR user_level='BEGINNER' ;
```

12 presentations selected.

Einsteiger-Stream auf der DOAG-Datenbank-Konferenz 2019

## Montag, 3. Juni

10:15	Codd & ACID – Ein Ausflug in die Datenbank-Theorie und -Geschichte	Markus Flechtner, Trivadis GmbH
11:15	Transaktionskonzept und Umsetzung	Martin Klier, Performing Databases GmbH
12:15	Überblick und Architektur	Martin Klier, Performing Databases GmbH
14:15	DB-Selbstverwaltung – das Data Dictionary	Markus Flechtner, Trivadis GmbH
15:15	Die Herausforderungen einer Datenbank	Martin Schmitter, RWE Supply+Trading GmbH
16:15	Oracle Multitenant	Ulrike Schwinn, Oracle Deutschland B.V. Co. KG

## Dienstag, 4. Juni

09:15	RMAN ist einfach	Johannes Ahrends, Carajan DB GmbH
10:15	Oracle Client Server Kommunikation	Dr. Matthias Mann, Value Transformation Services GmbH
11:15	Applikationsentwicklung: Microservices, REST und JSON aus der Datenbank	Robert Marz, its-people GmbH
12:15	Datentransfer mit Oracle-Tools	Christian Gohmann, Trivadis GmbH
15:15	Patching Oracle: Was muss ich tun?	Martin Bracher, Trivadis AG
16:15	Oracle Cloud Infrastructure: Netzwerk-Einrichtung für DBAs	Robert Marz, its-people GmbH

# Speaker

- Martin Klier
- Solution Architect and Database Expert
- My focus:
  - Performance + Tuning
  - highly available Systems
  - Cluster and Replication
- Linux since 1997
- Oracle Database since 2003



**ORACLE®**  
ACE Director



performing  
databases

Martin Klier // [twitter.com/MartinKlierDBA](https://twitter.com/MartinKlierDBA)  
[www.performing-databases.com](http://www.performing-databases.com)

# Performing Databases

- Experts for Database Technology
  - Concepts and Project Competence
  - Architektur- und System planning
  - Licensing
  - Implementation and Troubleshooting
- Contact
  - Performing Databases GmbH  
Wiesauer Straße 27  
95666 Mitterteich // Germany
  - Web: <http://www.performing-databases.com>
  - Twitter: @PerformingDB

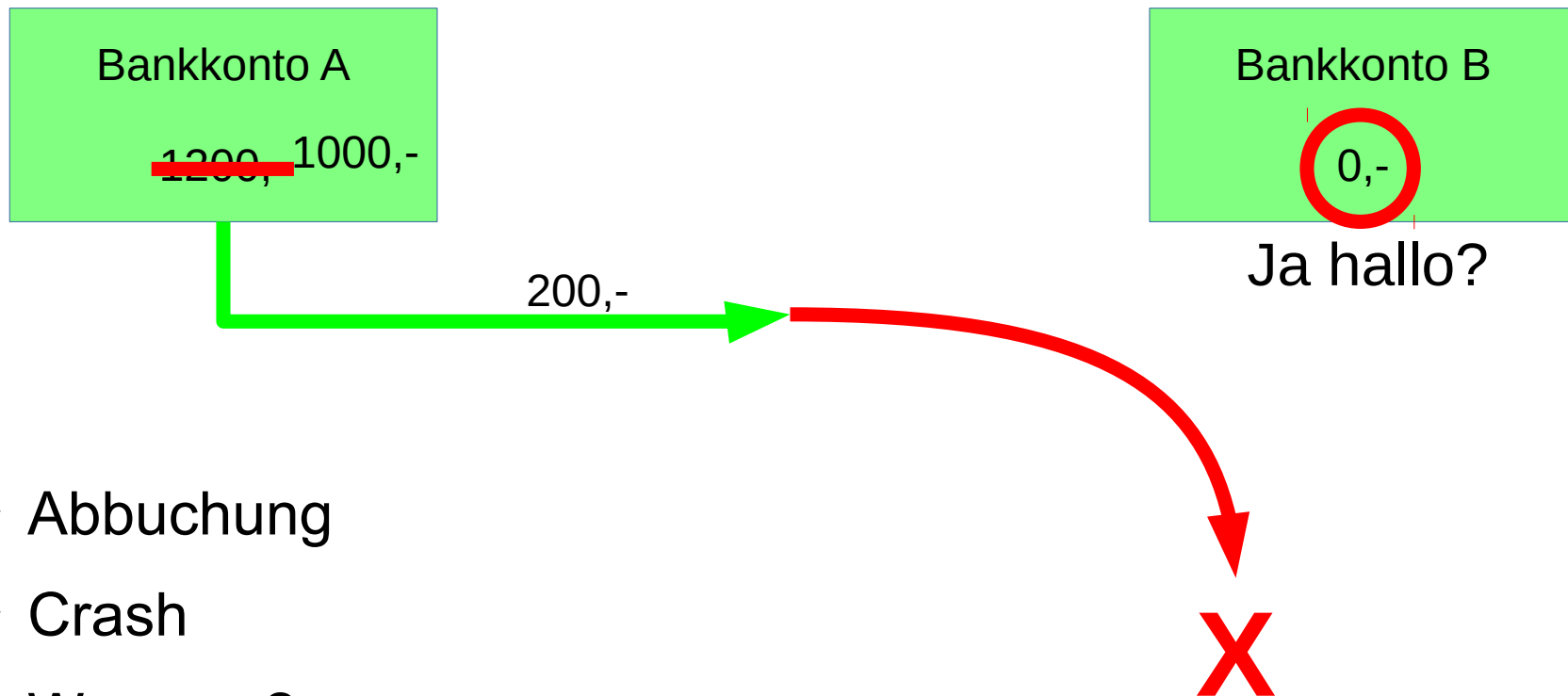


# Transaktionen



# Transaktionen

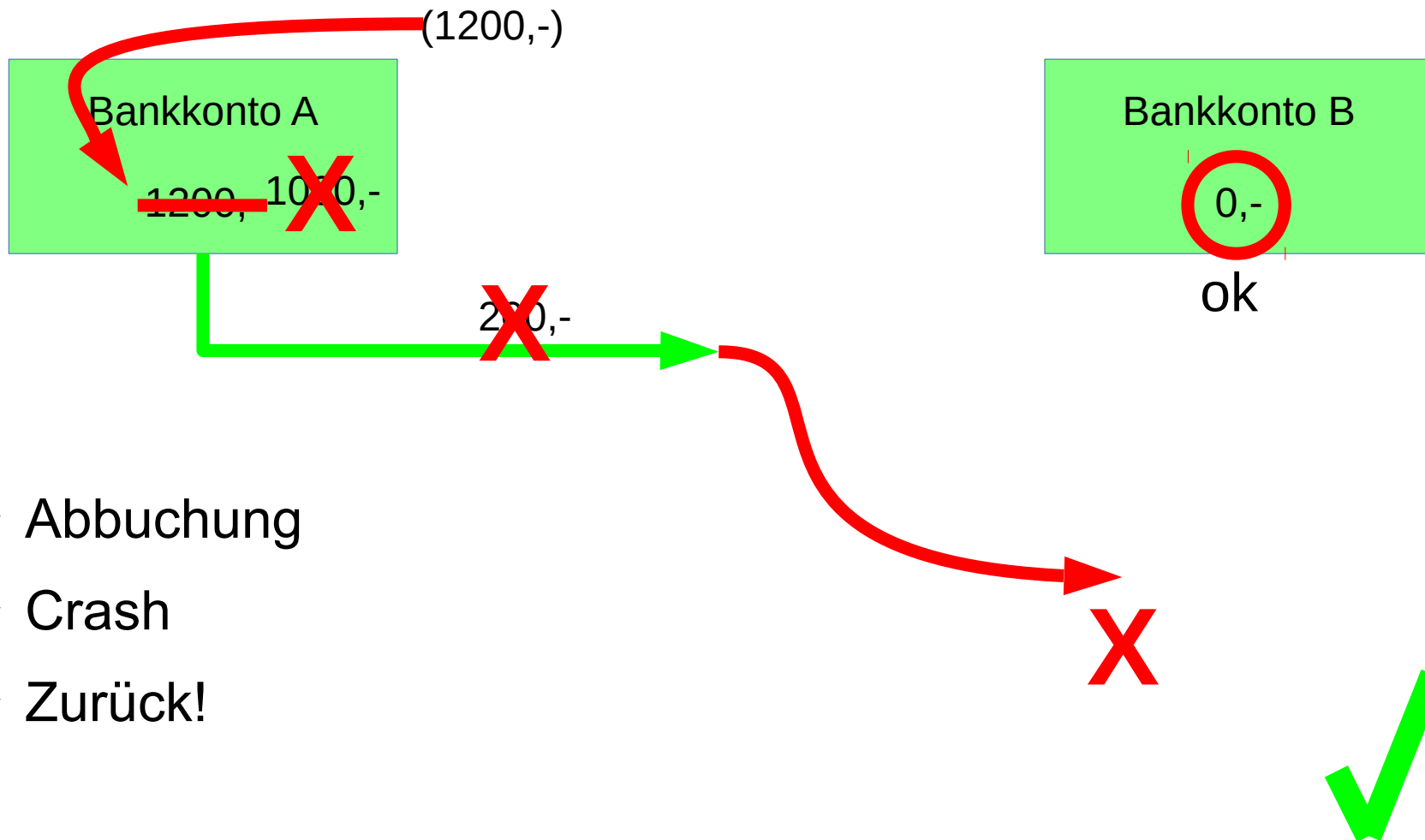
## Beispiel „Überweisung“



- Abbuchung
- Crash
- Was nun?

# Transaktionen

## Beispiel „Überweisung“



- Abbuchung
- Crash
- Zurück!

**Google darf gerne BASE für seine Suchergebnisse nutzen.  
Mir ist egal, ob es 999.963 oder 998.763 Treffer für meine Suche gibt.**

**Aber meine Bank macht das für die Anzeige  
meines Kontostandes und für meine Überweisungen hoffentlich nicht.**

Das musste mal gesagt werden.  
Danke, Markus Flechtner! :)

# Transaktionen

- **Atomicity** „Alles oder nichts“
- **Consistency** „Kein undefinierter Zustand“
- **Isolation** „Read Committed“
- **Durability** „Zugesicherte Speicherung“

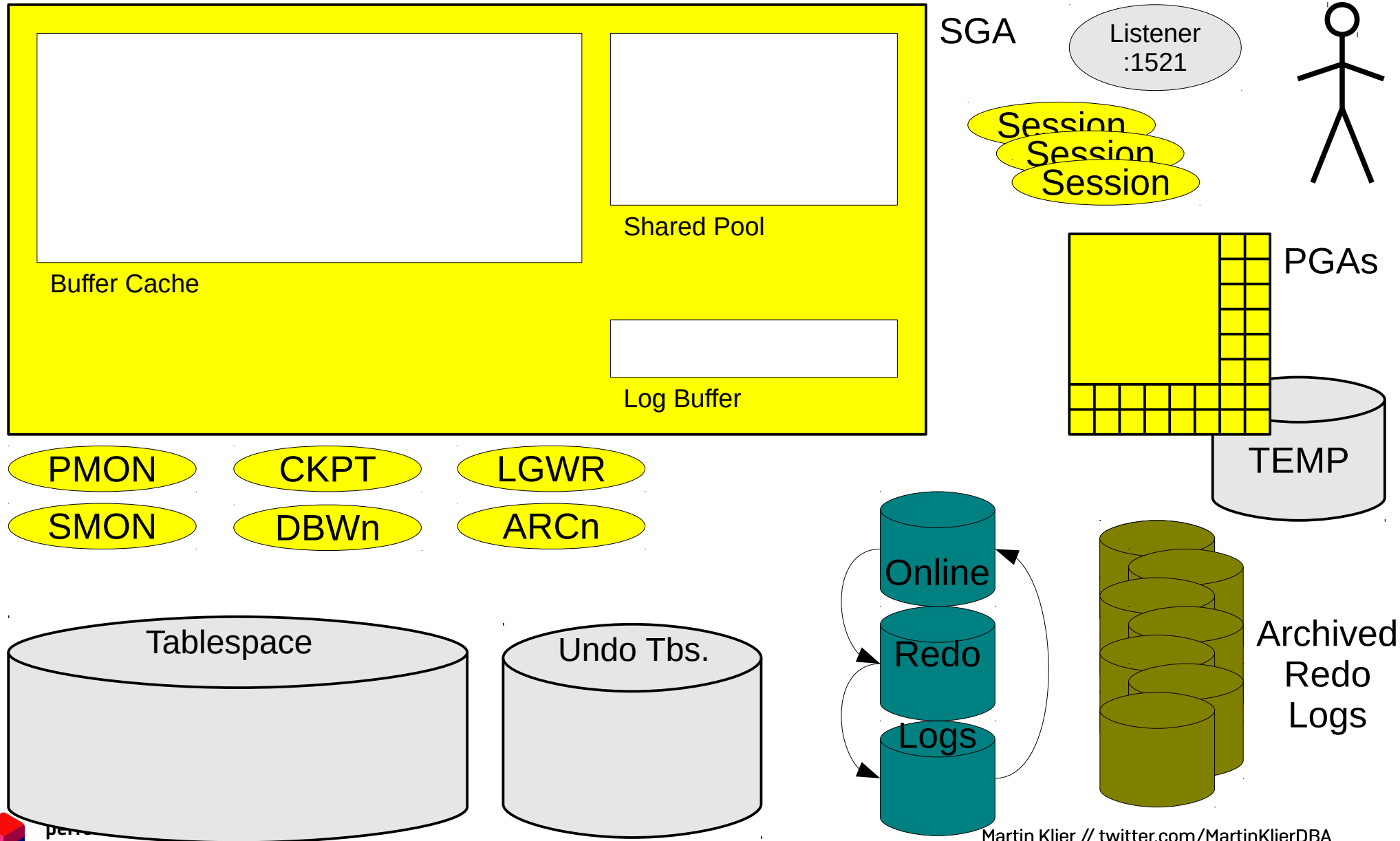
„Eine Transaktion **fasst** in sich

- mehrere logisch zusammengehörige
- jedoch nicht zwangsweise technisch voneinander abhängige Arbeitsschritte **zusammen**“

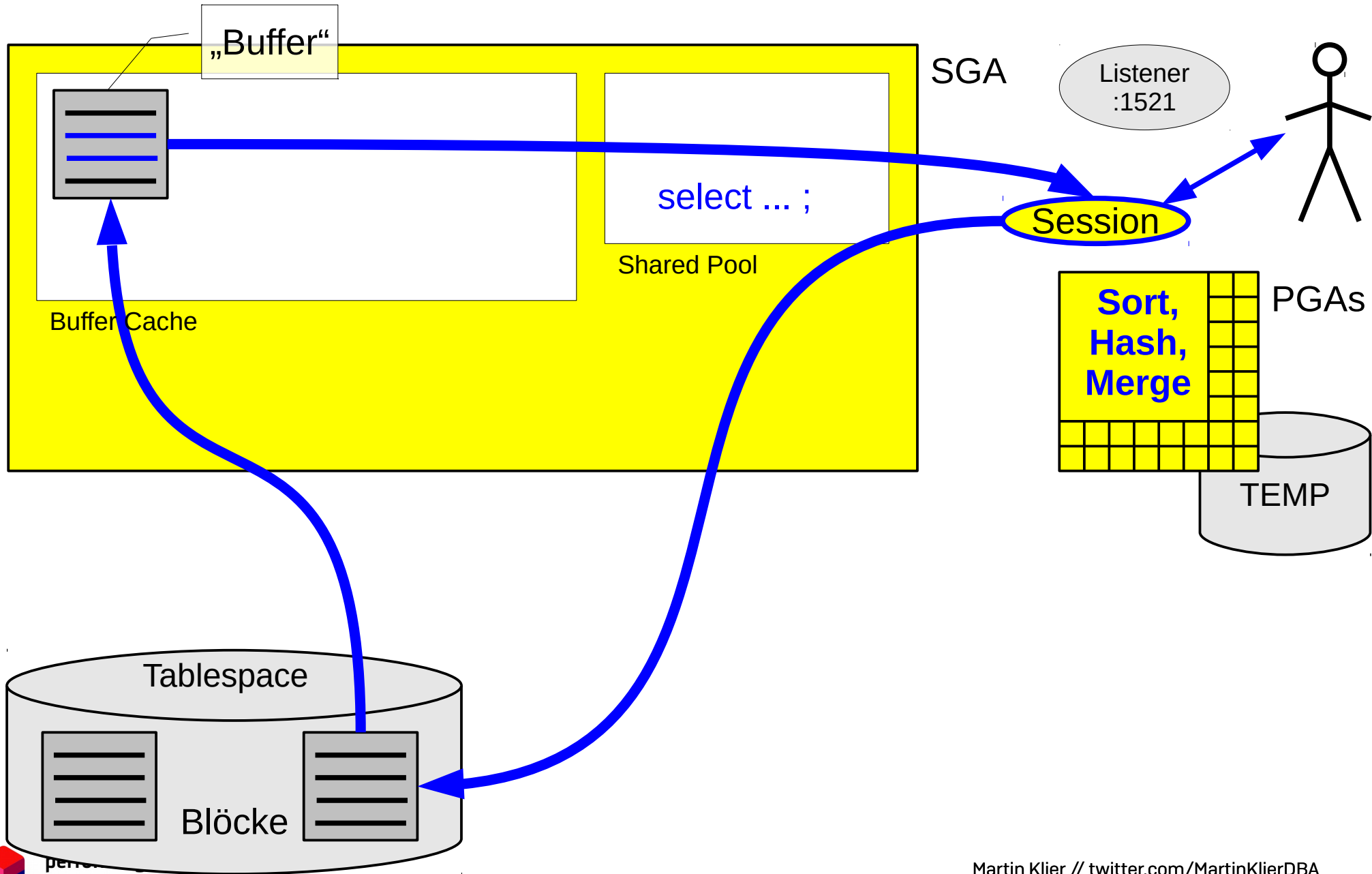
## Wie wird das technisch gelöst?

# Grundfunktionen der Oracle Engine

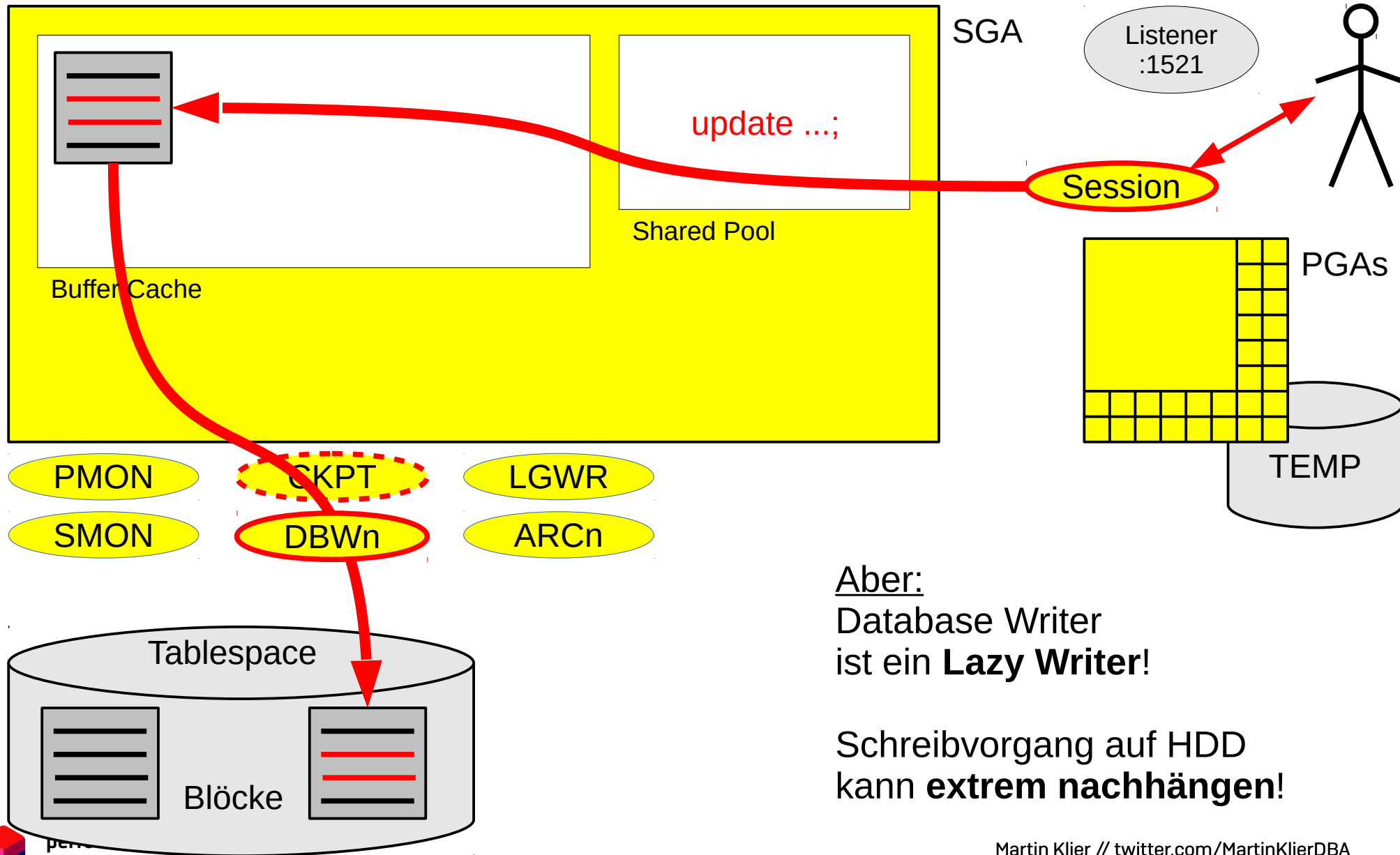
# Architekturschema [vereinfacht]



# Lesevorgang



# Daten-Änderung, Database Writer



Aber:  
Database Writer  
ist ein **Lazy Writer**!

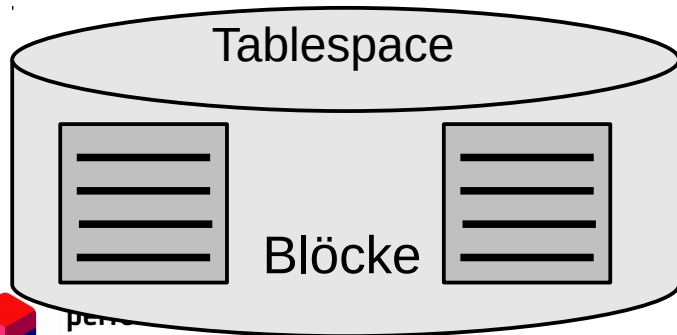
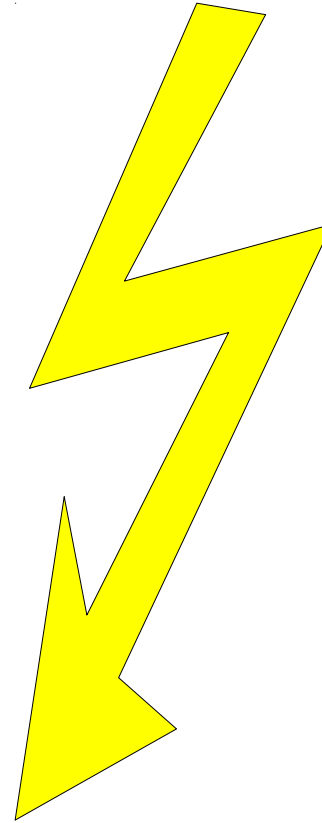
Schreibvorgang auf HDD  
kann **extrem nachhängen!**



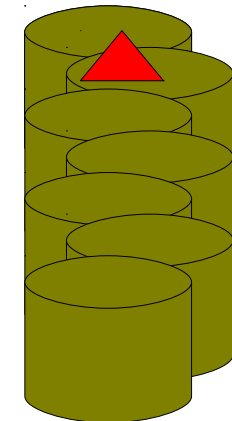
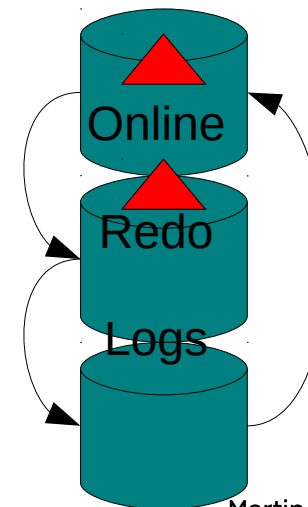
(nur Redo)



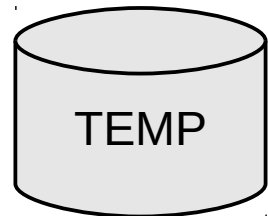
# Crash



performing  
databases



Archived  
Redo  
Logs

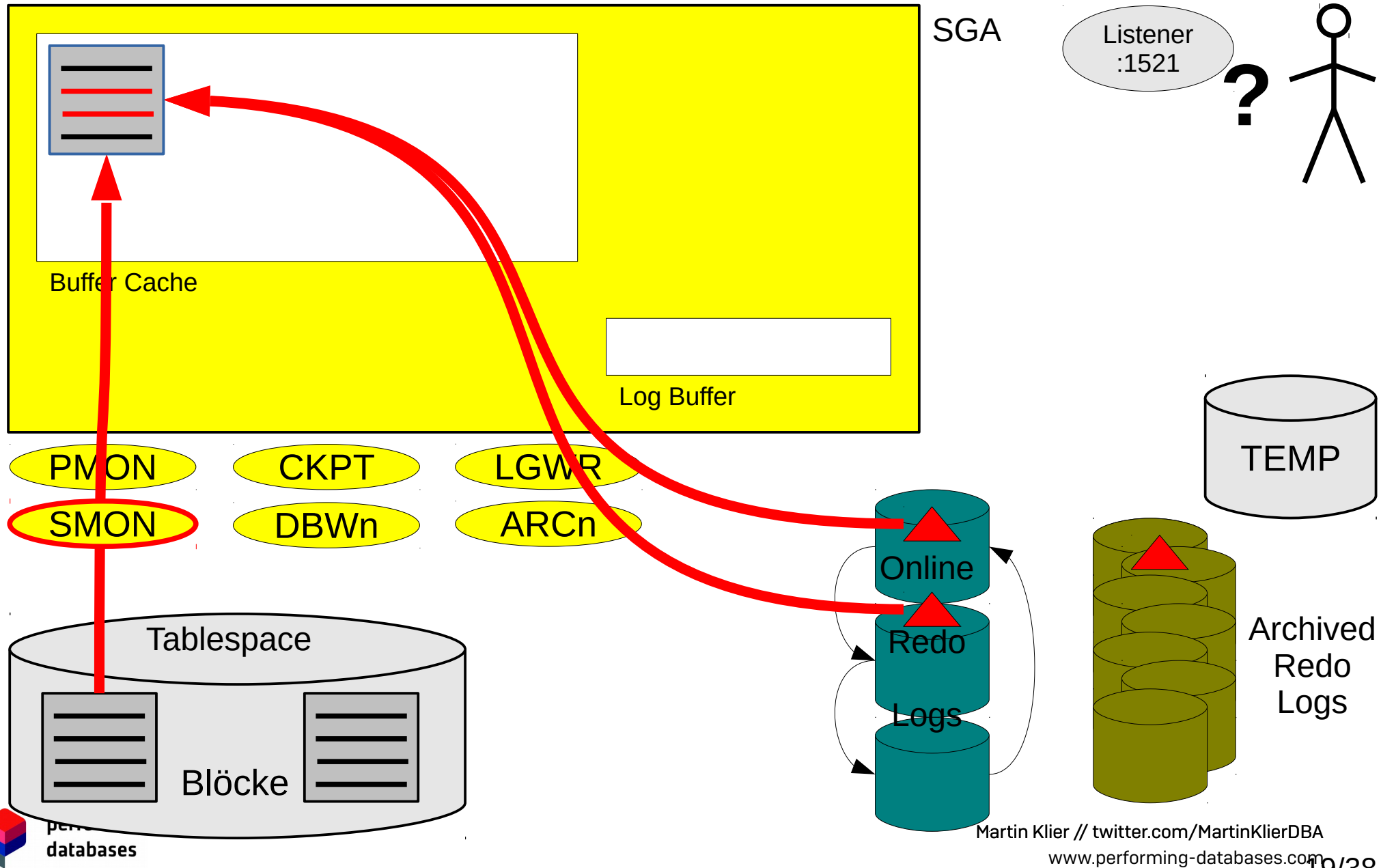


Martin Klier // [twitter.com/MartinKlierDBA](https://twitter.com/MartinKlierDBA)

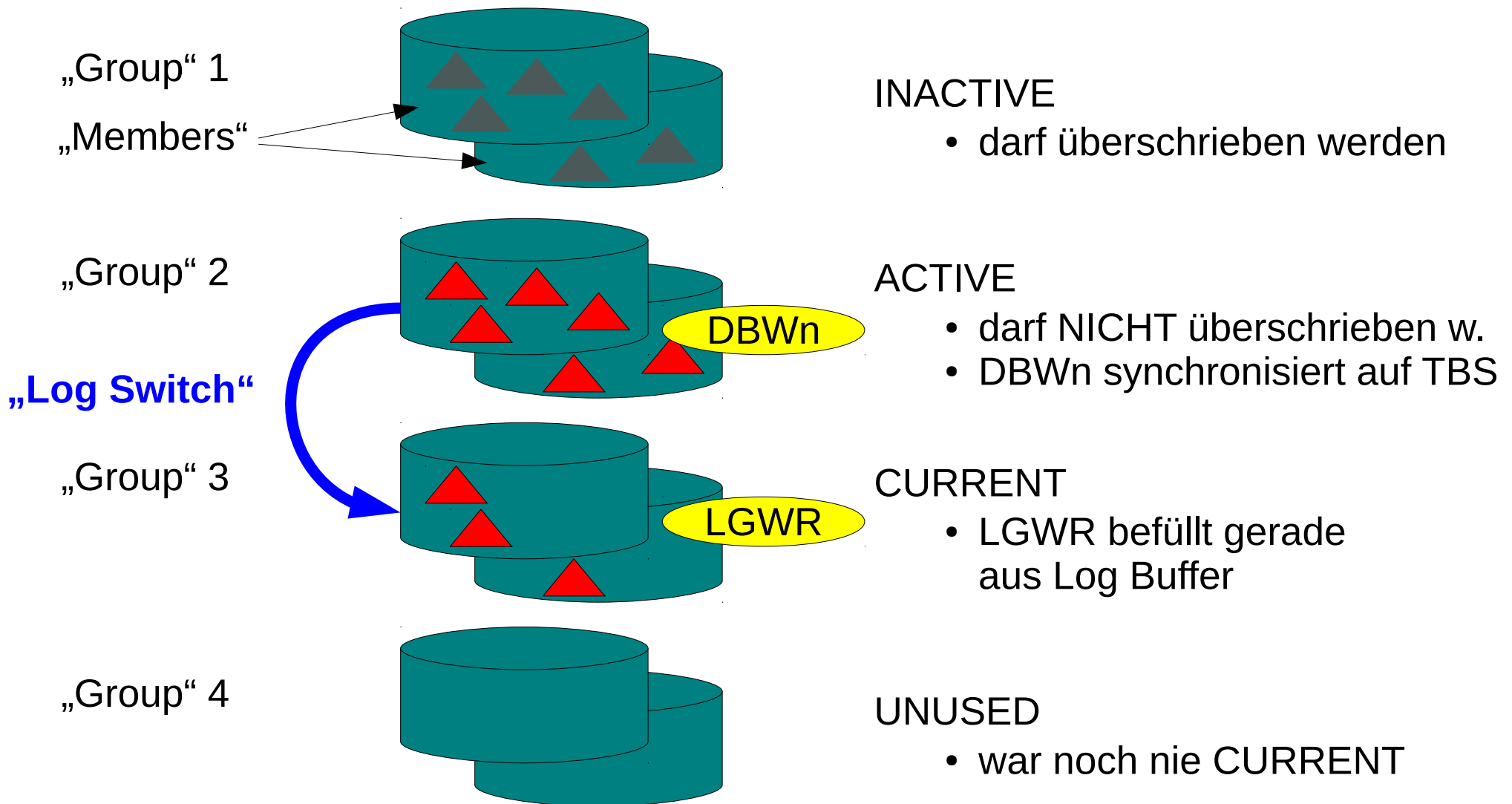
[www.performing-databases.com](http://www.performing-databases.com)

# Crash Recovery

[nur Redo]

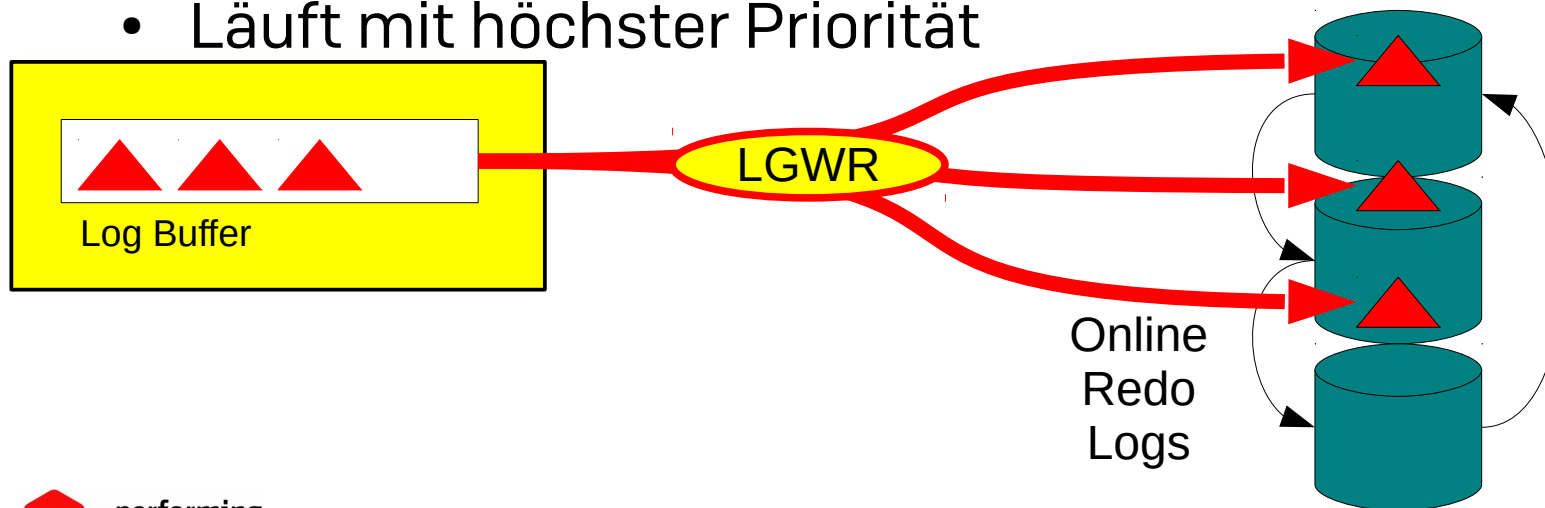


# Detail: Online Redo Logs



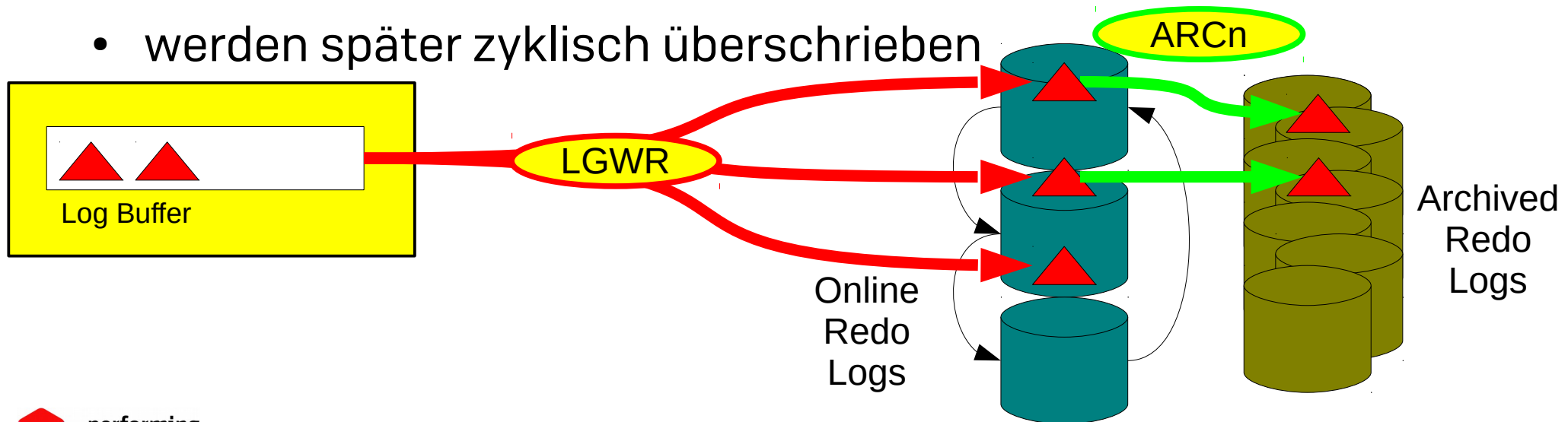
# Log Writer

- Schreibt den Log Buffer ins Online Redo Log
  - bei jedem Commit
  - wenn Log Buffer voll
  - nach spätestens 3 Sekunden
- Laufzeit bestimmt Commit-Zeit
- Läuft mit höchster Priorität



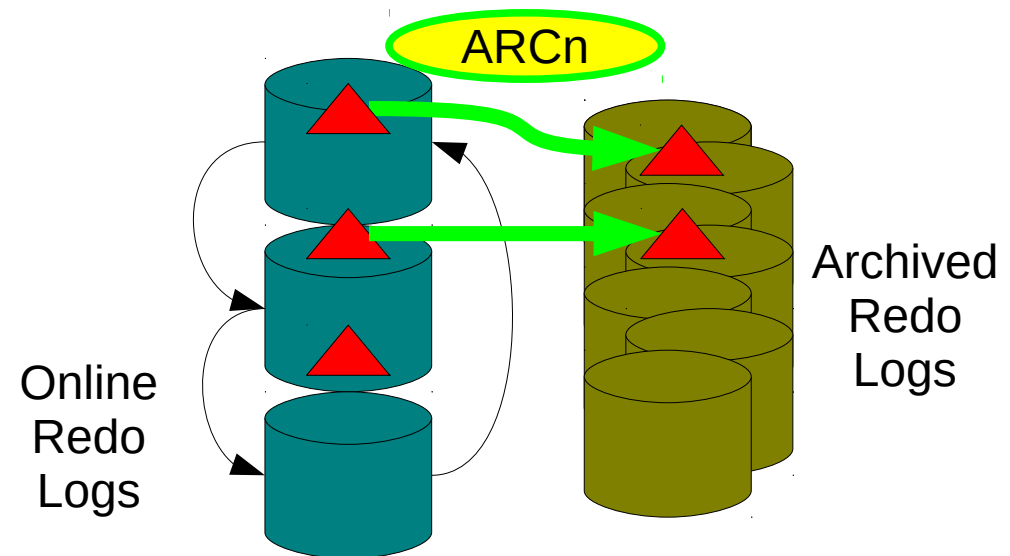
# Online Redo Logs

- dienen der Absicherung des Cache
- müssen aufbewahrt werden so lange DBWn nicht in Tablespace geschrieben hat
- müssen aufbewahrt werden so lange ARCn nicht in Archived Redo Log geschrieben hat
- werden später zyklisch überschrieben



# Archived Redo Logs

- werden pro RL-“Group” vom Archiver (ARCn) erzeugt
  - frühestens: nach Log Switch
  - spätestens: vor Überschreiben Online Redo Log
- dienen dem **Nachfahren von verlorenen Operationen beim Recovery** aus einem Backup
  - Complete Recovery
  - Point-in-Time-Recovery
- werden min. bis zum nächsten Backup aufbewahrt
  - nie überschrieben
  - später gelöscht



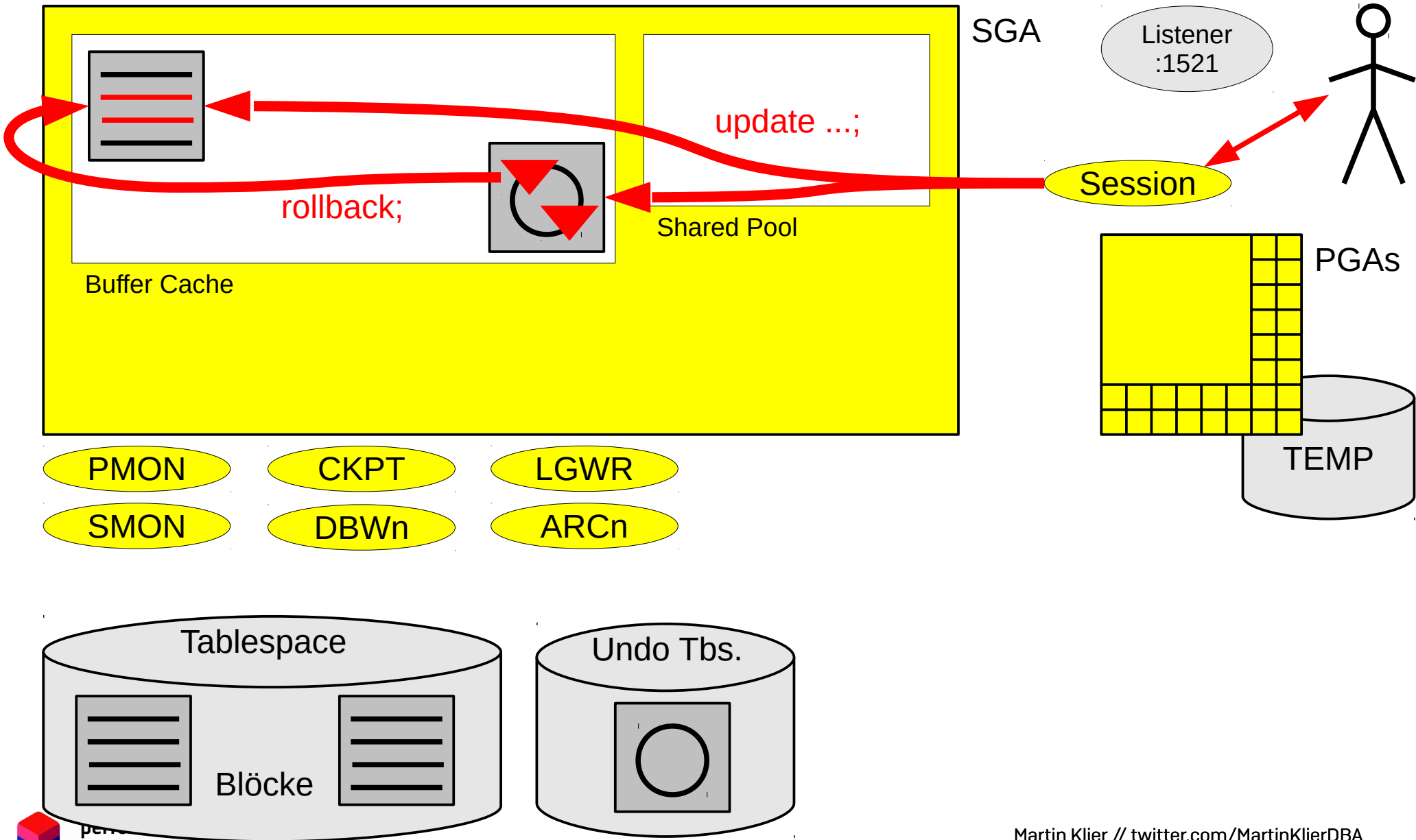
# Das war aber nur die halbe Wahrheit ...

Das benötigte Transaktionsverfahren macht die Sache etwas komplizierter.

Was ist UNDO...?

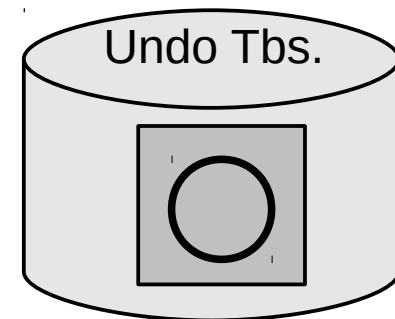


# Undo-Konzept

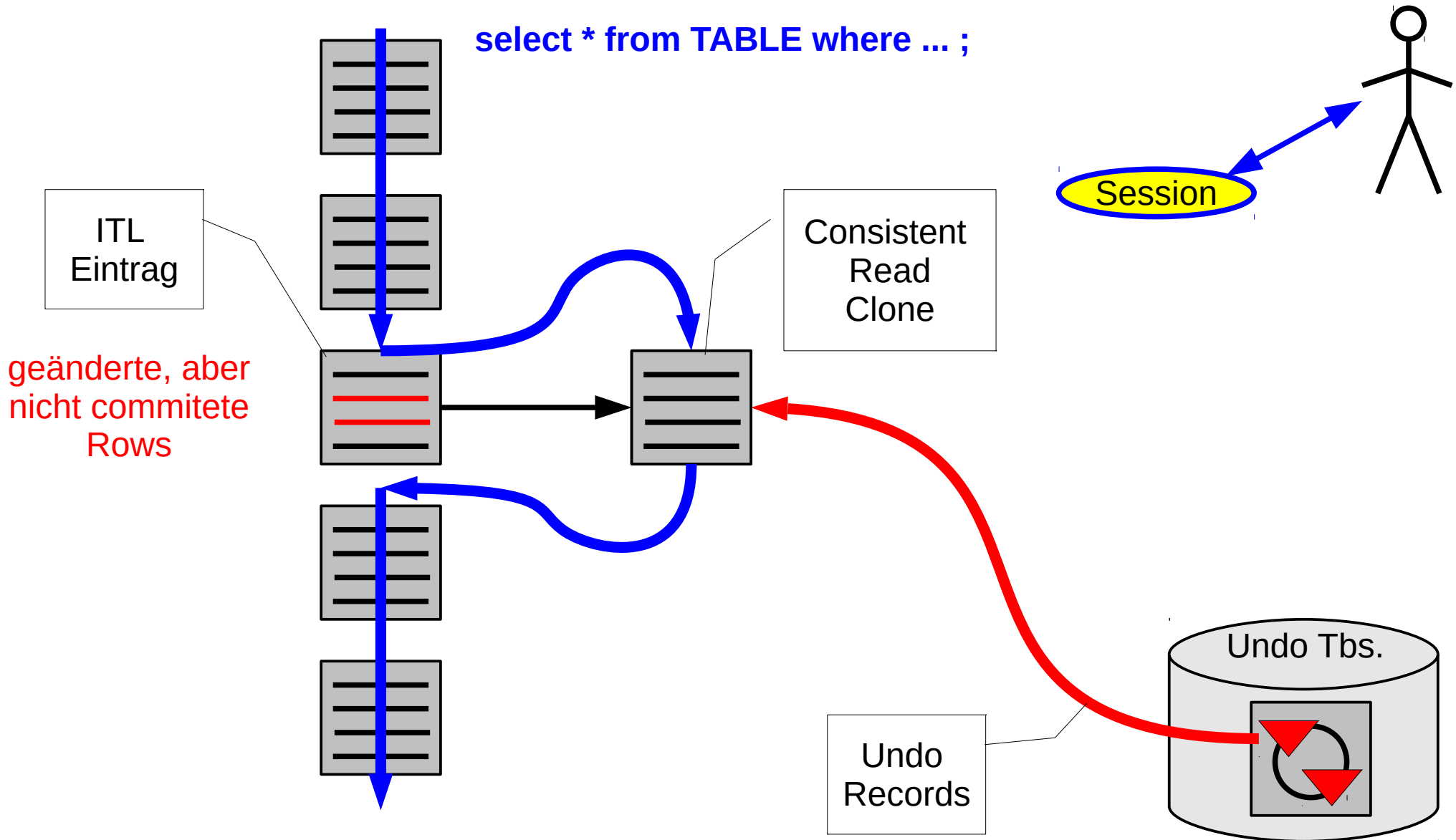


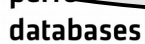
# Undo Records

- dienen dem Zurückrollen von Änderungen
- kein „Undo-“Hintergrundprozess  
[Session / Job / etc. schreibt selbst]
- werden später überschrieben („Ringpuffer“)
  - frühestens: nach Abschluss der Transaktion
  - meist: nach Ablauf der Undo Retention
  - spätestens: bei Platzbedarf
- weitere Verwendung
  - für konsistentes Lesen (stets)
  - Flashback-Technologien (optional)

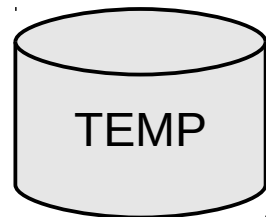
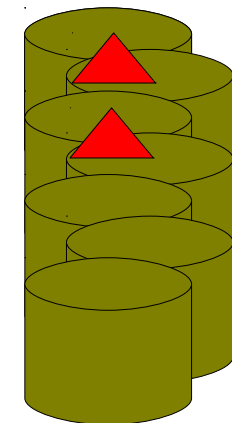
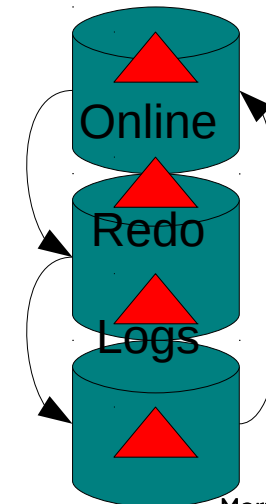
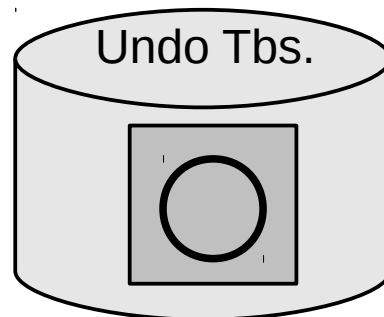
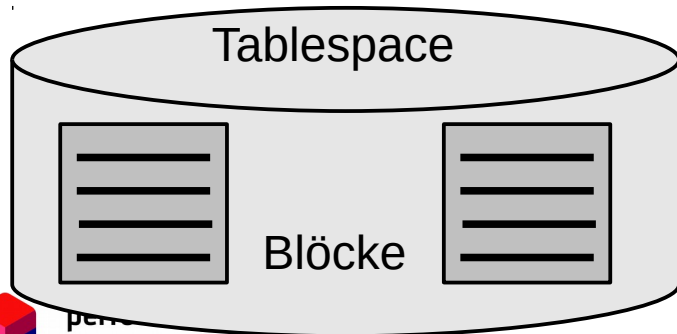
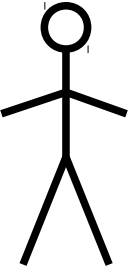
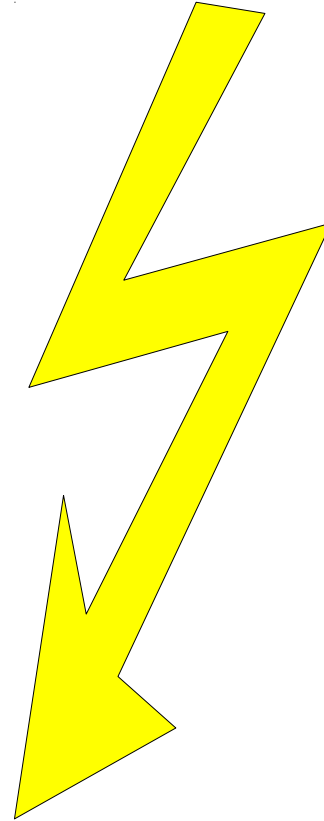


# Undo: Konsistentes Lesen

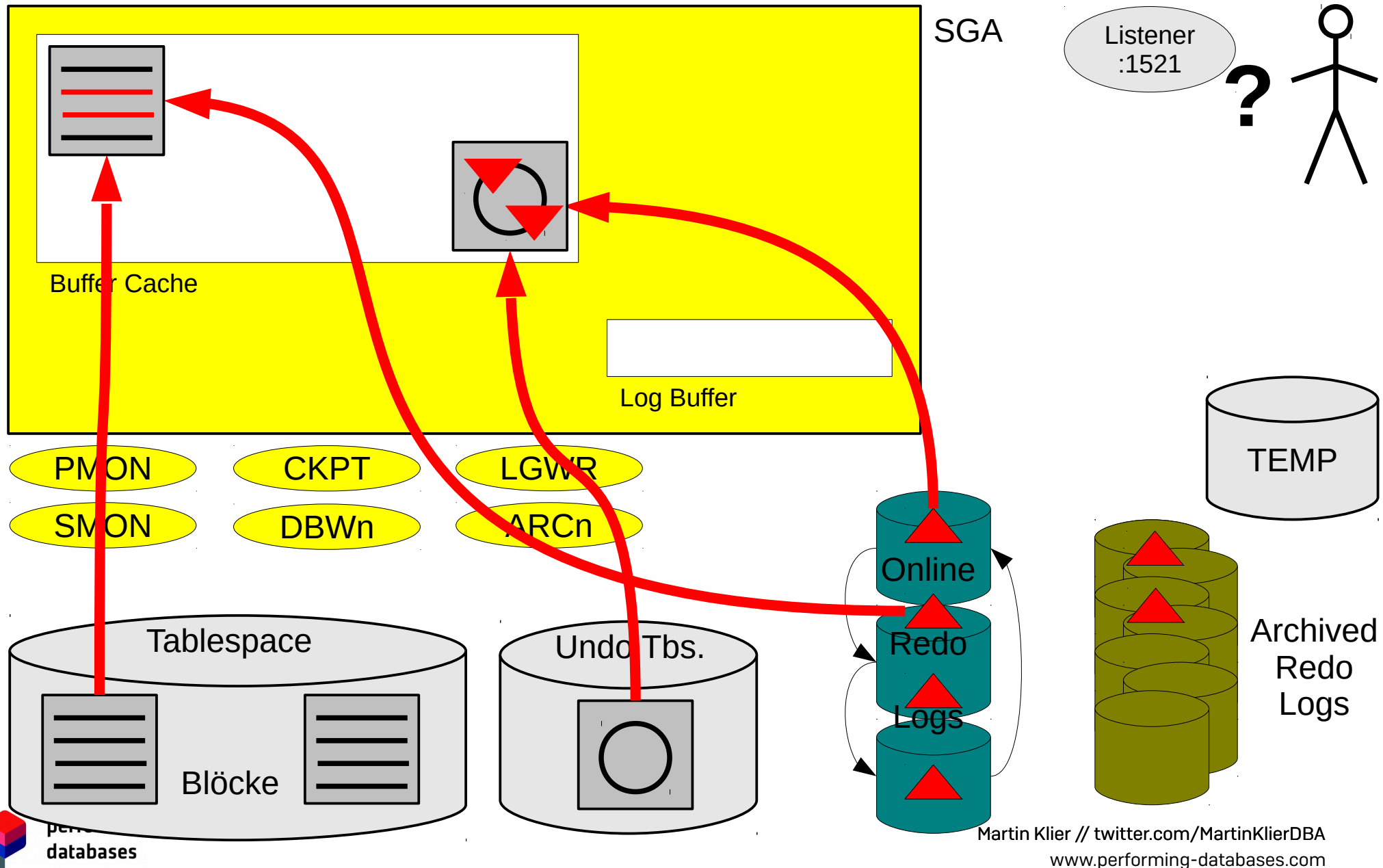




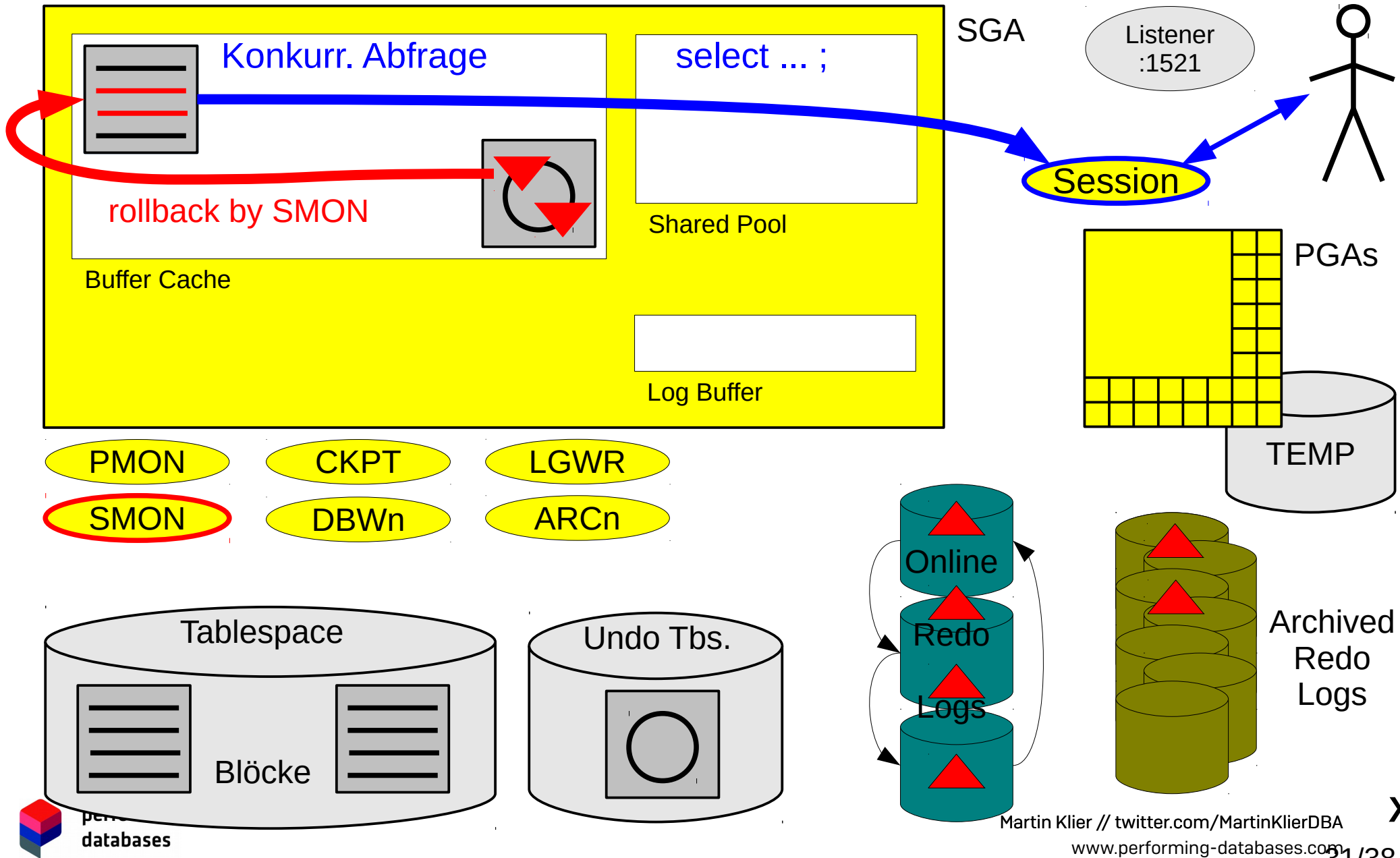
# Crash



# Crash Recovery



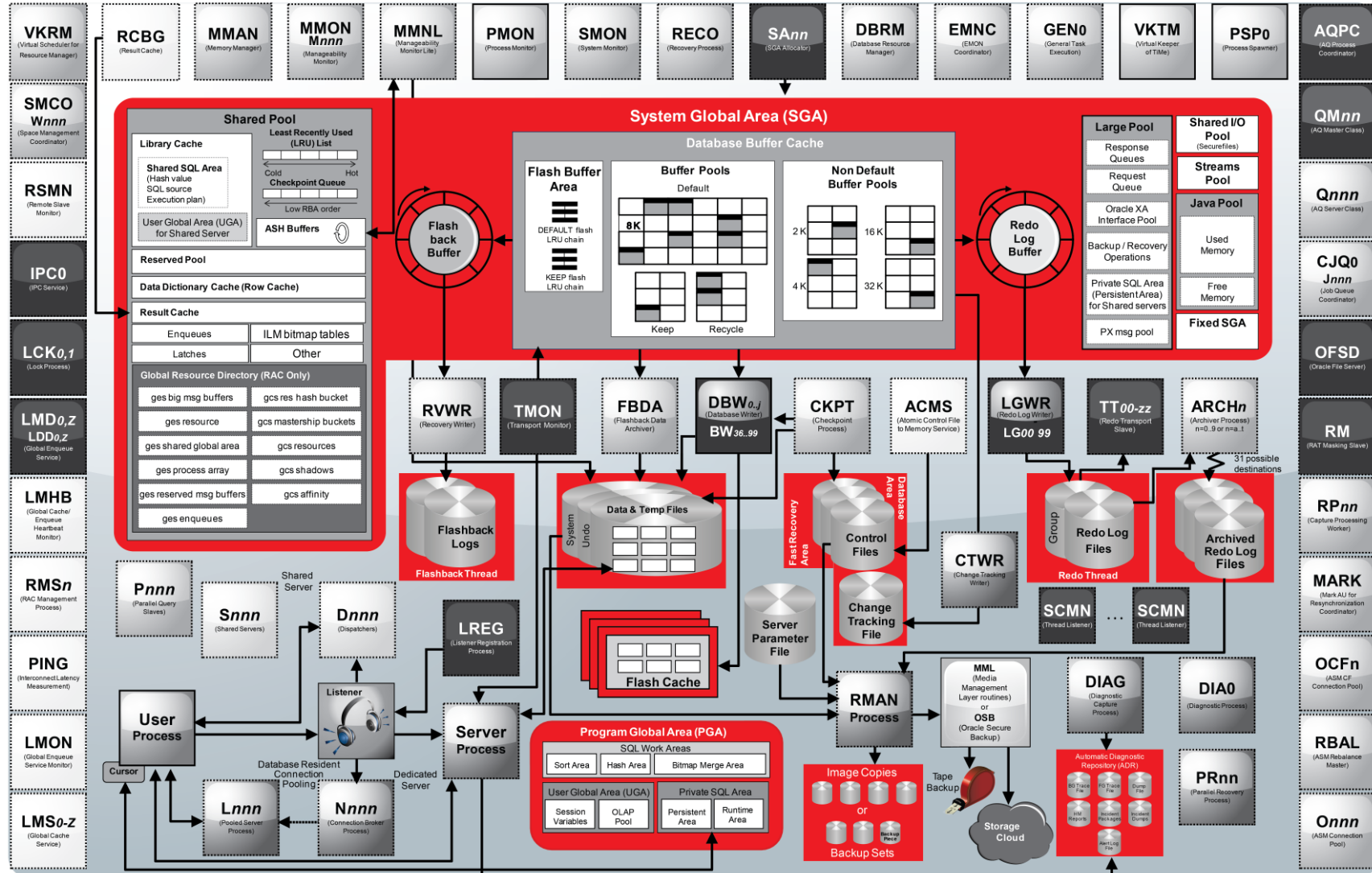
# SMON nach Crash Recovery



# Architektur 12.1

[Oracle Doc]

## ORACLE<sup>®</sup> 12<sup>c</sup> DATABASE Architecture Diagram



ORACLE<sup>®</sup>

Copyright © 2012, Oracle and/or its affiliates. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.



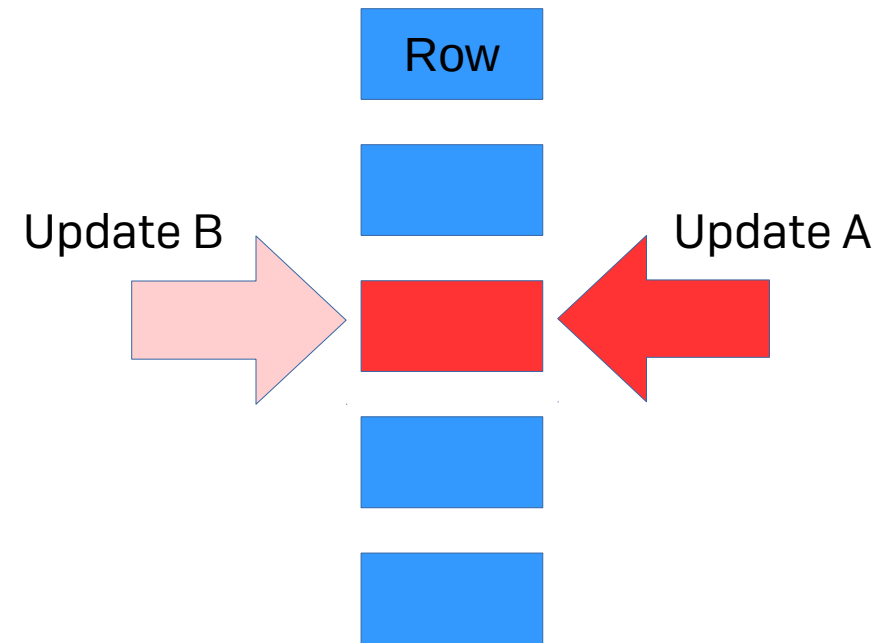
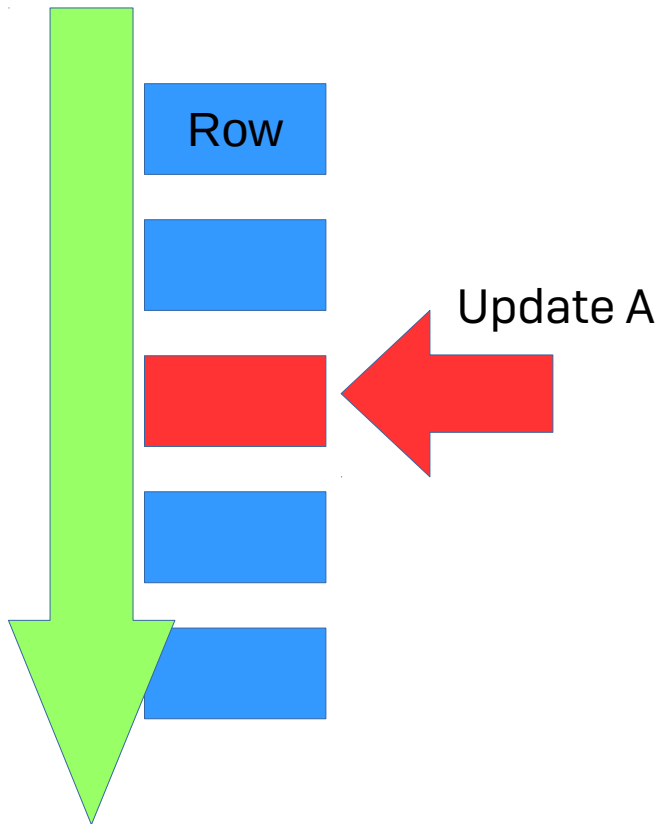
# Sperren

Irgendwas ist ja immer ... :)



# Sperren

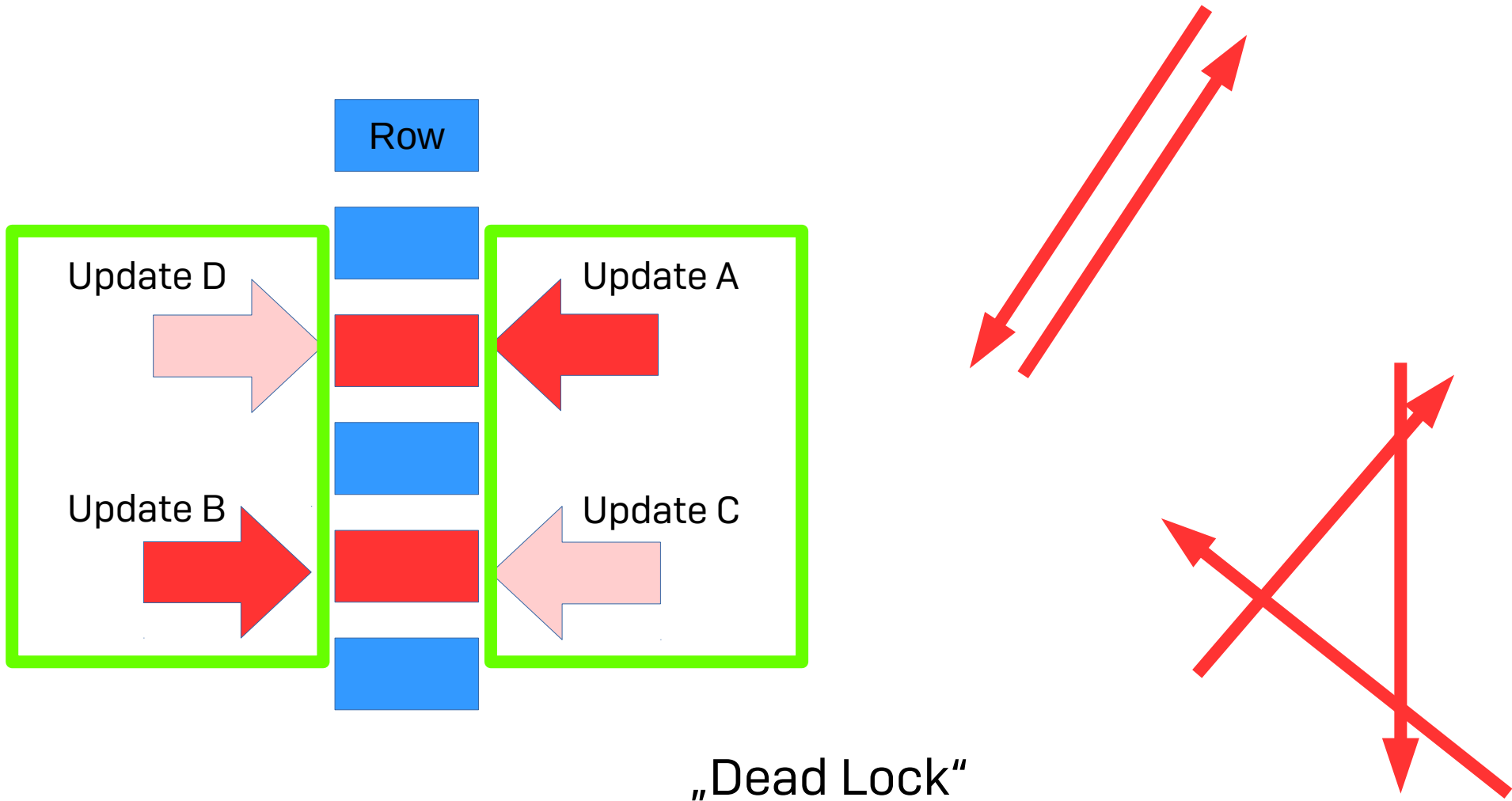
Select



„Row Lock Contention“



# Sperren



„Dead Lock“

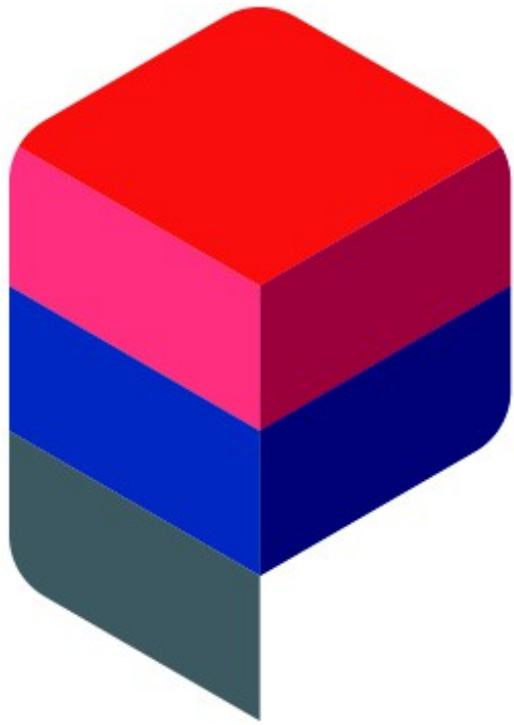
=> ORA-0060



# DANKE!







**performing  
databases**