




ORACLE®  
DATABASE

# Agenda

1. Ausgangslage
2. Übersicht ORACLE Architektur
3. Komponenten einer Datenbank
4. Binaries
5. Datenfiles
6. Memory Bereiche
7. Dokumentation

# Agenda

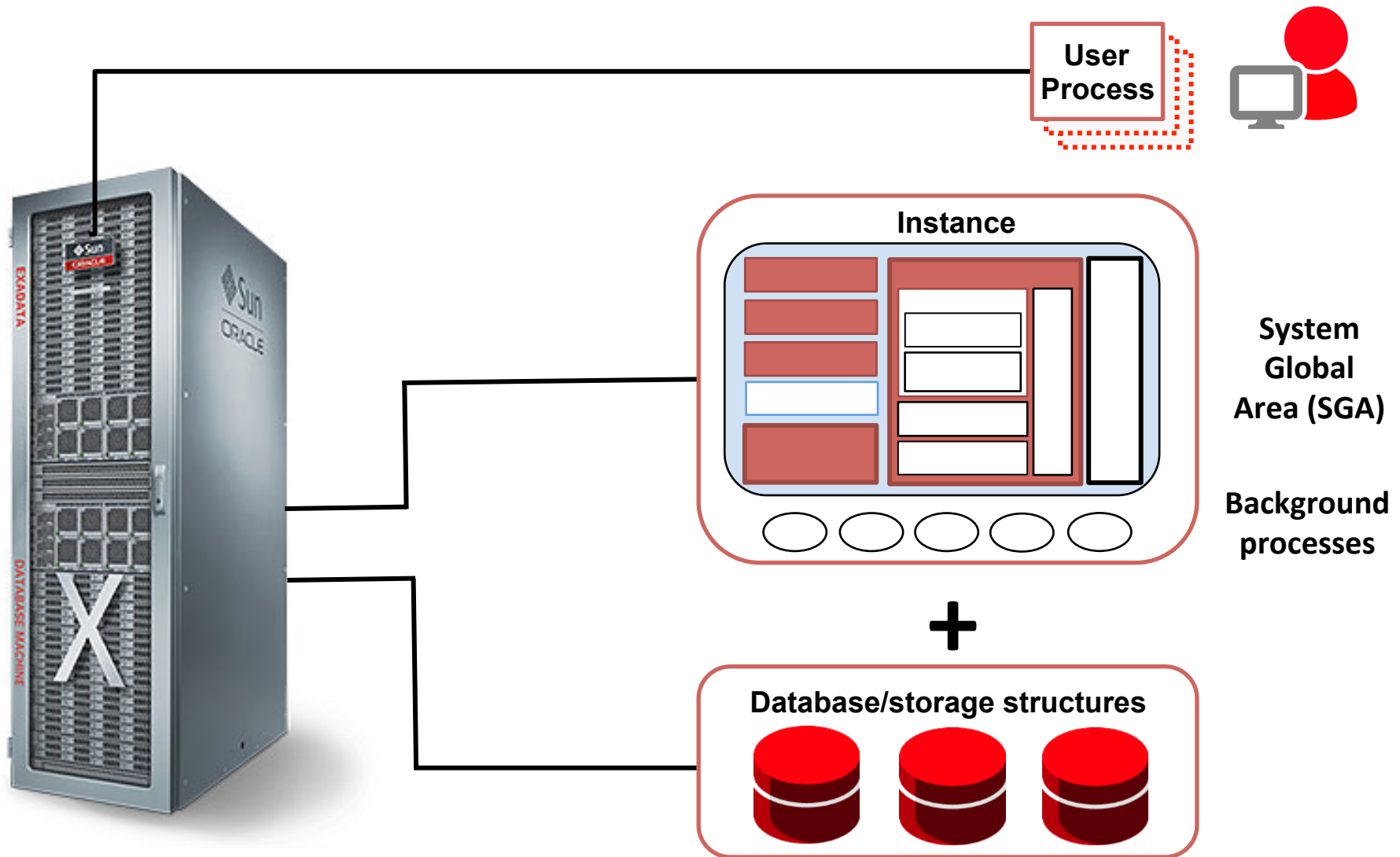
1. Ausgangslage
2. Übersicht ORACLE Architektur
3. Komponenten einer Datenbank
4. Binaries
5. Datenfiles
6. Memory Bereiche
7. Dokumentation

- Dies ist ein grober Überblick über die Funktionsweise von Relationalen Datenbanken an Hand von ORACLE
- Diese Übersicht ist nicht abschliessend komplett!
  - Es wurden bewusst Sachen ausgelassen, da eine vollständige Beschreibung von ORACLE den Rahmen dieses Moduls sprengen würde
  - Wer zusätzliche Informationen möchte oder wer sich tiefer in das Gebiet ORACLE einarbeiten möchte, kann sich gerne an mich wenden.
- An verschiedenen Stellen werden “Best Practices” beschrieben
- Auf Best Practices wird mit dem Symbol  hingewiesen

# Agenda

1. Ausgangslage
2. Übersicht ORACLE Architektur
3. Komponenten einer Datenbank
4. Binaries
5. Datenfiles
6. Memory Bereiche
7. Dokumentation

# Übersicht ORACLE Architektur

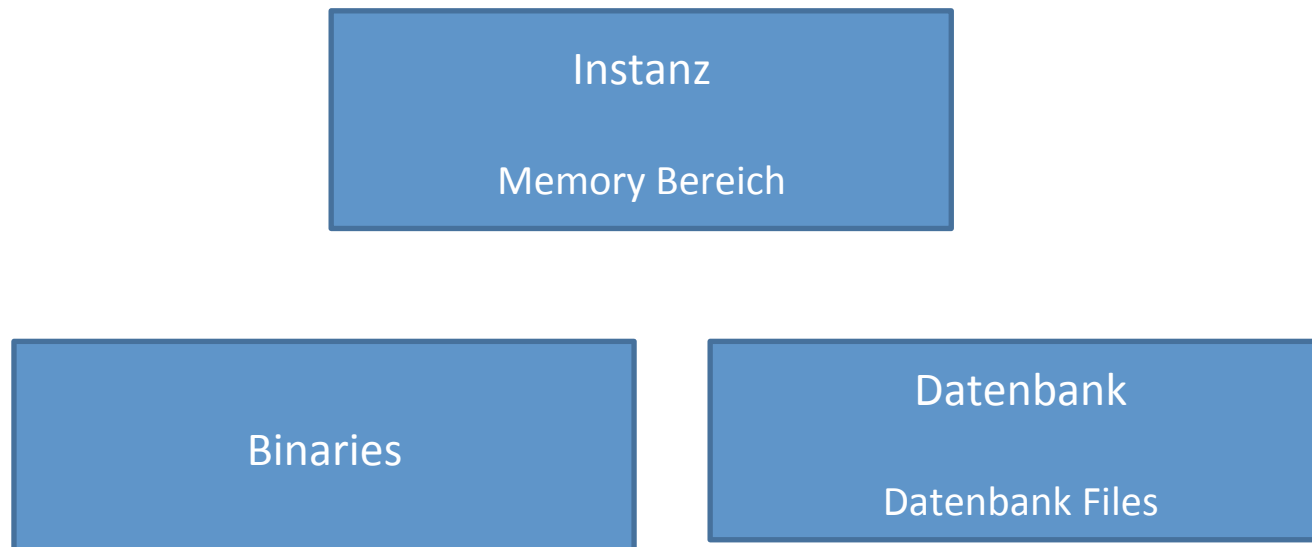


# Agenda

1. Ausgangslage
2. Übersicht ORACLE Architektur
3. **Komponenten einer Datenbank**
4. Binaries
5. Datenfiles
6. Memory Bereiche
7. Dokumentation

# Komponenten einer Datenbank

- Datenbanken bestehen aus 3 verschiedenen Komponenten:
  - Binaries, Datenbank und Instanz
- Vorsicht Verwechslungsgefahr!
  - Wenn man von “Datenbank” spricht, sind die persistenten Datenfiles gemeint
  - Im HA Umfeld (RAC, Real Application Cluster) kann eine Datenbank von mehreren Instanzen angesprochen werden





# Agenda

1. Ausgangslage
2. Übersicht ORACLE Architektur
3. Komponenten einer Datenbank
4. **Binaries**
5. Datenfiles
6. Memory Bereiche
7. Dokumentation

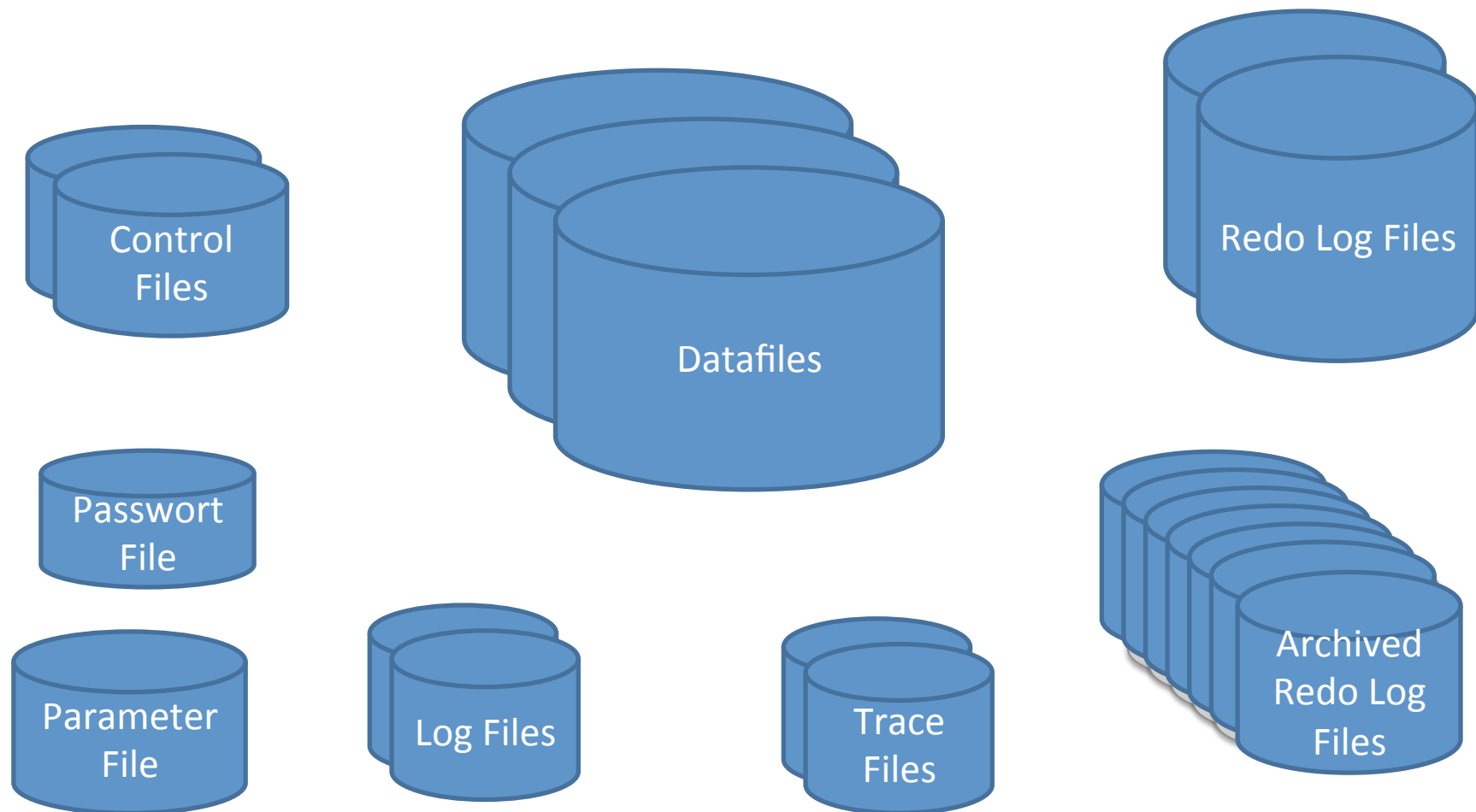
- Die ORACLE Binaries liegen im “ORACLE\_HOME”
- Wichtige Verzeichnisse:
  - `${ORACLE_HOME}/bin`
    - Beinhaltet viele Executables wie SQL\*Plus, impdp etc.
    - Sollte immer im Pfad liegen
  - `${ORACLE_HOME}/dbs` (UNIX) bzw. `%ORACLE_HOME%\database` (Windows)
    - Default location für Parameter File, Passwort File zT auch Datenfiles etc.
  - `${ORACLE_HOME}/network/admin`
    - Default Location für SQLNet Konfigurationsfiles (listener.ora, tnsnames.ora, sqlnet.ora, ldap.ora)
  - `${ORACLE_HOME}/OPatch`
    - Binaries zum Patchen von ORACLE

# Agenda

1. Ausgangslage
2. Übersicht ORACLE Architektur
3. Komponenten einer Datenbank
4. Binaries
5. **Datenfiles**
6. Memory Bereiche
7. Dokumentation

# Datenfiles

- Eine Datenbank besteht aus verschiedenen Datenfiles:

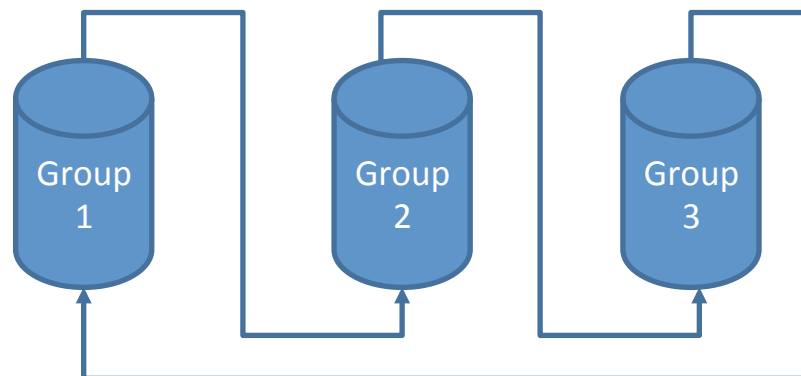


# Datafiles

- Binary Files
- Beinhalten alle Tablespaces und deren Inhalt also alle Daten welche in der Datenbank gespeichert sind
- Sind das Herz einer Datenbank

# Redo Log Files

- Binary Files
- Sämtliche Schreiboperationen werden als Erstes in die Redo Logs geschrieben
- Redo Logs werden zyklisch beschrieben
- Redo's sollten zur Sicherheit immer in Gruppen à 2 Member erstellt werden
- Mindestens 3 Gruppen
- Sind sie voll geschrieben, werden sie archiviert wenn die Datenbank im Archivelog Modus läuft
- Werden zum Restore einer Datenbank gebraucht



# Control File

- Binary Files
- Wird zur Sicherheit immer mindestens 2-fach auf unterschiedlichen Volumes gespiegelt
- Beinhaltet folgende Angaben zur Datenbank:
  - Namen der Datenbank
  - Namen und Pfad aller Datenfiles und Redo Logs
  - Das Characterset der DB
  - Timestamp der DB Erstellung
  - Log Sequenz Nummern
  - Checkpoint Informationen
  - Backup Informationen
  - Etc
- Sehr zentrales und wichtiges File!

- Alte Version: Ascii File (init\${ORACLE\_SID}.ora)
- Neue Version: Binary File (spfile\${ORACLE\_SID}.ora)
- Beinhaltet alle “nicht Default” Datenbankparameter
- Die Ascii Version init.ora ist nicht deprecated, kann verwendet werden, sollte aber nicht weil:
  - Inhalte des spfiles können via SQL Prompt geändert werden, das init.ora muss per Editor bearbeitet werden
  - Oracle schreibt uA dynamische Memory Parameter ins spfile, damit sie bei einem Neustart gleich wieder zur Verfügung stehen
  - Kein Syntaxcheck bis zum nächsten Neustart der DB
  - Einige Optionen setzen ein spfile voraus
- Sollte das spfile verloren gehen, stehen die Parameter vom letzten Start im Alertlog. Änderungen von Parametern werden ebenfalls im Alertlog protokolliert. Es kann neu erstellt werden
- Keine Default Parameter ins spfile schreiben, damit es übersichtlich bleibt. Erleichtert eine Migration auf eine neue Version.





# Trace Files

- Ascii Files
- Werden zum Analysieren von Fehlern benötigt
- Müssen oft an ORACLE gesendet werden, wenn ein Service Request (SR) eröffnet wurde
- Beinhalten Trace Angaben wenn Sessions oder Statements getraced werden
- Liegen in der user\_dump\_dest

```
SQL> show parameter user_dump_dest
```

- Ascii File
- Neue Version im XML Format, kann durch das ORACLE CLI Tool "adrci" verwendet werden
- **ERSTE ANLAUFSTELLE** bei Problemen oder Unklarheiten
- Beinhaltet wichtige ORACLE Fehler
  - ORACLE Fehler beginnen immer mit "ORA-" und stehen am Anfang der Zeile
- Sollte immer im Auge behalten werden
- Liegt in seit ORACLE 11g in der user\_dump\_dest

```
SQL> show parameter user_dump_dest
```

# Password File

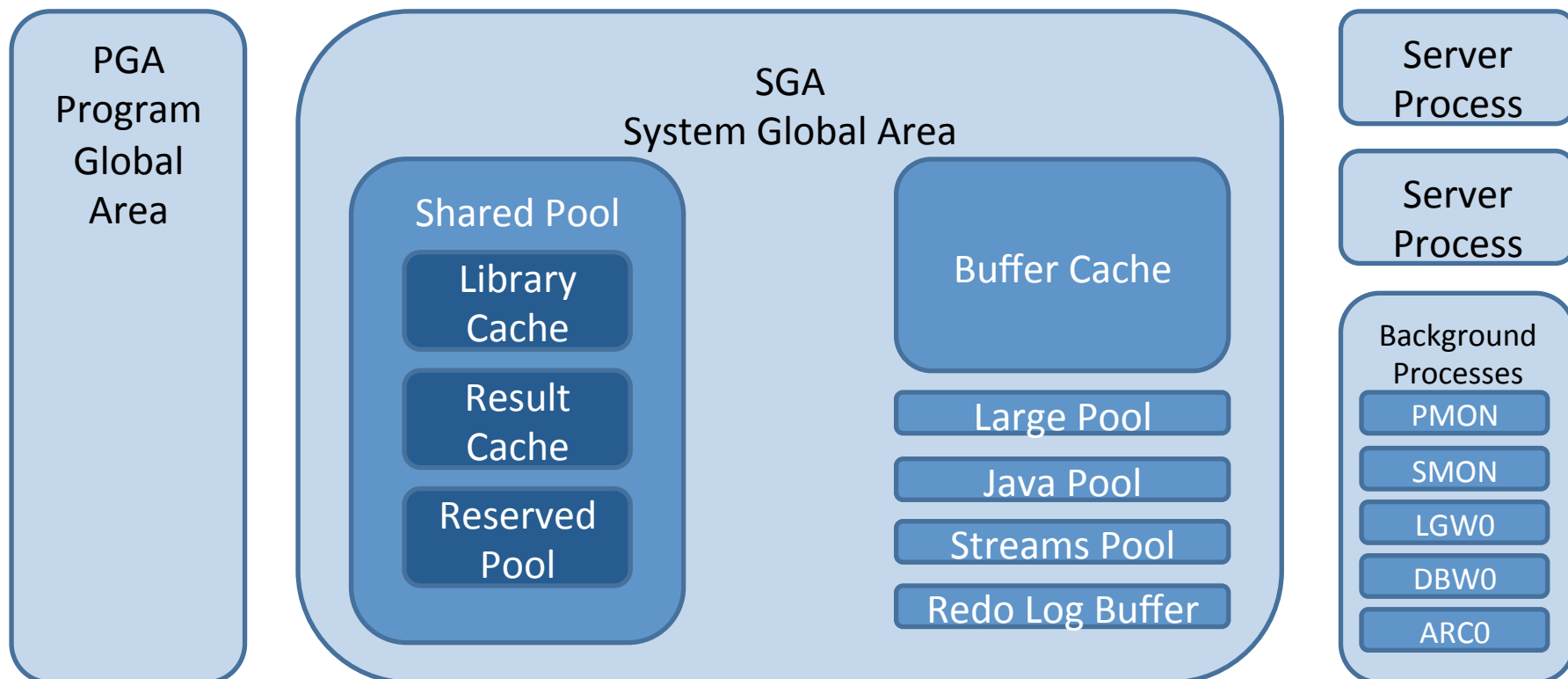
- Binary File
- Beinhaltet die Passworte von hochprivilegierten Systemusern
- Wird für den Remotezugriff mit hochprivilegierten Usern verwendet
- Ist kein mandatory File
- Wird mit dem Utility “orapwd” generiert
- Mit dem Parameter “remote\_login\_passwordfile” wird definiert ob ein Parameterfile existiert
- Ist Voraussetzung für verschiedene Optionen

```
SQL> show parameter remote_login_passwordfile
```

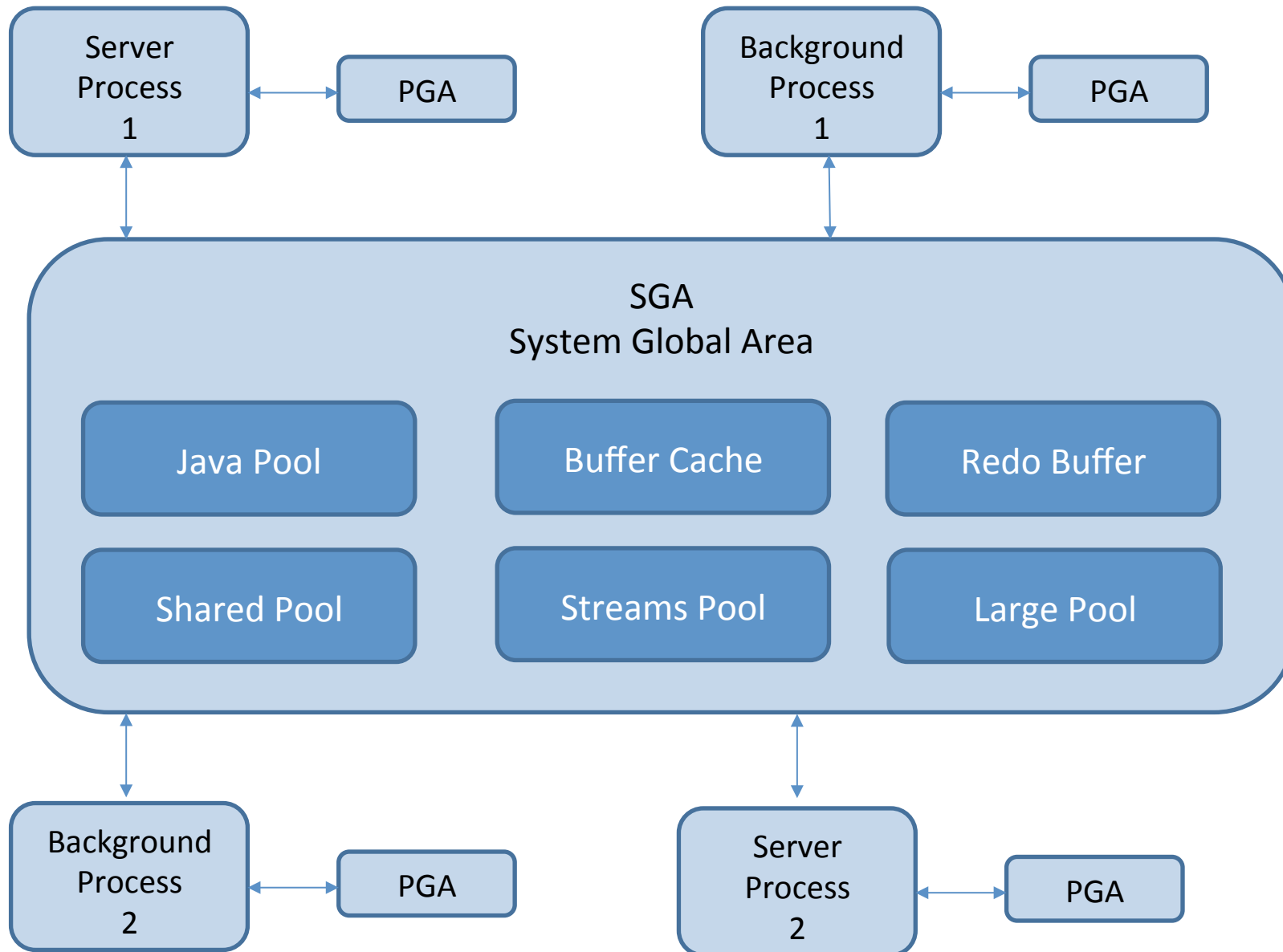
# Agenda

1. Ausgangslage
2. Übersicht ORACLE Architektur
3. Komponenten einer Datenbank
4. Binaries
5. Datenfiles
6. **Memory Bereiche**
7. Dokumentation

- Eine Instanz besteht aus verschiedenen Memorystrukturen
  - SGA - System Global Area
  - PGA - Program Global Area
  - Server Prozesse (User Sessions)
  - Background Prozesse



# ORACLE Memory Strukturen



# SGA – System Global Area

- Die SGA ist ein Memory Bereich, der verschiedene Unterbereiche verwaltet z.B.
  - Shared Pool
    - Beinhaltet einige Pools z.B. den Library Cache:
      - Speichert geparsete SQL – und PL/SQL Statements, damit sie wiederverwendet werden können und uU nicht mehr geparsed werden müssen.

**SHARED\_POOL\_SIZE = n [K|M|G]**

- Large Pool
  - Wird uA für Backup's und Restores mit RMAN (Recovery Manager) verwendet
  - In Shared Server Konfigurationen ist das Session Memory im Large Pool, nicht in der PGA

**LARGE\_POOL\_SIZE = n [K|M|G]**


# SGA – System Global Area

- (Database) Buffer Cache
  - Speichert Kopien der Blöcke, von den Datenfiles in der Default Block Size
  - Grösse einstellbar mit Parameter db\_cache\_size

**DB\_CACHE\_SIZE = n [K|M|G]**

- Andere Block Sizes brauchen eigene Pools (db\_2k\_cache\_size - db\_32k\_cache\_size)
- Weitere Pools zum Pinnen bzw. Schnellen Löschen der Daten sind vorhanden:
  - db\_keep\_cache\_size
  - db\_recycle\_cache\_size



- Die Grösse der Memory Pools der SGA können automatisch verwaltet werden 
  - AMM - Automatic Memory Management
  - Eingeführt mit ORACLE 10g
  - Parameter `sga_target`, `sga_max_size`
  - `sga_max_size` könnte grösser gesetzt werden als `sga_target` damit im laufenden Betrieb das Memory vergrössert werden könnte, das ist jedoch nicht zu empfehlen.
  - Ist `sga_target > 0`, ist AMM eingeschalten
  - Die Werte der dynamischen Pools (z.B. `shared_pool_size`, `db_cache_size`) gelten dann als Minimalwerte

**SGA\_TARGET = n [K|M|G]**

# PGA – Program Global Area

- In der PGA sind die privaten Memorystrukturen jeder Session abgelegt. Z.B.
  - Private SQL Area: Bind Variablen
  - Work Area: Raum für Sort Operationen
- Festlegen der Grösse der PGA mit Parameter `pga_aggregate_target`

```
PGA_AGGREGATE_TARGET = n [K|M|G]
```

- `pga_aggregate_target` ist ein “Target” – Wert. ORACLE versucht nicht mehr Platz zu verwenden wie angegeben, verwendet jedoch manchmal mehr.

```
SQL> select name, value
       2  from v$pgastat
       3  where name = 'over allocation count';
```

NAME	VALUE
-----	-----
over allocation count	1249

- Es laufen viele verschiedene ORACLE Prozesse, welche alle zu genau einer Instanz gehören.
- Unter UNIX sichtbar mit:

```
$ ps -ef |grep ${ORACLE_SID}
```

- Wichtige Prozesse:
  - SMON – System Monitor
  - PMON – Process Monitor
  - LGWR – Log Writer
  - DBW0 – Database Writer
  - CKPT – Checkpoint Prozess
  - ARC0 – Archiver Process

# Agenda

1. Ausgangslage
2. Übersicht ORACLE Architektur
3. Komponenten einer Datenbank
4. Binaries
5. Datenfiles
6. Memory Bereiche
7. **Dokumentation**

- ORACLE ist sehr gut dokumentiert
- Die Dokumentation liegt hier:
  - <http://docs.oracle.com>
- Es lohnt sich bei Problemen erst in der Dokumentation zu schauen, erst dann zu googeln
  - Man lernt nie aus 😊

# Fragen?

