

# Management der Oracle Datenbank 12c

Ralf Durben, ORACLE Deutschland B.V. & Co. KG

Die neue Version 12c der Oracle Datenbank bringt nicht nur viele neue hilfreiche Features, sondern muss auch verwaltet werden. Dazu steht neben den bekannten Tools jetzt mit Enterprise Manager Database Express auch ein neues Mitglied aus der Enterprise-Manager-Familie zur Verfügung.

Enterprise Manager (EM) ist eine Sammlung von Tools zur Verwaltung von Oracle-Produkten. Im Zentrum steht Cloud Control 12c als zentrale Verwaltungskonsole. Der modulare Aufbau wird dazu genutzt, bei neuen Produkten oder Produkt-Versionen eine rasche Verwaltungsunterstützung anzubieten. Das gilt auch und gerade für die neue Database 12c. Seit Februar, und damit schon vor Erscheinen der neuen Datenbank, beinhaltet das Modul zur Verwaltung einer Oracle-Datenbank die Unterstützung von prominenten Features wie Multitenant, Data Redaction, neuem Auditing, neuen Resource-Manager-Plänen und vielem mehr. Damit sind diese neuen Features seit dem ersten Tag der Verfügbarkeit sofort in Cloud Control 12c nutzbar (siehe Abbildung 1).

Die vollständige Unterstützung aller neuen Features der Oracle Datenbank 12c wird mit den folgenden neuen Versionen des Datenbank-Moduls in Cloud Control zeitnah realisiert. Wie für die Datenbank 10g beziehungsweise 11g gibt es auch in 12c eine Einzel-Datenbank-Verwaltungslösung, die als Web-Anwendung genutzt wird. Diese neue Lösung mit dem Namen „EM Database Express“ löst das alte „EM Database Control“ ab.

### EM Database Express

Da bei der Implementierung von EM Database Express auf die in der Datenbank beinhaltete XDB zurückgegriffen wird, ist im Gegensatz zum alten EM Database Control kein OC4J-Container mehr notwendig. Dies verschlankt nicht nur die Installation der Datenbank, sondern macht auch die Web-An-

wendung effizienter und schneller. EM Database Express wird mit jeder Oracle Datenbank 12c installiert und bei Verwendung des Database Configuration Assistant auch konfiguriert. Sollte dies für eine Datenbank mal nicht funktionieren, so muss nur ein Port für die XDB geöffnet werden. EM Database Express gibt schnell einen Überblick über die Datenbank und ermöglicht es, Basis-Verwaltungsaufgaben durchzuführen. Es gibt folgende Sektionen:

- Überblick
- Configuration
- Storage
- Security
- Performance

Die Überblicks-Sektion erscheint mit einem Klick auf den Namen der In-

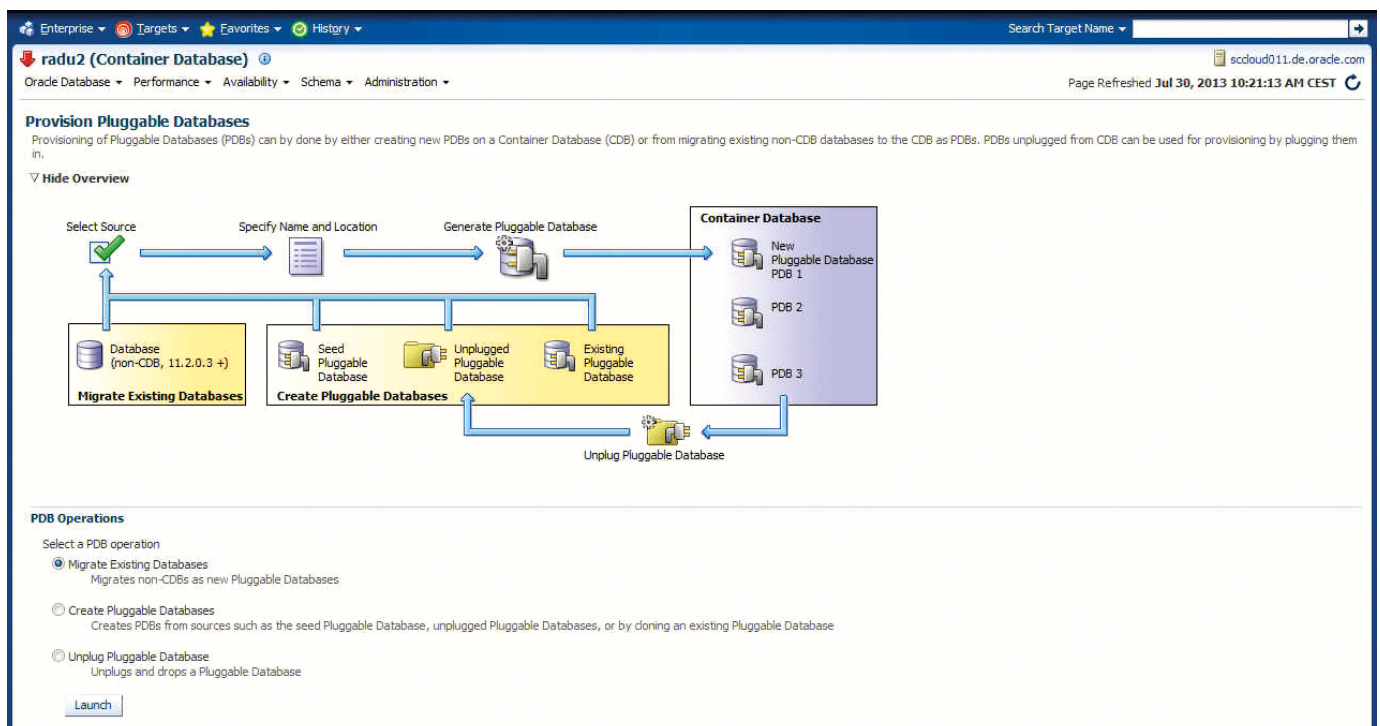


Abbildung 1: Umgang mit Pluggable Databases in Cloud Control

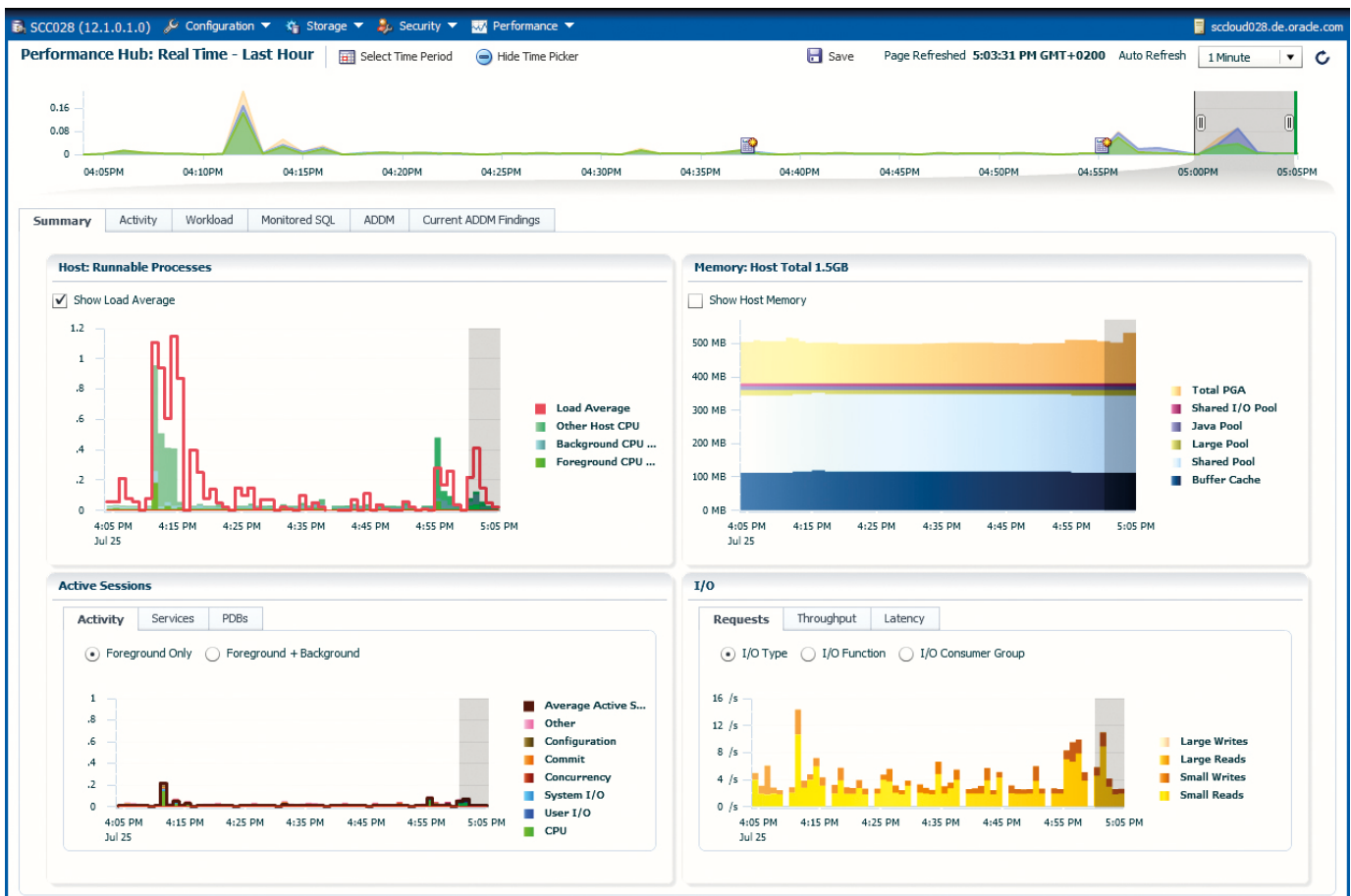


Abbildung 2: Performance Hub in EM Database Express

stanz oben links. Man sieht den Status der Datenbank, einige Angaben zu Performance, Incidents (nur CDB), laufende Jobs und laufende SQL-Kommandos. Im Bereich „Configuration“ kann man je nach Art der Datenbank verschiedene Konfigurationen vornehmen: Instanz-Parameter einstellen, die SGA konfigurieren (nicht bei PDB) und die in der Datenbank genutzten Features einsehen.

Unter „Storage“ lassen sich das Undo Management betreiben, Redo-Log-Einstellungen vornehmen sowie Archive logs, Kontrolldateien und Datendateien verwalten (letztere nicht bei CDBs). Im Bereich „Security“ können Datenbank-Benutzer, -Rollen und -Profile (letztere nicht bei CDBs) eingestellt werden. Der Performance-Bereich, nur nutzbar bei bestehender Lizenz für Diagnostic- beziehungsweise Tuning-Pack, gibt einen Performance-Überblick über die Datenbank; man kann zudem den Tuning Advisor nutzen (siehe Abbildung 2).

Die Aktivierung von EM Database Express ist sehr einfach. Mit der Listener-Control-Utility lässt sich er-

mitteln, welche Ports auf dem Server schon geöffnet sind (siehe Listing 1). Man erkennt jedoch nicht, welcher Port für welche Datenbank verwendet wird. Das wird in der jeweiligen Datenbank abgefragt (siehe Listing 2).

Sollten die Ergebnisse „0“ oder „NULL“ sein, sind die jeweiligen Ports nicht geöffnet. Im obigen Fall gibt es sowohl einen SSL- als auch einen Nicht-SSL-Port. Wenn es einen Port gibt, ist EM Database Express aktiviert und kann im obigen Fall mit „http://hostname:5500/em“ beziehungsweise „https://hostname:5501/em“ aufgerufen werden.

```
lsnrctl status
:
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)
(HOST=scccloud028.de.oracle.com)(PORT=5500))
(Security=
(my_wallet_directory=/opt/oracle/admin/scc028/xdw_wallet))
(Presentation=HTTP)(Session=RAW))
:
```

Listing 1

Man öffnet einen Port (zum Beispiel Port 5506) für den Betrieb mit SSL einfach mit „SQL> exec DBMS\_XDB\_CONFIG.SETHTTPSPORT(5506);“. Zum Deaktivieren von EM Database Express setzt man einfach den Port auf „NULL“.

### Weitere Utilities

Neben Oracle Enterprise Manager gibt es auch die altbekannten Utilities „Data Pump“ und „SQL\*Loader“, für die es auch neue Features gibt. So kann man jetzt für verschlüsselte Exportdateien, die mit einem Passwort ge-

schützt werden sollen, dieses auch interaktiv angeben und nicht wie zuvor nur als Parameter in der Kommandozeile beziehungsweise als Parameter in einer Datei (siehe Listing 3).

Darüber hinaus kann man Views als Tabellen exportieren, wobei nicht wie früher nur die View-Definition exportiert wird, sondern die Definition der View als „SELECT“-Abfrage für einen Daten-Export herangezogen wird. Das Ergebnis ist eine Export-Datei, die bei einem Import die entsprechenden Daten als Tabelle importiert (siehe Listing 4). Mit dem „TRANSFORM“-Parameter lässt sich jetzt auch beim Import eine Komprimierungsmethode wählen (siehe Listing 5).

Gerade bei initialen Imports ist es oft nicht sinnvoll, die dadurch normalerweise anfallenden Redo-Log-Daten generieren zu lassen – vor allem, wenn erst danach eine Backup-Strategie aufgesetzt und die Datenbank produktiv wird. Aus diesem Grund ist jetzt der auch von anderen Kommandos bekannte „NOLOGGING“-Modus für den Import möglich, wobei dieser auf Tabellen oder Indizes eingeschränkt werden kann (siehe Listing 6).

Schon seit langer Zeit gibt es die Möglichkeit, eine Datenbank per Export/Import zu transportieren. Dabei werden per Full-Database-Export alle Nicht-System-Daten in eine Export-Datei ausgelesen. Bei großen Datenbanken wird diese entsprechend umfangreich und der Vorgang dürfte auch seine Zeit benötigen. Dann wird diese große Export-Datei, die auch in kleine Stücke aufgesplittet werden kann, von Server zu Server kopiert. Beim Full-Database-Import werden zunächst eine leere Datenbank mit allen System-Daten erstellt und dann die in der Export-Datei gespeicherten Nicht-System-Daten importiert, zumeist auch ein zeitraubender Vorgang.

Mit der Oracle Datenbank 12c gibt es nun einen neuen Weg für diese Aufgabe: den Full Transportable Export/Import. Dabei werden beim Export keine Daten aus der Quell-Datenbank ausgelesen, sondern nur Metadaten, also die Beschreibung, welche Daten-Dateien zur Datenbank gehören. Die Idee dabei ist, dass nicht eine Export-

```
SQL> select DBMS_XDB_CONFIG.GETHTTPPORT, DBMS_XDB_CONFIG.
GETHTTPSPPORT from dual;
```

```
GETHTTPPORT  GETHTTPSPPORT
-----
           5500             5501
```

Listing 2

```
expdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR schemas=hr
encryption=all encryption_mode=password encryption_pwd_
prompt=y
```

Listing 3

```
expdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR views_as_
tables=[schema].tabelle,...
```

Listing 4

```
impdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR
transform=table_compression_clause:COMPRESS_FOR_OLTP
```

Listing 5

```
impdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR
transform=disable_archive_logging:Y|N[:TABLE|INDEX]
```

Listing 6

Datei mit den Benutzerdaten von Server zu Server kopiert wird, sondern die Daten-Dateien selbst. Dieses Prinzip ist seit geraumer Zeit als „Transportable Tablespace“ bekannt, was jetzt auf die ganze Datenbank ausgeweitet wird. Beim Import werden wieder eine leere Datenbank erzeugt und die kopierten Daten-Dateien dieser neuen Datenbank hinzugefügt. Damit entfällt das zeitraubende Aus- und Einlesen der

Benutzerdaten. Voraussetzung für dieses Verfahren ist, dass alle Tablespace der Quell-Datenbank „READ ONLY“ gesetzt sind.

Auch beim SQL\*Loader gibt es Neues zu vermelden: Für einfache Ladevorgänge mit Comma-Separated-Value-Dateien (CSV) benötigt man jetzt keine Steuerungsdatei mehr. Der SQL\*Loader ist in der Lage, die CSV-Datei automatisch zu analysieren und zu laden. Vor-

```
SQL> create table wdg1 (nu number,text varchar2(100),datum
date);
```

*Tabellendefinition*

```
1,Hans,01-jan-13
```

```
2,Peter,04-feb-13
```

*Datendatei wdg1.txt*

```
sqlldr hr/hr@ORCL1 table=wdg1 data=wdg1.txt
```

*Ladekommando*

Listing 7

```

REM Welche Patches sind installiert?
SQL> select dbms_qopatch.get_opatch_list from dual;
GET_OPATCH_LIST
-----
<patches/>

REM Das Ergebnis hier: Keine

```

Listing 8

```

REM Ist ein spezieller Patch installiert?
SQL> select dbms_qopatch.is_patch_installed(123456) from
dual;
DBMS_QOPATCH.IS_PATCH_INNSTALLLED(123456)

REM Das Ergebnis ist NULL, also ist der Patch nicht
REM installiert

```

Listing 9

aussetzung dafür sind die Verwendung von Kommas als Trennzeichen in der CSV-Datei sowie die Verwendung skalarer Datentypen (CHAR/VARCHAR, Number, Date), die den Datentypen in der Tabelle entsprechen müssen, also keine Konvertierung erfordern. Ein kleines Beispiel macht dieses neue Feature deutlich (siehe Listing 7).

### Neue Management-Features in der Datenbank

Auch ganz ohne Tools gibt es neue Management-Features in der Oracle Datenbank 12c. So können jetzt die Undo-Daten für temporäre Tabellen in eigenen Undo-Segmenten gespeichert werden, die im temporären Tablespace und nicht wie sonst im Undo-Tablespace liegen. Dies hat den großen Vorteil, dass der Umgang mit großen temporären Tabellen nicht mehr unnötigerweise die normalen Undo-Segmente aufbläht, die dann im weiteren Verlauf größer sind als notwendig. Auf diese Weise können auch temporäre Tabellen in einer Active-Data-Guard-Datenbank schreibend verwendet werden, denn sowohl die temporäre Tabelle als auch deren Undo-Daten liegen ja im temporären Tablespace, der im Gegensatz zu den anderen Tablespaces auch in einer sonst „READ ONLY“ geöffneten Active-Data-Guard-Datenbank beschreibbar ist.

Der Administrationsaufwand für dieses Feature ist denkbar gering: Man stellt es einfach mit dem Instanzparameter „temp\_undo\_enabled“ ein. Das Anlegen erfolgt dann automatisch und mit dem Löschen einer temporären Tabelle wird auch deren Undo-Segment automatisch gelöscht: „SQL> alter system|session set temp\_undo\_enabled=true|false;“.

Die Verwendung dieser Undo-Segmente ist in „V\$tempundostat“ einsehbar. Auch bei der Steuerung der SGA gibt es Neues: Der in früheren Versionen eingeführte Parameter „PGA\_AGGREGATE\_TARGET“ gibt eine Zielgröße für die Summe aller PGA-Inhalte an. Diese kann aber in Extrem-Situationen nicht eingehalten werden, wenn die Nutzungsart der Datenbank mehr PGA benötigt als angegeben. Bevor eine Datenbank-Sitzung einen Fehler aufgrund mangelnden PGA-Memorys bekommt, überschreitet die Datenbank das per Instanz-Parameter vorgegebene Ziel. Mit dem neuen Instanz-Parameter „PGA\_AGGREGATE\_LIMIT“ lässt sich dies verhindern und eine stärkere Kontrolle bezüglich des genutzten Hauptspeichers ausüben, im Ernstfall natürlich zu Lasten von Datenbank-Sitzungen, die dann Memory-Fehler bekommen.

Von vielen DBAs wurde schon lange eine Möglichkeit gewünscht, aus SQL

beziehungsweise PL/SQL heraus einen Blick in das Inventory einer Oracle-Installation zu werfen. Aber auch aus Anwendungssicht ist es sicherlich interessant, Prüfungen hinsichtlich vorhandener Patches vorzunehmen und im Ernstfall Anwendungsmodule zu blockieren. Dieser Zugriff war auch früher schon recht aufwändig möglich; das neue PL/SQL-Package „DBMS\_QOPATCH“ zeigt jetzt zum Beispiel sehr einfach an, welche Patches installiert sind (siehe Listing 8). Man kann damit auch gezielt suchen (siehe Listing 9).

Wer das Diagnostics-Pack lizenziert hat, kann sich auf eine neue Variante des Automatischen Datenbank Diagnostic Monitor (ADDM) freuen, die den Namen „Spot ADDM“ trägt. Dabei wird bei speziellen Ereignissen wie zum Beispiel Deadlocks, Hochlast, CPU-Bound etc. automatisch ein Diagnostik-Lauf gestartet, der später jederzeit eingesehen werden kann. Damit ist gewährleistet, dass man Performance-kritische Situationen später genauestens analysieren kann.

Ralf Durben

Ralf.Durben@oracle.com



## Kurz gemeldet: Oracle ZFS Storage ZS3

Die neue Speicher-Lösung unterstützt Anwender dabei, entscheidende Geschäftsprozesse schneller abzuwickeln oder Datenbank-Abfragen zu beschleunigen. Sie erreicht Bestleistungen bei verschiedenen Benchmarks und lässt Konkurrenten wie HP, IBM und NetApp hinter sich. Die ZS3-Serie basiert ausschließlich auf Oracle-Storage-Komponenten und ermöglicht somit eine einfachere Bereitstellung von Datenbanken.