

Kosteneffizient und ohne Risiko: Upgrade auf Oracle Database 12c mit dem Oracle Database Cloud Service

Manuel Hoßfeld, ORACLE Deutschland B.V. & Co. KG

Selbst wenn man gar nicht vorhat, mittel- oder langfristig Datenbanken in der Public Cloud zu betreiben, kann es trotzdem temporär sinnvoll sein, entsprechende Angebote wahrzunehmen. Der Grund dafür liegt in dem teilweise nicht unerheblichen Aufwand, der für ein Datenbank-Upgrade anfällt.

Ein solches Upgrade – zum Beispiel von Oracle 11g auf 12c – kann aufgrund der damit verbundenen Vorbereitungen und Tests zu hohen zeitlichen und finanziellen Aufwänden führen. Dieser Artikel beschreibt, wie sich diese durch die Nutzung des Public-Database-as-a-Service-Angebots von Oracle verringern lassen. Dazu werden zunächst exemplarisch und ohne Anspruch auf Vollständigkeit die notwendigen Schritte betrachtet, um ein Upgrade von einer älteren auf eine aktuelle Datenbank-Version durchzuführen:

- **Hardware (oder VM) beschaffen**
Zu Beginn steht natürlich die Auswahl oder gar Beschaffung der Zielumgebung, auf der die neue Datenbankversion laufen soll. Die prinzipiell bestehende Möglichkeit, direkt ein In-Place-Upgrade durchzuführen, bei der keine neue Umgebung benötigt wird, betrachten wir an dieser Stelle nicht; zum einen, weil die Voraussetzungen dafür in vielen Situationen gar nicht gegeben sind, zum anderen, weil diese Art des Upgrades ein paralleles Testen der neuen Version unmöglich macht.
- **Gegebenenfalls zusätzliche Datenbank-Lizenzen beschaffen/sicherstellen**
Ähnlich wie beim zuvor genannten Punkt der Hardware-Beschaffung ist es nicht selbstverständlich, dass der während der Upgrade- und Testphase benötigte zusätzliche Bedarf an Datenbank-Lizenzen auch tatsächlich abgedeckt ist.

- **Systemadministrator beauftragen zur Vorbereitung/Bereitstellung der Hardware/VM**

In der Regel wird die Bereitstellung der Zielumgebung, also eines physischen oder virtuellen Servers inklusive Betriebssystem, nicht vom DBA, sondern von einer anderen Abteilung vorgenommen. Selbst in gut organisierten Unternehmen mit schlanken Prozessen ist es keine Seltenheit, dass an dieser Schnittstelle gewisse Reibungsverluste – und somit Wartezeiten – entstehen.

- **DBA installiert/konfiguriert neue Version als Test-Datenbank**

Erst jetzt beginnt der eigentliche Vorgang der Installation und/oder Konfiguration der neuen Datenbank. Je nachdem, ob und wie weit eine Vorbereitung der Umgebung aus dem vorangegangenen Schritt bereits erfolgt ist, fällt hier mehr oder weniger Arbeit für den DBA an.

- **Quell-/Ursprungs-Datenbank importieren**

Zwar gibt es durchaus auch Szenarien, in denen es nicht unbedingt nötig ist, Daten aus der alten Datenbank in die soeben frisch installierte Version 12c zu laden, etwa wenn es nur um die prinzipielle Erprobung neuer Features geht. In der Regel wird es aber der Fall sein, dass eine oder mehrere bestehende Anwendungen auch mit der neuen Datenbank laufen sollen und diese daher nicht leer sein kann.

- **Testen („Funktioniert meine Anwendung noch?“ etc.)**

Wie gerade schon angedeutet, wird der Sinn und Zweck der hier beschriebenen Vorgehensweise normalerweise der sein, dass man nun überprüft, ob die mit der Datenbank verwendeten Anwendungen immer noch genauso funktionieren, wie dies unter der Vorversion der Fall war. Diese Phase kann über rein funktionale Tests hinaus auch Tests der Leistung (Last-Tests) umfassen, um sicherzustellen, dass es zu keiner Verschlechterung der Performance kommt.

- **Inbetriebnahme der Anwendung auf der neuen Datenbank**

Erst nach erfolgreichem Abschluss der Tests kann nun die neue Datenbank in die Produktionsphase überführt werden.

Der Oracle Database Cloud Service

Bevor auf die Vorteile von Upgrades in Zusammenhang mit dem Database-as-a-Service-Angebot (DBaaS) der Oracle Cloud eingegangen wird, lohnt sich zum besseren Verständnis zunächst ein allgemeiner Blick auf dessen Eigenschaften. Derzeit stehen verschiedene DBaaS-Ausprägungen in der Oracle Cloud zur Verfügung. Zum einen gibt es den Database Schema Service in drei festen Größenordnungen (5, 20 und 50 GB Storage). Dieser bietet den Nutzern, wie der Name

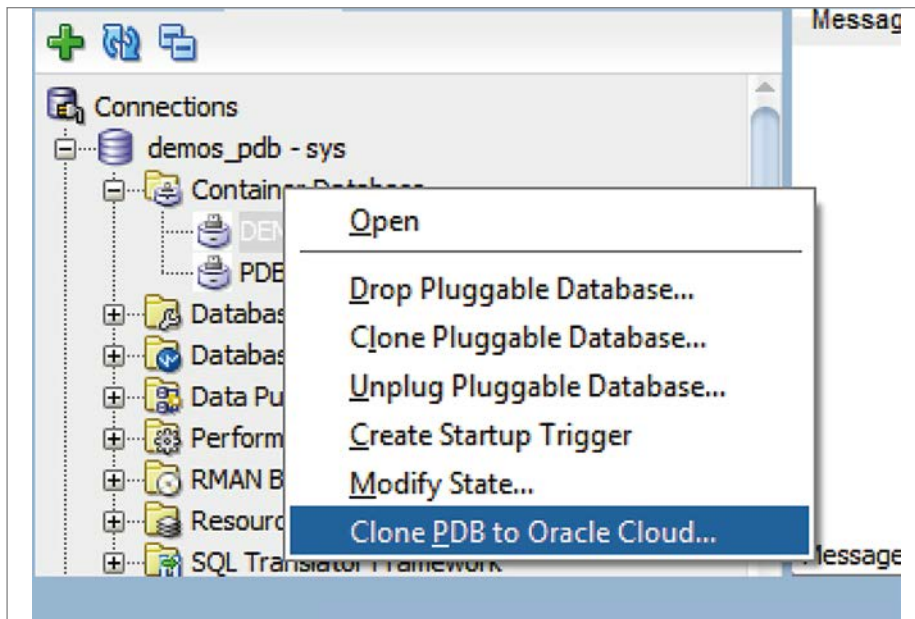


Abbildung 1: Verschieben einer PDB in die Cloud mit SQL Developer

bereits suggeriert, lediglich ein Schema einer Oracle-Datenbank, in dem Entwickler oder Power User mittels Apex kleinere bis mittlere Anwendungen für ihr Unternehmen bauen und betreiben können. Abgesehen von den zusätzlich auch über ein RESTful-API möglichen Datenzugriffen verfügt dieser Dienst jedoch über keinerlei direkten Einfluss auf die Datenbank selbst oder gar das zugrunde liegende Betriebssystem.

Zum anderen gibt es den Database Cloud Service (DBCS), der im Gegensatz zum zuvor genannten Schema-Service eine vollständige Datenbank in der Cloud darstellt, inklusive aller Konfigurations- und Zugriffsmöglichkeiten. Technisch gesehen handelt es sich hierbei um eine VM auf Oracle-Linux-Basis, in der eine Oracle-Datenbank bereitgestellt ist. Neben den bekannten Editionen (wie Standard oder Enterprise Edition) stehen Bundles zur Verfügung, die auch die Nutzung einiger oder aller Datenbank-Optionen ermöglichen, ohne diese separat lizenzieren zu müssen. Ebenso im Preis des jeweiligen Dienstes enthalten ist der Support. Außerdem verfügen alle diese Bundles beziehungsweise Varianten über verschiedene vorgefertigte Werkzeuge, etwa für automatisiertes Backup oder vereinfachtes Patching, die nicht nur für Gelegenheits-DBAs eine willkommene Erleichterung darstellen. Dieser Artikel behandelt abschließlich die zuletzt erwähnte Variante,

also den Database Cloud Service mit eingebautem Cloud Tooling.

Ein Upgrade der Datenbank zuerst in der Cloud durchführen

Wenn man ein Upgrade auf die Version 12c zuerst in der Cloud durchführt, fällt zunächst die schnelle Bereitstellungszeit auf: Betrachtet man die vier ersten Schritte der im vorigen Abschnitt typischen Schritte eines Datenbank-Upgrades, können diese in einer klassischen („on Premise“) Umgebung zwischen einigen Stunden und mehreren Wochen liegen. Nicht so beim Database Cloud Service: Eine fertig konfigurierte und lauffähige Datenbank ist hier über einen grafischen Wizard in wenigen einfachen Schritten erzeugt:

- Version der Datenbank auswählen (11g oder 12c)
- Edition auswählen (SE, EE oder größere Bundles inklusive Optionen wie RAC, Partitioning etc.)
- „Shape Size“ festlegen (letztlich die gewünschte Größe der VMs bezüglich Prozessor-Cores und Memory)
- Sofern gewünscht, automatisiertes Backup auswählen
- SSH-Key hochladen; der Zugang zur Datenbank-VM in der Cloud erfolgt aus Sicherheitsgründen zunächst nur über SSH-Keys

- Zusammenfassung bestätigen und dreißig bis sechzig Minuten warten

Hat man die, wie eben beschrieben, erzeugte Database-Cloud-Service-Instanz vor sich, kann man diese nun sehr gut zum Üben von Upgrade-Szenarien verwenden.

Wie gesagt steht ja neben der aktuellen Version 12c auch noch die Version 11g zur Verfügung. Da sich die Datenbank-Software im Database Cloud Service nicht von der außerhalb der Cloud verwendeten unterscheidet und man vollen Zugriff auf Datenbank und Betriebssystem hat, bestehen hier auch die gleichen Upgrade-Möglichkeiten wie zum Beispiel der DB Upgrade Assistant oder skriptbasierte Ansätze. Hinweis: Mehr zum Thema „12c-Datenbank-Upgrade“ gibt es neben der Oracle-Dokumentation auch in den Aufzeichnungen und Folien der im Herbst 2015 gelaufenen „12c jetzt“-Kampagne (siehe „<http://tinyurl.com/12cjetzt>“).

Nach einem erfolgreich erprobten Upgrade (gegebenenfalls mit vorherigem oder anschließendem Laden von Daten, siehe später im Artikel) können nun sowohl die Grundfunktionen der Datenbank-Anwendung getestet werden als auch etwaige neue Features und Optionen, die man on Premise (noch) gar nicht im Einsatz hat. Möchte man zum Beispiel testen, ob eine Anwendung von In-Memory-Datenbank oder Partitioning profitiert, kann man dies in der Cloud durch vorherige Auswahl eines entsprechenden Bundles, das die gewünschten Optionen enthält, leicht durchführen – und das ganz ohne zusätzliche Investitionen in Hardware oder Lizenzen; lediglich die Kosten für die gewünschten Cloud-Dienste müssen betrachtet werden.

Hinweis: Da es aus Gründen wie Netzwerk-Bandbreite oder -Latenz oft sinnvoll ist, die zu testende Anwendung ebenfalls in der Cloud laufen zu lassen, wird der Database Cloud Service allein nicht die komplette zu testende Umgebung abdecken können. Daher gibt es auch andere flankierende Dienste in der Oracle Cloud, die dabei helfen können: etwa einen Middleware Service, um Java-Programme laufen zu lassen, oder einen Compute Service, um prinzipiell beliebige Software in einer VM laufen zu lassen. Die Behandlung dieser zusätzlichen Dienste würde den Rahmen dieses Artikels jedoch sprengen.

Auch das Risiko eines fehlgeschlagenen Upgrades oder Testlaufs verliert in der

Cloud etwas von seinem Schrecken: Natürlich ist es immer ärgerlich, wenn die Datenbank – zum Beispiel durch Konfigurations- oder Bedienungsfehler nach einem Testlauf – nicht mehr so funktioniert, wie sie sollte. Aber ebenso schnell wie die initiale DBCS-Instanz erzeugt war, kann man diese auch wieder löschen und weitere erzeugen. In Kombination mit dem optionalen Database Cloud Backup Service lässt sich das erneute Aufsetzen einer gefüllten Datenbank beschleunigen, indem man diese einfach aus einer ebenfalls in der Cloud abgelegten Sicherung wiederherstellt.

Daten in eine DBCS-Instanz laden

Wie erwähnt, wird es meist nicht ausbleiben, dass man die Daten der bestehenden/vorherigen Datenbank oder zumindest ein zu Testzwecken geeignetes Subset davon in die DBCS-Instanz laden möchte. Besonders für alle diejenigen, die Bedenken haben, ihre möglicherweise vertraulichen Daten einer Datenbank in der Public Cloud anzuvertrauen, an dieser Stelle ein paar wichtige Hinweise: Sämtliche Tablespace, die man einer DBCS-Instanz hinzufügt, werden über Transparent Data Encryption (TDE) verschlüsselt – und zwar unabhängig von der gewählten Edition beziehungsweise dem gewählten Bundle. Selbst eine Standard Edition pro-

fiziert in der Oracle Cloud also von einem Verschlüsselungsschutz, der ansonsten eine Enterprise Edition inklusive Advanced Security Option erfordern würde.

Außerdem erfolgen Backups, die von RMAN in den Database Backup Cloud Service geschrieben werden, grundsätzlich nur verschlüsselt. Dabei spielt es keine Rolle, ob dieser Backup-Dienst als Quelle eine lokale On-Premise-Datenbank hat oder eine Instanz des Database Cloud Service, die diesen Dienst ebenfalls nutzen kann. Darüber hinaus steht es jedem Datenbank-Anwender natürlich frei, die im Data Masking und Subsetting Pack verwendeten Funktionen zu benutzen, um Daten vor dem Laden und Testen in der Cloud derart zu maskieren, dass sie keine Rückschlüsse mehr auf ihren eigentlichen Inhalt zulassen.

Erfreulicherweise stehen für das Laden von Daten – wie schon bei den Upgrade-Mechanismen – sämtliche Werkzeuge zur Verfügung, die man aus dem On-Premise-Datenbank-Betrieb bereits kennt, zum Beispiel die allgemein gültigen und in der Regel schon seit vielen Versionen bestehenden:

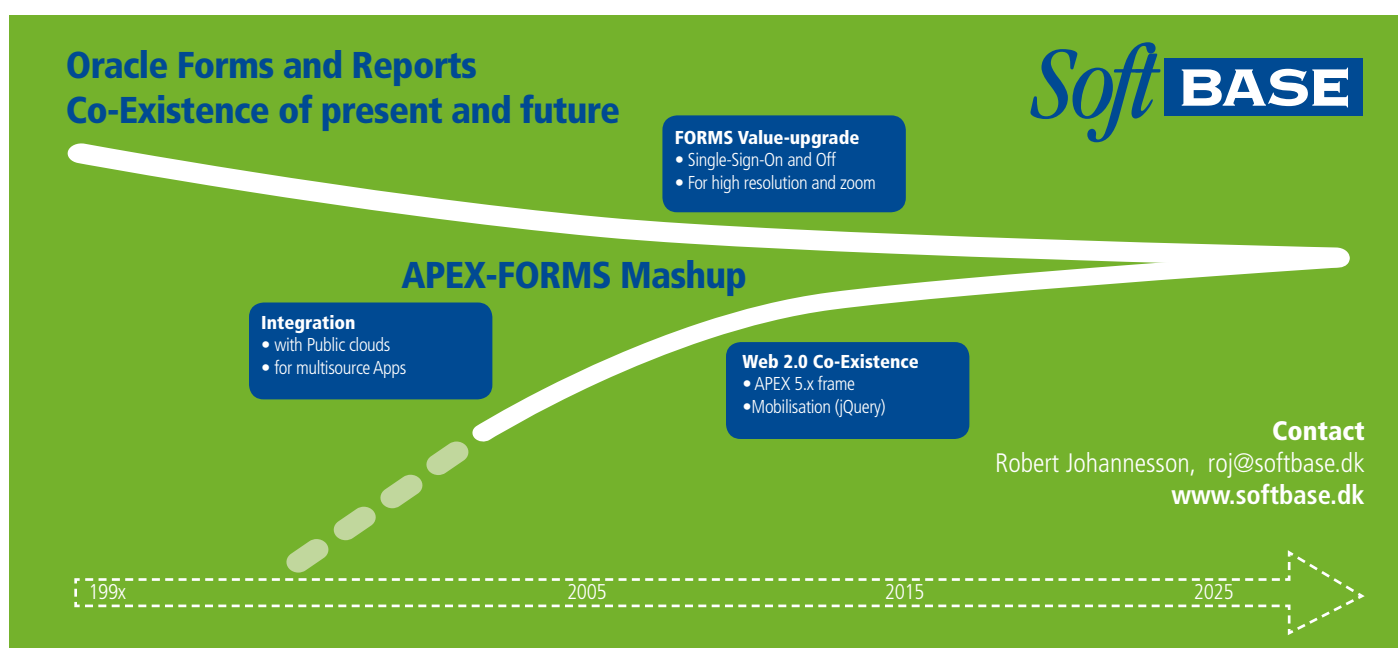
- SQL*Loader
- Import/Export
- Oracle Data Pump Export/Import Utility
- Transportable Tablespaces

Besitzt man also zum Beispiel bereits eine Data-Pump-Exportdatei mit den gewünschten Quelldaten, kann diese nach Upload in

das Filesystem der DBCS-Instanz (etwa über SFTP oder SCP) wie gewohnt in die dort laufende 12c-Datenbank geladen werden.

Zusätzlich gibt es mit der Version 12c als Nebeneffekt der damit verfügbaren Multitenant-Option – also der Möglichkeit, sogenannte „Pluggable Databases“ (PDBs) in einem Datenbank-Container (CDB) laufen zu lassen – noch weitere Datenbewegungs-Szenarien:

- *Plug-in und Upgrade von Pluggable Databases*
Als Sonderfall eines Plug-ins einer PDB in eine CDB kann die Vorgehensweise gewertet werden, bei der zunächst eine 11g-Datenbank in eine klassische, also Non-CDB-12c-Datenbank migriert wird, um dann in eine 12c-CDB eingesteckt zu werden. Als Resultat hat man also nicht nur die Daten bewegt, sondern auch gleich ein Upgrade auf 12c durchgeführt.
- *Remote Klonen von Pluggable Databases*
Unter der Prämisse, dass man bereits eine oder mehrere 12c-CDBs betreibt, kann man auch neue PDBs als Klon einer bestehenden PDB erzeugen, also letztlich deren Daten kopieren.
- *„Lift and Shift“ von Pluggable Databases*
Auch hier ist die Voraussetzung, dass man bereits über eine oder mehrere 12c-CDBs auf der On-Premise-Seite verfügt. Will man eine davon direkt in



die Cloud befördern, spricht man auch von „Lift and Shift“.

Besonders für die letzte der gerade genannten neuen Möglichkeiten im Rahmen von Oracle 12c sind automatisierte Tools verfügbar, die deren Nutzung besonders bequem machen: Zum einen kann der kostenfrei von OTN herunterladbare SQL Developer direkt aus dem DBA Panel heraus per Aufruf der entsprechenden Funktion aus dem Kontextmenü eine bestehende PDB in die Oracle Cloud klonen und anschließend wahlweise auch gleich löschen (siehe Abbildung 1).

Hierbei verwendet SQL Developer seine eingebaute Fähigkeit, SSH-Tunnel zur Verbindung mit entfernten Datenbanken aufzubauen. Neben den Metadaten werden also auch die eigentlichen Datendateien über SSH verschlüsselt an die DBCS-Instanz übertragen. Auch Enterprise Manager 12c als umfassendes Verwaltungswerkzeug nicht nur für Oracle-Datenbanken beherrscht seit der Version 12.1.0.5 das eben erwähnte „Lift and Shift“, wie man in Abbildung 2 sehen kann. Die Voraussetzungen für die Nutzung dieses Features sind etwas höher als beim SQL Developer, da hier zunächst die Kommunikation zwischen dem im eigenen Rechenzentrum laufenden Enterprise Manager und den in der Oracle Cloud laufenden DBCS-Instanzen sichergestellt werden muss. Dies erfolgt über die Einrichtung eines Hybrid Cloud Gateways, das dann auch die Basis für ein übergreifendes Hybrid Cloud Management von Datenbanken innerhalb oder außerhalb der Oracle Cloud darstellt, in dem „Lift & Shift“ nur einer von vielen nützlichen Aspekten ist (siehe „<http://www.oracle.com/technetwork/oem/cloud-mgmt/pr-hybridcloud-ps4-2608209.pdf>“).

Zuletzt soll an dieser Stelle nicht unerwähnt bleiben, dass sämtliche in diesem

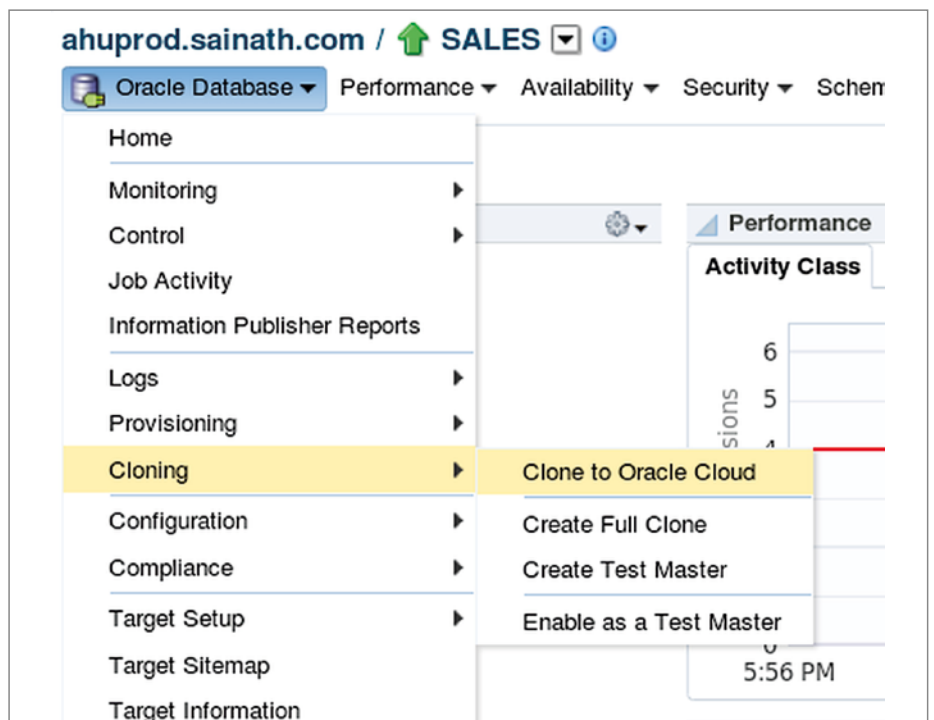


Abbildung 2: Auch Enterprise Manager 12c beherrscht das Verschieben von PDBs

Abschnitt genannten Methoden zum Laden von Daten in die Cloud auch umgekehrt für das Laden von Daten aus der Cloud verwendet werden können.

Weitere Informationen

Auf der Oracle Cloud Homepage gibt es einen Bereich, der über alle Database-as-a-Service-Varianten und deren Merkmale informiert („siehe „<https://cloud.oracle.com/database>““). Dort kann außerdem eine dreißig Tage gültige, kostenfreie Trial-Umgebung beantragt werden. Zusätzlich wird es ab Februar 2016 eine kostenfreie, deutschsprachige Kampagne mit praxis-

nahen Tipps und Anleitungen sowie einem abschließenden Symposium zu den Datenbank-Diensten in der Oracle Cloud geben (siehe „<http://tinyurl.com/oracloudnow>““).



Manuel Hoßfeld
manuel.hossfeld@oracle.com

Die Gewinner der BI Community Survey stehen fest

Die BI-Community hat eine Umfrage organisiert, bei deren Teilnahme drei Amazon-Gutscheine verlost wurden. Nun stehen die Gewinner fest. Mit der Umfrage wollte die neu zusammengestellte BI-Community eruieren, welche Veranstaltungsformate und Themen der Verein im Bereich BI und DWH anbieten sollte. 139 Personen haben

sich daran beteiligt. Das wertvolle Feedback wird in die Planung für die Aktivitäten der Community einfließen.

Die Gewinner der drei Gutscheine, im Wert von 100, 50 und 25 Euro stehen bereits fest: Michael Graening, angestellt bei der Ferdinand Bilstein GmbH, erhält den Hauptgewinn von 100 Euro, Rafaelo Giovannelli, beschäftigt bei der Walser Privatbank AG, hat einen Gutschein im Wert von 50 Euro gewonnen und Sandra Ertl, tätig bei

der Wacker Chemie AG, kann sich über 25 Euro in Form eines Gutscheins freuen. Wir wünschen allen Teilnehmern viel Freude mit dem jeweiligen Preis. Mit der Ernennung der Gewinner, geht die Auswertung der Umfrage in die nächste Runde. Mithilfe der Ergebnisse und des enthaltenen Feedbacks aller Teilnehmer, wird die Business Intelligence Community die kommenden Veranstaltungsformate und Themen bestimmen, die im kommenden Jahr umgesetzt werden.