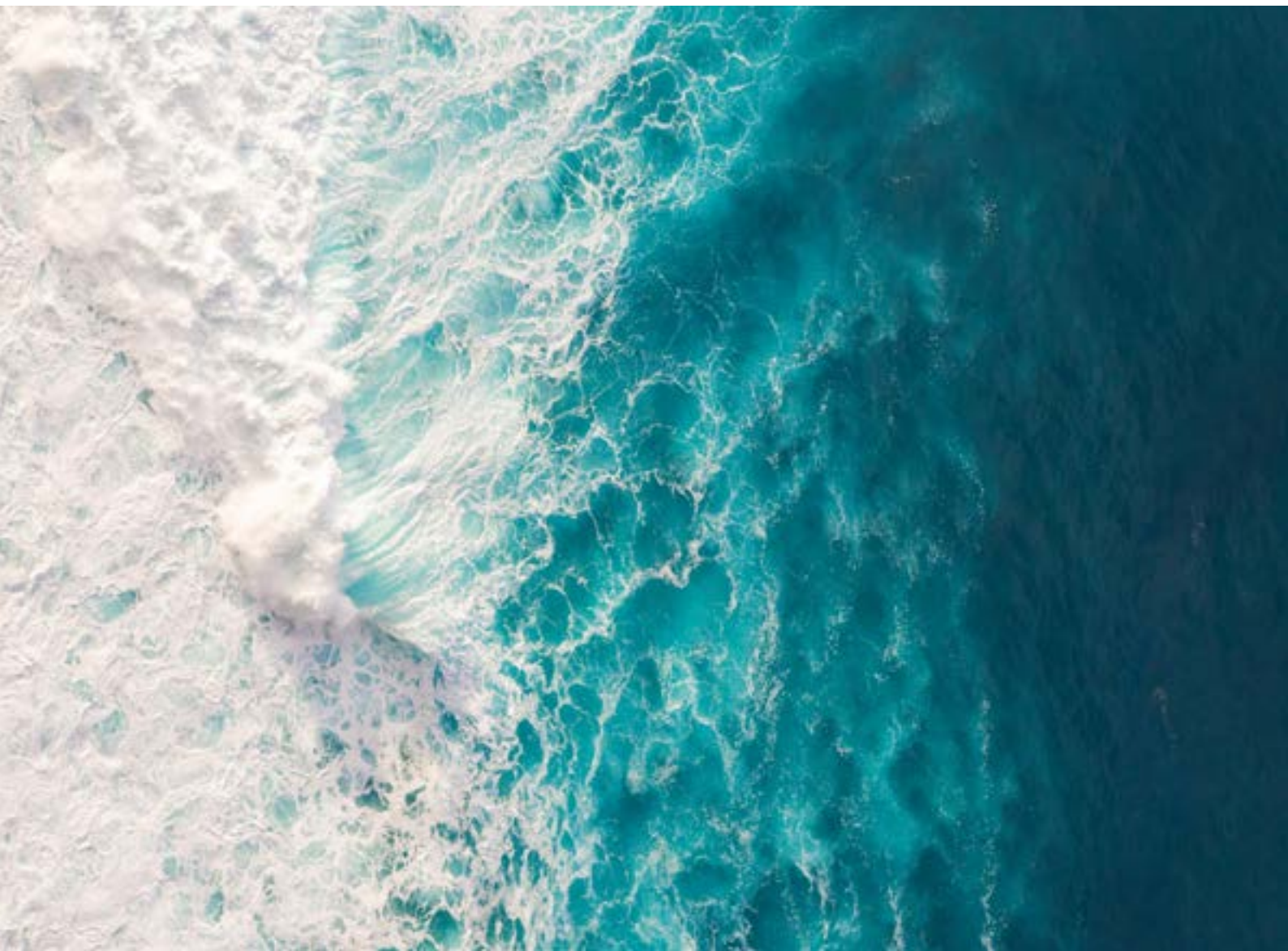




Datenbank-Migration und Modernisierung

So migrieren Sie Ihre Oracle-Datenbank nach PostgreSQL auf AWS



Inhalt

Einführung **4**

Warum migrieren Oracle-Anwender auf eine andere Datenbank? **4**

Finanzielle und rechtliche Risiken 4

Mangelnde Flexibilität 5

Gründe für die Datenbank-Migration 5

Oracle-Cloud-Migration: Welche Alternativen gibt es? **8**

7 Rs: Migrationsstrategien 8

Optimieren und modernisieren: PostgreSQL auf AWS 9

Erfolgreich von Oracle zu PostgreSQL **12**

Das grundsätzliche Vorgehen bei der Migration 12

Schritt für Schritt: Migration von Oracle nach PostgreSQL auf AWS 13

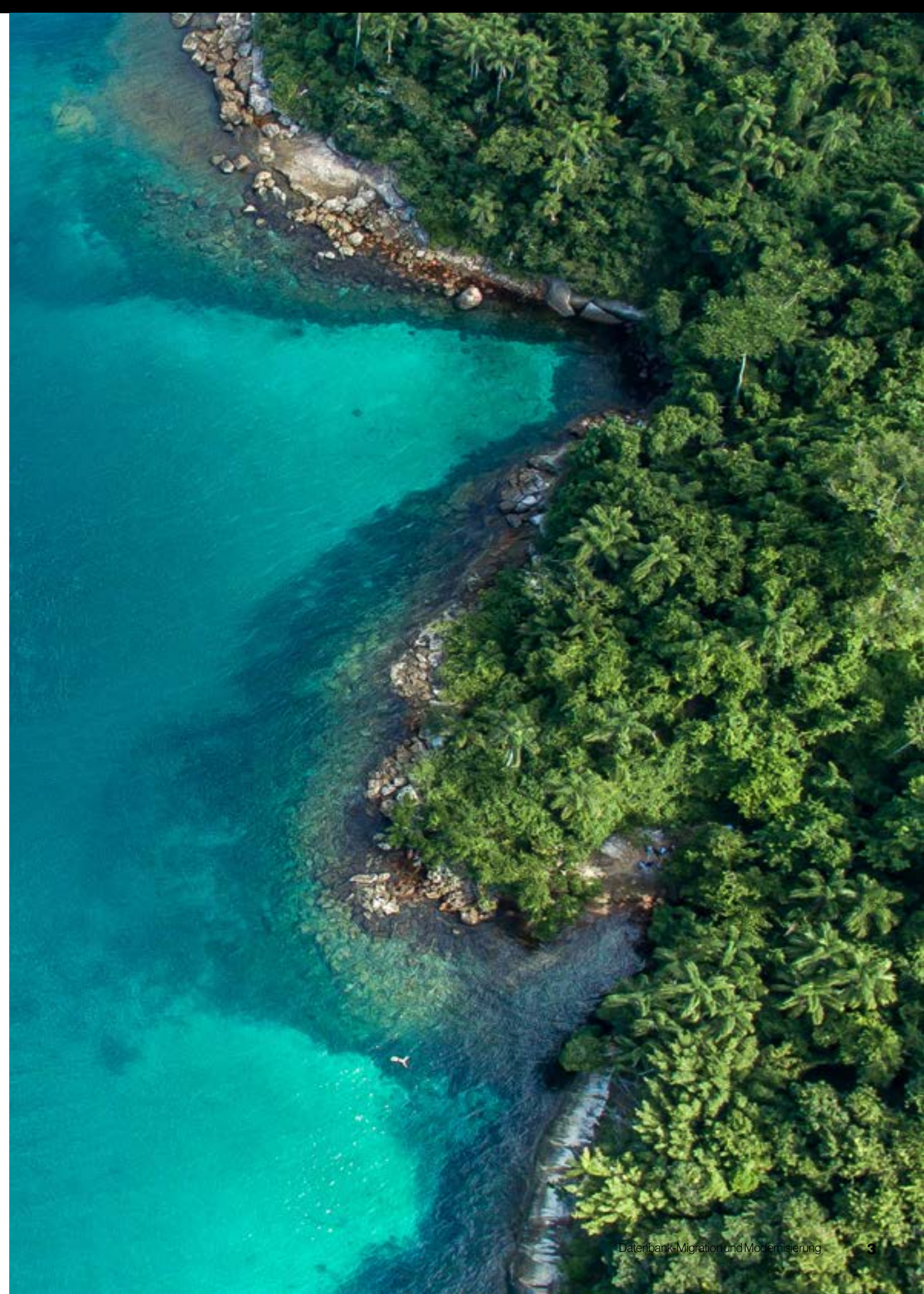
Komplexität und Risiken reduzieren 14

Hand in Hand: Cloud-Migration und Modernisierung **20**

Unterstützung Ihrer Datenbankmigration **21**

So unterstützt AWS 21

So unterstützt SoftwareOne 21



Einleitung

Immer mehr CIOs erwägen die Nutzung von Open-Source-Software, um hohe Lizenzgebühren, finanzielle Risiken durch Audits oder Vendor-Lock-in zu vermeiden – vor allem auch Anwender von Oracle Database. Noch mehr würden gern den Schritt in die Cloud gehen, um ihre IT effizienter und kostengünstiger aufzustellen und ihre Transformation voranzutreiben. Die Cloud-Journey ist daher nicht zuletzt auch ein Anlass, nach Open-Source-Alternativen zu Oracle Ausschau zu halten, zum Beispiel PostgreSQL.

Aber Datenbanken sind unternehmenskritische Systeme für jedes Unternehmen; eine Migration will gut überlegt sein. Das vorliegende Whitepaper zeigt Alternativen für migrationswillige Oracle-Kunden auf, beschreibt das Vorgehen bei der Cloud-Migration nach PostgreSQL auf AWS, benennt kritische Erfolgsfaktoren und erklärt, wie Sie Ihre Datenbankmigration in die Cloud beschleunigen, Risiken reduzieren und Unterstützung erhalten können.

Warum migrieren Oracle-Anwender auf eine andere Datenbank?

Es gibt viele Gründe, warum Oracle-Database-Anwender auf ein anderes Datenbank-Managementsystem (DBMS) umsteigen möchten. Einer der wichtigsten: die Kosten. Zu den hohen Lizenzkosten, bei denen zudem viele Optionen nur gegen Aufpreis erhältlich sind, kommen noch obligatorische Supportkosten hinzu (die sogar für ungenutzte Lizenzen fällig werden). Außerdem ändert Oracle immer wieder seine Lizenzbedingungen, meist zu Ungunsten des Kunden.

Finanzielle und rechtliche Risiken durch Lizenzaudits

Damit sind auch finanzielle und rechtliche Risiken verbunden. Oracle behält sich das Recht für Lizenzaudits vor und überprüft regelmäßig, ob Anwender alle Workloads vertragsgemäß lizenziert haben. Fast immer sind Mehrkosten die Folge – die wenigsten der Geprüften sind vollständig compliant. Oracle verwendet erhebliche Ressourcen und Kreativität darauf, Lizenzverstöße festzustellen, und ist auch nicht zimperlich, wenn es um die rechtliche Durchsetzung vermeintlicher Ansprüche geht. Die auf geistiges Eigentum und Lizenzrecht spezialisierte Anwaltskanzlei Tactical Law Group betreibt einen eigenen [Blog zum Thema Oracle-Audits](#) und beschäftigt sich mit zahlreichen Klagen des Softwareherstellers, darunter [11 Verfahren allein in 2021](#).

Tipp: Die Oracle Advisory Services von SoftwareOne helfen Ihnen, die finanziellen, betrieblichen und rechtlichen Risiken Ihrer Oracle-Verträge zu überblicken, Einsparpotenziale zu nutzen und die Gesamtbetriebskosten (TCO) so niedrig wie möglich zu halten.



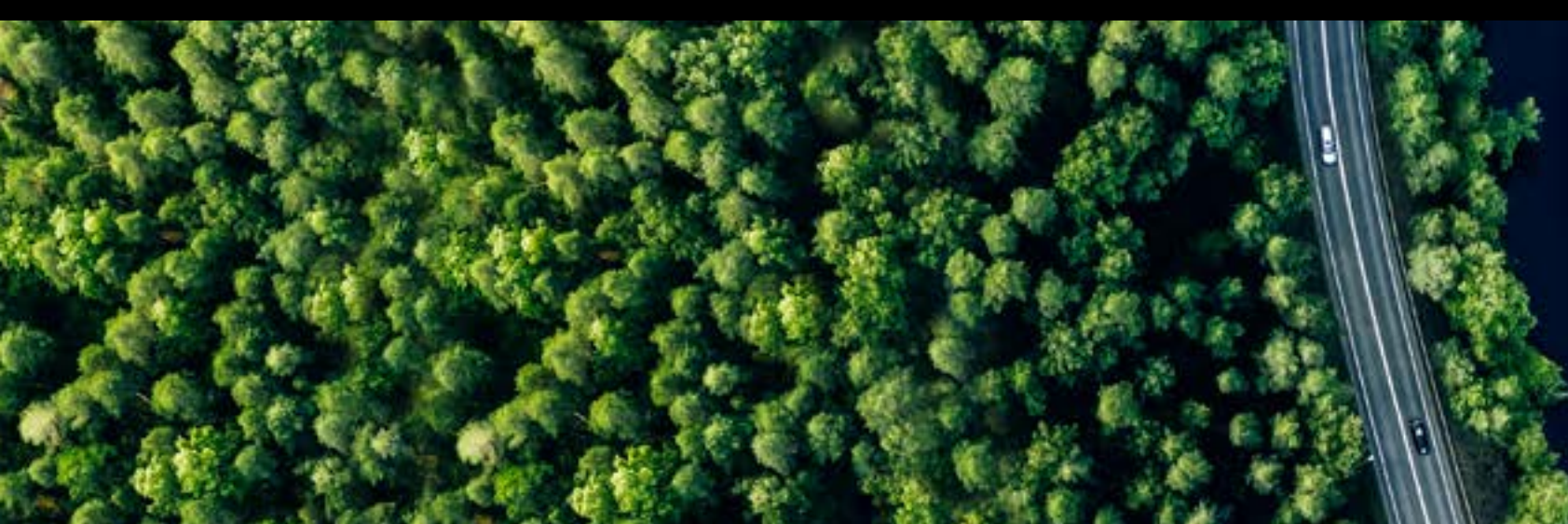
Mangelnde Flexibilität und Vendor Lock-in

Auch die Flexibilität leidet unter der Preispolitik des Anbieters, der die Kunst des „Vendor Lock-in“ perfektioniert hat. Zum Beispiel soll, wer keine „Named-User-Plus“-Lizenz nutzt, Lizenzkosten für alle Prozessoren zahlen, die er theoretisch für Oracle-Workloads nutzen könnte (Soft Partitioning). Bei der Virtualisierungstechnologie von VMware kann das (ab vCenter Server 6.0) im Zweifel die komplette IT-Infrastruktur betreffen, weil Ressourcen dort per „Live Migration“ instanzübergreifend genutzt werden können, egal ob auf dem gerade genutzten Knoten Oracle überhaupt installiert ist. Anwender, die Oracle-Virtualisierung nutzen, haben dieses Problem nicht.

Wer mit seiner Oracle-Datenbank in die Public Cloud umziehen möchte, bekommt die Einschränkungen besonders zu spüren. Denn Oracle möchte Kunden in der eigenen Oracle Cloud Infrastructure halten und gestaltet seine Lizenzbedingungen entsprechend ungünstig für Drittanbieter (mehr Informationen). Zudem drohen unerwartete Zusatzkosten, weil es Anwendern in der Public Cloud, gerade auch in Amazon Web Services (AWS), sehr einfach gemacht wird, zusätzliche Datenbankfunktionen zu aktivieren.

Gründe für die Datenbank-Migration: ein Überblick

Datenbanken sind unternehmenskritische Systeme für jedes Unternehmen, und eine Migration ist für viele CIOs die letzte Option. Niedrigere Total Cost of Ownership (TCO), weniger Komplexität oder andere Vorteile von Open-Source-Datenbanktechnologien reichen oft noch nicht, um diesen Schritt zu motivieren. Aber an einen Anbieter wie Oracle gebunden zu sein, ist nicht nur kostspielig, es kann zudem die Agilität einschränken und damit womöglich die Zukunftssicherheit gefährden.



Critical business events	Migration	Innovation
IaaS to PaaS	Cost Reduction	Digital Transformation Initiative
Outages/Slowdowns Reduction	Optimize ITSM	Services Transformation
Simplification / Reduction (Contract)	Increase Agility	Technical Debt Conversion

Mögliche Gründe für die Datenbank-Migration

Oft kommen weitere Argumente hinzu, die im Einzelfall den Ausschlag geben: Hardwareumstellungen stehen an, Lizenzupgrades werden nötig oder das alte System ist schlicht an seine Grenzen geraten. Ein besonders wichtiger Anlass ist der Umzug in die Cloud – sei es, um einen Managed Service (Platform as a Service, PaaS) zu nutzen (Stichwort Fachkräftemangel, optimiertes IT-Management) oder um die digitale Transformation voranzubringen.

Wenn Sie mit oder ohne Ihre Oracle-Datenbank in die Cloud umziehen wollen, steht Ihnen eine Reihe von Optionen offen. Das nächste Kapitel beleuchtet diese Alternativen am Beispiel von AWS.

Warum AWS?

AWS ist der weltweit führende Public-Cloud-Anbieter vor Azure und Google Cloud (Quelle: [Statista 2/2023](#)), und das besonders deutlich bei Infrastructure as a Service/IaaS (Quelle: [Gartner 2023](#)).

Die Amazon-Tochter bietet schon seit 2004 öffentlich verfügbare IT-Infrastrukturservices an, mehrere Jahre vor Microsoft Azure und Google Cloud Platform. Heute betreibt AWS die umfassendste globale Cloud-Infrastruktur mit derzeit über 450 Points of Presence in 32 aktiven Regionen mit insgesamt 102 Availability Zones ([Stand August 2023](#)). Availability Zones (AZ) sind physisch isolierte Gruppen von Rechenzentren, die über redundante, extrem latenzarme Netzwerke (< 1ms) miteinander verbunden sind.

In seinen weltweiten Rechenzentren bietet AWS mehr als 230 verschiedene AWS-Services an, darunter Datenbanken, Entwicklerwerkzeuge, Serverless-Computing-Funktionen, Analytics, maschinelles Lernen, Anwendungsdienste oder IoT-Werkzeuge. Erweitert wird das AWS-Ökosystem durch Tausende von Softwarelösungen und Services im [AWS Marketplace](#), die bei Cloud-Implementierung, Migration, Betrieb und Innovation unterstützen.

Die AWS-Infrastruktur wurde mit Blick auf die Sicherheitsanforderungen von militärischen Einrichtungen, Banken, Gesundheitseinrichtungen u. a. kritischen Infrastrukturen konzipiert. AWS unterstützt [nach eigenen Angaben](#) 140 verschiedene Sicherheitsstandards und Compliance-Zertifizierungen, darunter C5, PCI-DSS, ISO 27001/27017/27018, ISO 9001, SOC 1/ISAE 3402, SOC 2, SOC 3 oder HIPAA/HITECH. Mit der [Amazon Virtual Private Cloud](#) (Amazon VPC) können AWS-Kunden für ihre kritischen Workloads ein logisch isoliertes virtuelles Netzwerk innerhalb der AWS Cloud nutzen, das viele Vorteile einer privaten Cloud bietet, darunter höhere Sicherheit und vollständige Kontrolle.

Oracle-Cloud-Migration: Welche Wege gibt es?

Oracle-Anwender, die ihre Datenbank in die Cloud umziehen wollen, haben wie bei jeder Cloud-Migration grundsätzlich die Wahl zwischen den bekannten R-Optionen wie Rehost, Replatform, Repurchase oder Refactor.

7 Rs: Migrationsstrategien

Rehost (Lift and Shift): Verschieben der lokalen Oracle-Datenbank ohne Änderungen auf eine virtuelle Maschine (VM) in der Cloud, zum Beispiel eine EC2-Instanz (Amazon Elastic Compute Cloud)

Relocate (Lift and Shift auf Hypervisor-Ebene): Verschieben einer mit VMware virtualisierten Oracle-Datenbank nach VMware Cloud on AWS

Replatform (Lift and Reshape): Verschieben der lokalen Oracle-Datenbank ohne größere Änderungen zu Amazon Relational Database Service (RDS) for Oracle in der AWS-Cloud (PaaS)

Repurchase (Drop and Shop): Erwerb eines neuen Produkts, oft bereits cloudbasiert (Software-as-a-Service, SaaS), und Migration der Daten aus Oracle in die neue Datenbank

Refactor (Neue Architektur): Umstieg von Oracle auf eine andere Architektur, um etwa die Vorteile Cloud-nativer Funktionen zu nutzen, z. B. mit Amazon Aurora. Umfasst auch Neuentwicklungen.

Retain (Beibehalten): Weiternutzung der On-premises-Umgebung für ausgewählte Workloads (zum Beispiel um aufwendige Optimierungen auf einen späteren Zeitpunkt zu verschieben)

Retire (Ausmusterung): Stilllegen bzw. Entfernen nicht mehr benötigter Anwendungen

Optimieren und modernisieren: PostgreSQL auf AWS

Wenn Sie die Gelegenheit nutzen wollen, auf eine neue Datenbank in der AWS-Cloud umzusteigen, also eine **heterogene Datenbankmigration** anstreben, haben Sie insbesondere die Wahl zwischen Repurchase und Refactor:

Repurchase: Umstieg auf eine andere Datenbank, meist eine Open-Source-Datenbank (z. B. PostgreSQL, MySQL, MariaDB)

Refactor: Umstieg auf einen Cloud-nativen AWS-Datenbank-Service: als relationale DB (z. B. Amazon Aurora), NoSQL-Datenbank (Amazon DynamoDB) oder Data Warehouse (Amazon Redshift)

PostgreSQL: Verschiedene AWS-Optionen

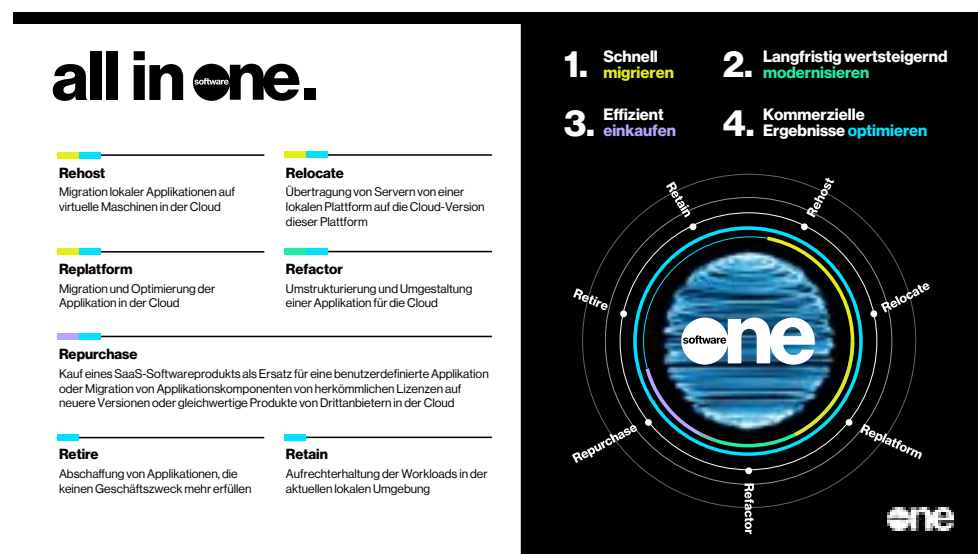
Häufig entscheiden sich Organisationen, von Oracle auf PostgreSQL zu wechseln – aufgrund der relativen technischen Nähe beider Datenbanktechnologien eine sinnvolle Wahl. PostgreSQL, kurz Postgres, ist ein schlankes, modulares und leistungsfähiges Datenbankmanagementsystem, das seit den 1980er Jahren entwickelt wird, seit 1996 als Open Source. Wie Oracle unterstützt es objektrelationale Daten, Trigger sowie Prozeduren und kommt bereits in Tausenden businesskritischen Anwendungen zum Einsatz. Mehr Infos: siehe Kasten „Vorteile von PostgreSQL“.

Auf AWS ist PostgreSQL mit beiden R-Optionen möglich: Sie können PostgreSQL auf einer EC2-Instanz installieren (Repurchase) oder die komplett gemanagten Datenbankservices von AWS nutzen (Refactor): entweder PostgreSQL auf Amazon RDS oder Amazon Aurora PostgreSQL für höhere Ansprüche an Performance, Skalierbarkeit und Verfügbarkeit.

PostgreSQL auf Amazon EC2: Die preiswerteste Option. Sie installieren und verwalten Ihre Datenbank auf einer VM in Amazon EC2, haben Zugriff auf sämtliche Konfigurationsmöglichkeiten inkl. Server-Betriebssystem, müssen sich aber selbst um Updates und Funktionen wie Hochverfügbarkeit, Backups oder Streaming Replication für bessere Skalierbarkeit kümmern.

PostgreSQL auf Amazon RDS: Amazon RDS erleichtert das Einrichten, Betreiben und Skalieren relationaler Datenbanken in der AWS-Cloud (EC2) und nimmt Anwendern die meisten Admin-Aufgaben ab. Mit Funktionen wie automatisierten Backups, Replikation zur Skalierung sowie Multi-AZ-Bereitstellung (synchon über verschiedene Availability Zones) und automatischem Failover für Hochverfügbarkeit und Disaster Recovery ist er für wartungsarme Produktivumgebungen bestens geeignet.

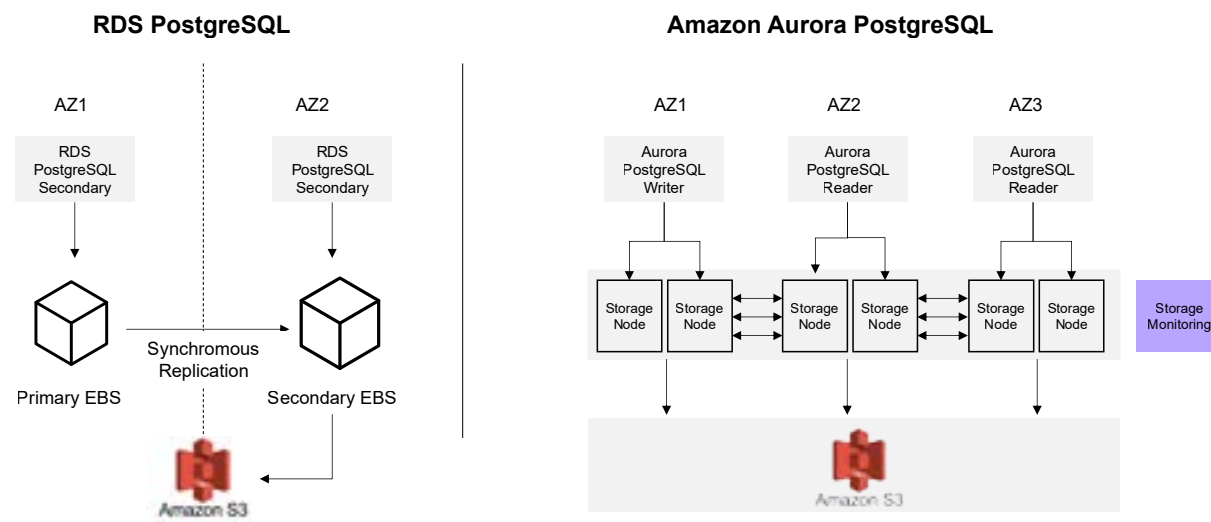
Amazon Aurora PostgreSQL: Amazon Aurora PostgreSQL nutzt ein performanteres Speichersystem: Statt EBS-Volumes (Amazon Elastic Block Storage) für jede DB-Instanz nutzt es dynamischen Speicher, der Replikationen, Failover, Recovery sowie Workloads mit



Migrationsstrategien: die sieben Rs (Quelle: SoftwareOne)

hohem Parallel-Durchsatz erheblich beschleunigt. Es bietet einige zusätzliche Funktionen (u. a. DB-Klonen, Serverless, regionenübergreifende Datenbanken), aber weniger kleine Instanzen. Für Umsteiger von kommerziellen Datenbank-Engines wie Oracle bietet Aurora PostgreSQL It. AWS die gleiche Leistung zu einem niedrigeren Preis.

Tip: Weitere Hinweise zur Auswahl der für Sie geeigneten Service-Option finden Sie in diesem AWS-Beitrag: [Ist Amazon RDS for PostgreSQL oder Amazon Aurora PostgreSQL die bessere Wahl für mich?](#)

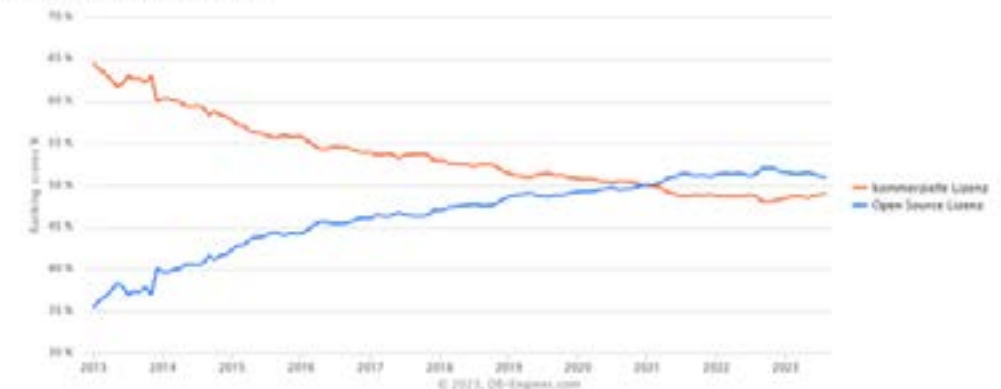


Amazon RDS und Amazon Aurora nutzen unterschiedliche Speicherkonzepte (Quelle: AWS)

Die Vorteile von PostgreSQL

Open Source: Open Source wird für viele Unternehmen immer interessanter (vgl. Abb.). Denn es bedeutet umfangreiche Funktionalität ohne Lizenzgebühren, Audits oder Vendor-Lock-in, kostenlosen Support durch eine große Community und die volle Kontrolle über den Code.

Trend der Popularität



Open-Source-Datenbanken sind inzwischen populärer als kommerzielle DBMS (Quelle: DB Engines)

Für komplexe Anforderungen: Unter den populärsten relationalen DBMS liegt PostgreSQL auf Platz 4 – Tendenz steigend. Es bietet mehr Funktionen als MySQL oder MariaDB und ist anders als diese stets vollständig ACID-kompatibel (Atomicity, Consistency, Isolation, Durability). Beiden ist es bei komplexer Datenmodellierung sowie dank Multiversion Concurrency Control (MVCC) bei Parallelbearbeitung und Datenintegrität überlegen, MySQL auch bei der Skalierbarkeit



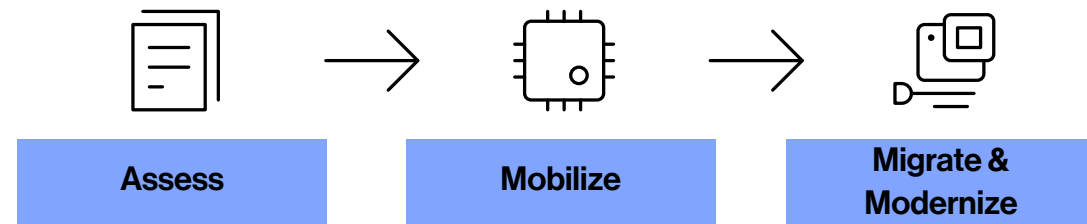
Die Top 4 der relationalen DBMS im Populartätsvergleich (Quelle: DB Engines)

Mehr Informationen: Was ist der Unterschied zwischen MariaDB und PostgreSQL?; Was ist der Unterschied zwischen MySQL und PostgreSQL?

Erfolgreich von Oracle zu PostgreSQL

Das grundsätzliche Vorgehen bei der Migration

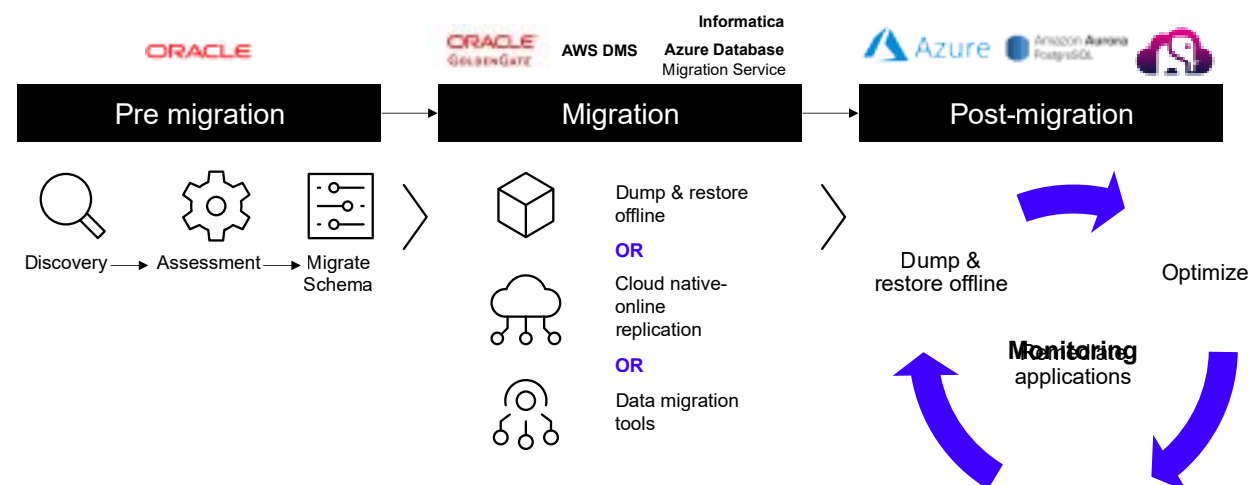
AWS empfiehlt für Cloud-Migrationen ein [dreistufiges Vorgehen](#), das sich bereits vielfach bewährt hat: **1. Bewerten; 2. Mobilisieren; 3. Migrieren und Modernisieren**



Auch die Migration von Oracle Database zu PostgreSQL in der Cloud folgt im Wesentlichen diesem Schema. Das genaue Vorgehen variiert in Abhängigkeit von Ihren lokalen Gegebenheiten (wie Komplexität der Geschäftslogik, Anbindung der Applikationen, Anforderungen an Hochverfügbarkeit, vorhandene Ressourcen).

Für das grundsätzliche Vorgehen gibt es aber klare Empfehlungen:

- 1. Pre-Migration:** Zunächst sind alle Voraussetzungen für die Datenmigration zu schaffen – vom Überblick über Ziele, Anforderungen und die eigene Cloud Readiness über die Festlegung der Migrations- und Modernisierungsstrategie (welches der 7 R?) bis hin zur Planung, Bewertung und dem testweisen Durchspielen der eigentlichen Migrationsprozesse.
- 2. Migration:** Die Datenmigration erfolgt gemäß den konkreten Anforderungen und Quellsystemen (on-premises oder schon in der Cloud) mit geeigneten Werkzeugen zur Automatisierung.
- 3. Post-Migration:** Nach dem Cutover müssen Anwendungen und Datenbank über mehrere Tage sorgfältig überwacht und ggf. optimiert werden.



Drei Phasen der Oracle-Cloud-Migration (Quelle: SoftwareOne)

Schritt für Schritt: Migration von Oracle nach PostgreSQL auf AWS

Ihre Migration von Oracle zu PostgreSQL auf AWS sollte in der Regel die folgenden Schritte umfassen:

1. Bewertung und Planung

Bewerten Sie Ihre Migration Readiness, den zu erwartenden Aufwand und Ressourcenbedarf und schaffen Sie alle notwendigen Voraussetzungen für die Migration.

Verstehen Sie Ihr Oracle-Datenbankschema, Tabellen, Indizes und gespeicherten Prozeduren.

Prüfen Sie, welche der von Ihnen verwendeten Oracle-spezifischen Funktionen Entsprechungen in PostgreSQL haben und welche eine Neuimplementierung erfordern.

Bewerten Sie die erforderliche Datenbankgröße und die Leistungsanforderungen.

2. Schema-Konvertierung

Erstellen Sie ein Schema in PostgreSQL, das Ihrem Oracle-Schema ähnelt.

Konvertieren Sie Datentypen, Funktionen und Prozeduren von der Oracle-Syntax in die PostgreSQL-Syntax, ggf. inkl. Umschreiben von SQL-Abfragen und PL/SQL-Code und Berücksichtigen von Unterschieden in Syntax und Verhalten.

3. Datenmigration

Extrahieren Sie mithilfe geeigneter Werkzeuge die Daten aus Ihrer Oracle-Datenbank.

Transformieren Sie die Daten bei Bedarf, um sie an das Ziel-Schema anzupassen.

Laden Sie die transformierten Daten in PostgreSQL.

4. Konvertierung des Anwendungscodees

Aktualisieren Sie Ihren Anwendungscode, um PostgreSQL-kompatible SQL-Syntax und Funktionen zu verwenden.

Ersetzen Sie Oracle-spezifische Funktionen durch PostgreSQL-Äquivalente.

Berücksichtigen Sie Unterschiede bei Fehlerbehandlung, Transaktionen und Sperrmechanismen bei gleichzeitigem Zugriff.

5. Testen

Testen Sie Ihre Anwendungen gründlich mit PostgreSQL, vor allem im Hinblick auf Datenintegrität, Funktionalität und Leistung.

6. Optimierung

Optimieren Sie schrittweise Datenbankschema, Indizes und Abfragen für die PostgreSQL-Umgebung, um die Testergebnisse zu verbessern.

7. Datenvalidierung und -überprüfung

Vergleichen Sie die Daten in Ziel- und Quelldatenbank, um die Datenkonsistenz sicherzustellen.

Validieren Sie, dass sich Ihre Anwendung mit der neuen Datenbank wie erwartet verhält.

8. Sicherung und Rollback-Plan

Erstellen Sie vor Beginn der abschließenden Migration ein Backup Ihrer Oracle-Datenbank. Bereiten Sie einen Rollback-Plan für den Fall vor, dass kritische Probleme auftreten.

9. Endgültige Migration

Führen Sie eine abschließende Datenmigration zu PostgreSQL durch, um sicherzustellen, dass aktuelle Änderungen in die Zieldatenbank übernommen werden.

10. Umstellung (Cutover)

Planen Sie ein Wartungsfenster, um Ihre Anwendungen auf PostgreSQL umzustellen.

Leiten Sie Ihre Anwendungen auf die neue Datenbank um und überwachen Sie ihr Verhalten.

11. Post-Migration-Aufgaben

Überwachen Sie die Leistung des neuen Setups und optimieren Sie bei Bedarf.

Aktualisieren Sie die Dokumentation und schulen Sie Ihr Team in der neuen Umgebung.

Komplexität und Risiken reduzieren

SoftwareOne hat Tausende von Cloud-Migrationsprojekten begleitet, darunter auch zahlreiche Datenbankmigrationen. Aus Erfahrung wissen wir, wie Sie die Komplexität einer heterogenen Datenbankmigration meistern oder mindern und ihre Risiken erheblich reduzieren können: nämlich durch

- 1. sorgfältige Analyse und Planung im Vorfeld,**
um alle Voraussetzungen für die Migration zu schaffen,
- 2. den Einsatz der richtigen Werkzeuge**
für Analyse, Bewertung und Automatisierung,
- 3. das Befolgen bewährter Best Practices.**

1. Sorgfältige Analyse und Planung im Vorfeld

Die Basis jeder erfolgreichen Migration – auch einzelner Projekte – sind Analyse, Planung und das Schaffen der notwendigen Voraussetzungen, oben „Pre-Migration“ und bei AWS „Bewerten“ und „Mobilisieren“ genannt.

Die Experten von SoftwareOne begleiten Sie bei allen diesen Aufgaben und helfen Ihnen, Risiken zu managen und Ihre Ziele schneller zu erreichen. Wir beraten bei der Entwicklung Ihrer Migrationsstrategie, vermessen Ihre Datenbank- und Applikationslandschaft in Bezug auf Datenbankarchitektur, Datenbankschema oder Anwendungscode, helfen beim vollständigen Verständnis technischer Abhängigkeiten und bestimmen den Grad der zu erwartenden Komplexität und Automatisierbarkeit der Migration nach AWS. [Mehr erfahren](#)

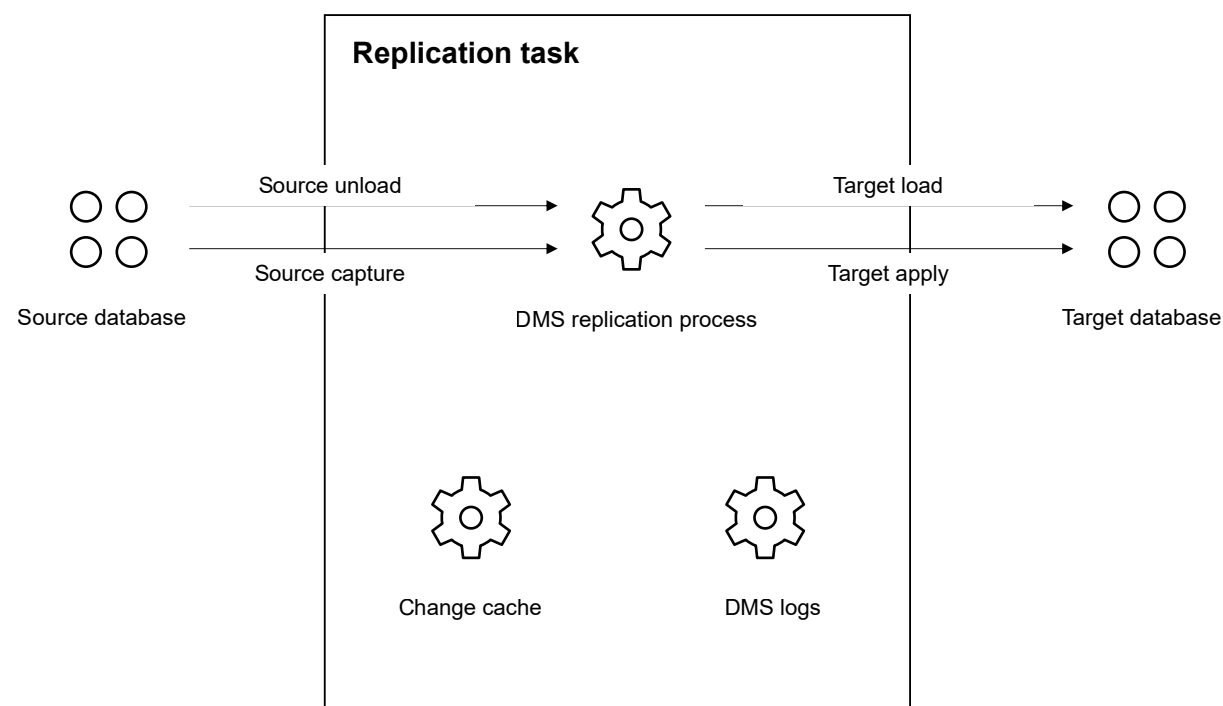
2. Werkzeuge für die Migration

Um die Migration von Oracle-Datenbanken nach AWS bzw. PostgreSQL zu unterstützen, steht Ihnen eine Reihe von Werkzeugen zur Verfügung. Dazu zählen natürlich Tools von Oracle selbst (z. B. Oracle GoldenGate), Werkzeuge von AWS (AWS DMS, AWS SCT) oder Drittanbieter-Tools wie Fivetran/HVR oder Ora2Pg. Für Schemakonvertierung und Datenmigration kommen meist SCT, DMS oder Ora2Pg zum Einsatz – diese drei Tools stellen wir im Folgenden kurz vor.

AWS Database Migration Service (AWS DMS)

AWS DMS ist ein verwalteter Migrations- und Replikationsservice von AWS, mit dem Sie AWS zufolge Datenbank- und Analyse-Workloads schnell, mit minimalen Ausfallzeiten und ohne Datenverlust zu AWS verlagern können. Der Service enthält diverse Komponenten, um verschiedene Migrationsaufgaben zu automatisieren, von der Database Discovery (DMS Fleed Advisor) über die Schemakonvertierung (DMS Schema Conversion) bis hin zur Datenmigration und laufenden Replikation/Synchronisation (Replikationsinstanzen). Die Nutzung der Funktionen ist kostenlos; Sie zahlen nur für die genutzten Instanzen und ggf. Speicher für Logs.

DMS kann mit einer Vielzahl von Quell- und Ziel-Endpoints umgehen, die auf Amazon EC2, Amazon RDS oder on-premises gespeichert sein können – darunter natürlich auch **Oracle** und **PostgreSQL**.

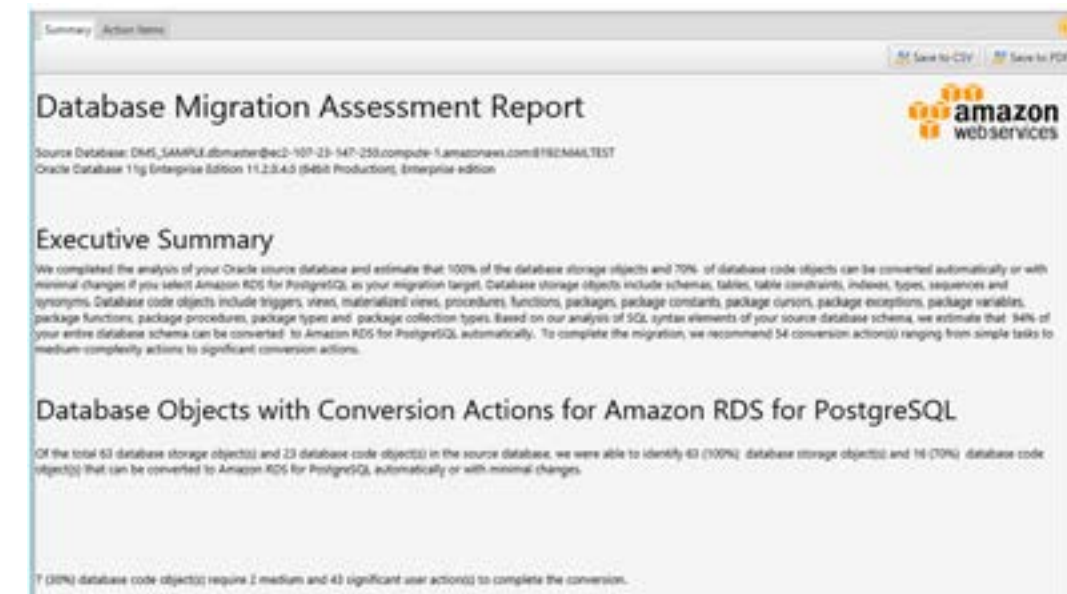


AWS Schema Conversion Tool (AWS SCT)

AWS SCT ist ein Standalone-Werkzeug für die lokale Nutzung unter Microsoft Windows oder Linux (Fedora, Ubuntu). Wie DMS Schema Conversion bewertet und konvertiert AWS SCT automatisch das Schema und viele Codeobjekte der Quelle inkl. Ansichten, gespeicherter Prozeduren und Funktionen und kennzeichnet Objekte, die nicht automatisch konvertiert werden können, für die manuelle Konvertierung. AWS SCT unterstützt außerdem die cloudnative Code-Optimierung und Modernisierung von Anwendungen durch Konvertierung von Oracle-Funktionen und eingebetteten SQL-Anweisungen. Oracle-Quelldatenbanken werden ab Version 10.1 und höher unterstützt.

Bei der Datenmigration wird AWS SCT hauptsächlich für die Migration großer Data-Warehouse-Arbeitslasten nach Amazon Redshift verwendet. Es unterstützt keine laufende Replikation.

Besonders nützlich: SCT kann in der Pre-Migration zur Analyse von Quelldatenbanken und zur Abschätzung des Migrationsaufwandes sowie möglicher Einsparungspotenziale der Cloud-Migration verwendet werden.



AWS SCT hilft bei der Abschätzung des Migrationsaufwandes (Quelle: [AWS](#))

Mehr Informationen:

[Migrating an Oracle Database to PostgreSQL \(mit AWS SCT und DMS\)](#) [Converting Oracle to Amazon RDS for PostgreSQL or Amazon Aurora PostgreSQL \(mit AWS SCT\)](#) [Migrating Oracle databases to Amazon RDS for PostgreSQL with DMS Schema Conversion \(mit DMS\)](#) [Oracle to Aurora PostgreSQL Migration Playbook \(mit AWS SCT und DMS\)](#)

Ora2PG

Ora2Pg, kurz für „Oracle zu PostgreSQL Datenbank Schema Konverter“, ist ein kostenloses und plattformunabhängiges Perl-Werkzeug zur Konvertierung von Oracle-Datenbankstrukturen bzw. Inhalten in ein PostgreSQL-kompatibles Schema. Es verbindet sich mit Ihrer Oracle-Datenbank, scannt sie automatisch, extrahiert die gewünschten Inhalte und generiert SQL-Skripte, die Sie in Ihre PostgreSQL-Datenbank laden können. Ein direkter Import nach PostgreSQL ist ebenfalls möglich.

Neben der Schemakonvertierung beherrscht Ora2Pg auch den Export bzw. die Konvertierung von Oracle-Daten, Tabellen, Ansichten, Funktionen, Triggern, Prozeduren und Packages sowie den Vergleich zwischen Oracle- und PostgreSQL-Datenbanken zu Testzwecken. Ähnlich wie SCT kann es auch zur Bewertung der Migrationskosten und des Schwierigkeitsgrades genutzt werden.

3. Bewährte Best Practices für die Datenbankmigration

Klare Einbettung in eine übergeordnete Cloud-Strategie

Beziehen Sie übergeordnete Ziele in Ihre Migrationsplanung ein – insbesondere in Bezug auf die Modernisierung der Anwendungen, die mit der zu migrierenden Datenbank arbeiten.

Systematisch planen – und den Plan befolgen

- Entwickeln und testen Sie stets Proofs of Concept (PoC)!
- Identifizieren und verstehen Sie funktionale und nicht-funktionale Anforderungen.
- Konzipieren Sie Ihre Lösung frühzeitig, um Ihren IT-Ressourcenbedarf abschätzen zu können.

Komplexität und Datenmengen reduzieren

- Mustern Sie Datenbankobjekte aus, die nicht mehr benötigt werden, inkl. historischer Daten, temporärer und Backup-Tabellen und Daten für überflüssige Funktionen.
- Verschieben Sie Flat Files und LONG-Zeichenketten in LOBs, CLOBs, LONGs usw. nach Amazon S3 oder Amazon DynamoDB (Anwendungen anpassen).

„Fail fast“ – Probleme schnell erkennen

Verwenden Sie Entwicklungs- und Testumgebungen, um Probleme vor der Produktivmigration zu erkennen und zu beheben.

Das richtige Team

Beziehen Sie Anwendungsentwickler und Systemadministratoren frühzeitig in die Migration ein.

Geschäftssimulation

Richten Sie eine für den Test von Anwendungen mit der Datenbank eine Testumgebung ein, die die der Produktionsumgebung und ihrer Nutzungsweise möglichst nahe kommt.

Feinabstimmung & Indizierung

Laden Sie Indizes und Constraints erst nach der Datenmigration

Schulungen

Stellen Sie sicher, dass die Betriebsverantwortlichen rechtzeitig auf der neuen Plattform geschult werden.

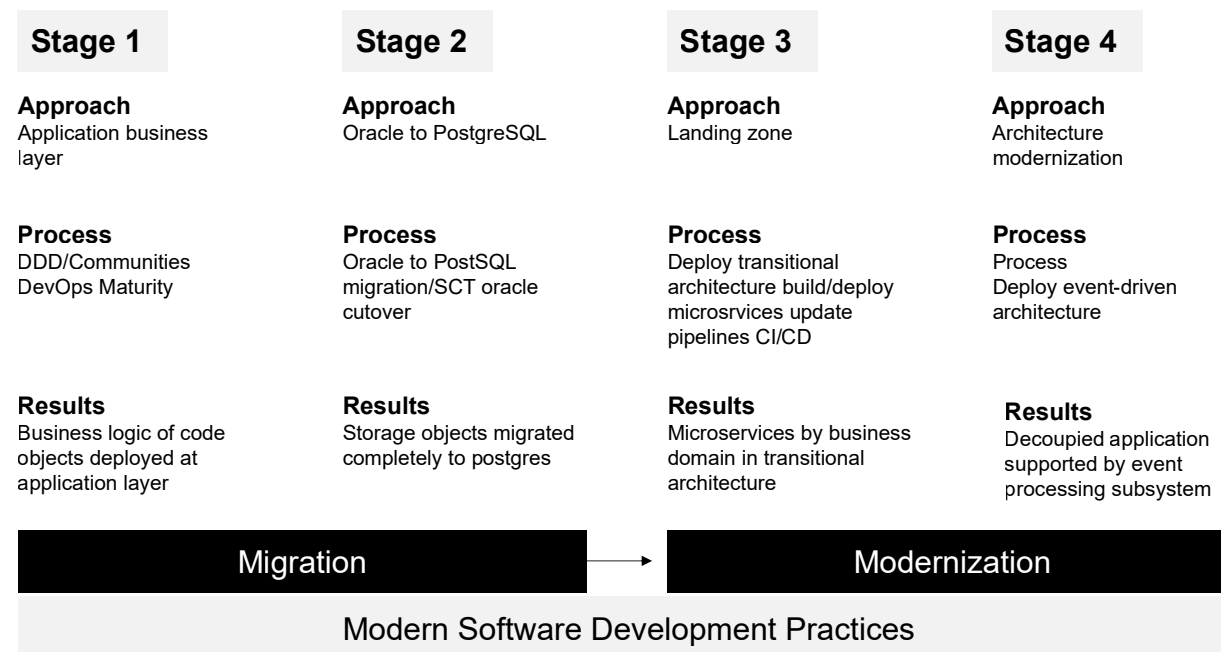


Hand in Hand: Cloud-Migration und Modernisierung

Datenbankmigrationen in die Cloud sind kein Selbstzweck – sie dienen den geschäftlichen Zielen eines Unternehmens. Wer aber den vollen Nutzen aus einer Hyperscaler-Cloudplattform ziehen will, wird nicht dabei stehen bleiben, sondern früher oder später auch seine Anwendungen modernisieren.

Denn auch in der Cloud leiden monolithische Legacy-Anwendungen unter denselben Unzulänglichkeiten wie im Rechenzentrum: Sie sind unflexibel, schlecht skalierbar, machen Entwicklungsprozesse und Wartung komplexer und langsamer und behindern Innovation. Cloud-native Anwendungen mit Microservice-Architektur ermöglichen demgegenüber mehr Produktivität, Agilität und Kostenkontrolle. SoftwareOne-Kunden sparen nach einer Restrukturierung in der Regel bis zu 40 Prozent der Cloud-Kosten ein und erzielen Mehrwert durch neue Cloud-Funktionen.

Deshalb sollten Möglichkeiten und Potenziale der Anwendungsmodernisierung in die Cloud-Strategie eines Unternehmens einfließen, welche wiederum die Richtung für Migrationsentscheidungen vorgibt – zum Beispiel die Entscheidung für eine selbstverwaltete oder von AWS gemanagte Lösung. Zudem können von Anfang an moderne Praktiken der Softwareentwicklung wie Domain-driven Design (DDD) und DevOps angewandt bzw. aufgebaut werden.



Datenbankmigration und Anwendungsmodernisierung gehen Hand in Hand (Quelle:SoftwareOne).

Konkret könnten nach der Migration in PostgreSQL etwa weitere Datenbankschemata eingerichtet werden, sodass beispielsweise ein Schema für einen oder mehrere Microservices reserviert wird. Alternativ wären auch mehrere Datenbanken oder sogar Datenbank-Instanzen möglich – mit Amazon RDS kein Problem. Mehr Infos: [Monolith to Microservices: Refactoring Relational Databases](#).

Tipp: Mit [AWS Prescriptive Guidance](#) bietet AWS bewährte Strategien, Leitfäden und Vorgehensmuster (Patterns), um Ihre Cloud-Migrations-, Modernisierungs- und Optimierungsprojekte zu beschleunigen. Die [Application Modernisation Services](#) von SoftwareOne unterstützen Kunden bei der Evaluierung, Planung, Entwicklung und Bereitstellung modernisierter Anwendungen.

Unterstützung Ihrer Datenbankmigration

AWS und SoftwareOne unterstützen die Migrationsprojekte von SoftwareOne-Kunden durch eine Vielzahl von Services und Programmen.

So unterstützt AWS

AWS Migration Acceleration Program (MAP): [AWS MAP](#) ist ein umfassendes Cloud-Migrationsprogramm, das auf der Erfahrung von AWS aus Tausenden Cloud-Migrationsprojekten basiert. MAP beschleunigt Ihre Modernisierung und Cloud-Migration mithilfe einer ergebnisorientierten Methodik mit dem schon erwähnten dreistufigen Vorgehen Bewerten – Mobilisieren – Migrieren und Modernisieren. [vice-Credits](#).

Tipp: Wenn Sie mit einem Migrationspartner zusammenarbeiten, stehen die Chancen gut, dass AWS Ihr Migrationsprojekt finanziell unterstützt, in der Regel mit AWS-Service-Credits.

So unterstützt SoftwareOne

Enger AWS-Partner: SoftwareOne ist ein weltweit führender Anbieter von End-to-End-Software- und Cloud-Technologie-Services und einer der größten AWS-Servicepartner weltweit. Wir helfen unseren Kunden, Anwendungen und Workloads zu erstellen, zu migrieren, zu managen und zu modernisieren, und wir unterstützen sie bei der Beschaffung und Optimierung von Software und ihrer Bereitstellung auf AWS.

SoftwareOne nimmt an 9 verschiedenen AWS-Partnerprogrammen teil und ist unter anderem AWS Premier Tier Services Partner. Wir verfügen über 500+ AWS-Zertifizierungen und 9 von AWS validierte Kompetenzen, darunter für Migration, DevOps und Security Consulting sowie als Migration and Modernization ISV.



AWS Cloud Services: Wir nutzen AWS MAP, um die Cloud-Einführung zu beschleunigen und Ihren Arbeitsaufwand und die Kosten der Cloud-Einführung zu reduzieren. Gemeinsam mit SoftwareOne optimieren Sie Ihre Cloud-Nutzung und beschleunigen Ihre geschäftliche Transformation, um Ihre Ziele schneller zu erreichen.

Unsere flexiblen End-to-End-Services unterstützen Ihre AWS-Strategie – von der Auswahl optimierter Cloud-Services über die Entwicklung einer maßgeschneiderten Migrations-Roadmap und die Anwendungsmodernisierung bis hin zum laufenden Management Ihrer Umgebung.

3.500 Cloud-Experten: Jede Cloud-Migration ist komplex und erfordert neben technischem Know-how etwa in Bezug auf Datenbanken, Datenmanagement und Security auch unternehmerisches und kaufmännisches Wissen, um die besten Optionen für jeden Kunden zu finden. Bei AWS kommt noch die Vielzahl an verfügbaren Optionen für Instanzen, Services und Tools hinzu, die Wissen und Erfahrung voraussetzt. Unsere Experten unterstützen Sie dabei, diese Komplexität zu bewältigen. Ob Startup oder Großkonzern, SoftwareOne begleitet Sie auf Ihrer AWS Migration – von der Planung über die Migration bis hin zu Betrieb und Optimierung Ihrer Infrastruktur auf AWS.

Nehmen Sie Kontakt zu einem unserer Experten auf.

KONTAKTIEREN SIE UNS

Besuchen Sie uns auf
www.softwareone.com

DE
T. +49 341 2568 000
E. info.de@softwareone.com

AT
T. +43 1 878 10 0
E. info.at@softwareone.com

CH
T. +41 844 44 55 44
E. info.ch@softwareone.com



Copyright © 2023 by SoftwareOne AG, Riedenmatt 4, CH-6370 Stans. Alle Rechte vorbehalten. SoftwareOne ist eine eingetragene Marke der SoftwareOne AG. Alle anderen Marken sind Eigentum der jeweiligen Inhaber. SoftwareOne übernimmt für die Aktualität, Vollständigkeit und Richtigkeit keine Gewähr. © Bildmaterial von: Adobe Stock und Getty Images.