MOVING ORACLE WORKLOADS TO THE CLOUD

5 KEY DECISIONS TO MAKE WHEN MOVING YOUR ORACLE® WORKLOADS TO THE CLOUD

EDB POSTGRES

IN THIS GUIDE

EDB
POSTGRES

5 KEY DECISIONS TO MAKE WHEN MOVING YOUR ORACLE WORKLOADS TO THE CLOUD

If your organization is considering moving your Oracle® workloads to the cloud, it is likely part of a broader initiative to move enterprise workloads to the cloud. This is not uncommon. Many organizations are actively exploring ways to deploy new or upgraded applications to the cloud whenever possible.

Enterprises are already running both development and production databases in the cloud, and many of these are Oracle workloads. If you have your eye on moving your Oracle databases to the cloud, you have a lot of choices to make and options to consider. We've outlined five key decision points to help you navigate your choices and enable your success.

migration might proceed. For highly sensitive or mission critical applications that are operating well in their current environment, a move to the cloud may not be worth the risk.

Expectations around transaction processing must be taken into account. If your organization is coming from a bare metal environment, where the norm is 8,000 transactions per second (TPS) or more in an Oracle environment in your data center, it will not be the same in the cloud.

CHOICE 1: YOUR WORKLOAD

While there is no shortage of applications you might want to move, not every workload is equally suited for cloud deployment. You must consider the operational characteristics of your applications in determining how well and how easily a potential

EDB™
POSTGRES

On-premises, you may have the luxury of a massive, fiber channel-connected box where there is an average of 150,000 input/output operations per second (IOPS). You won't see that kind of high end performance on an individual system in the public cloud. The IOPs might be only half of your organization's required capability, even in an ideal scenario with no overhead.

To properly prioritize workloads, experts recommend segmenting applications into different tiers based on their business importance and fitness for the cloud. In the end, IT must ensure that the organization makes a decision that takes the benefits and trade-offs into full consideration.

CHOICE 2: YOUR DATABASE

Once you select your workloads, you need to choose your database. It's important to realize that database migrations to the cloud should not be treated as a "lift and shift" exercise. Technologies in the cloud do not operate in the same way as comparative technology on-premises. For example, Oracle's database technology was built to run best on bare metal. It is very well tuned for scale-up scenarios, especially on large servers. In addition, Oracle's RAC architecture is geared toward optimizing use of shared disk storage. These factors generally make Oracle databases ill-fitted for the cloud.

While the Oracle® Cloud has introduced a number of proprietary features to cope with this inherent mismatch, many organizations are reluctant to become further entangled with Oracle in their move to the cloud. In fact, cloud migration projects are a great opportunity to explore alternative database approaches, like Postgres.

Because of this, it is not surprising to see such a rapid growth of interest in Postgres as a strong Oracle-alternative. The PostgreSQL database management system gained more popularity in the

DB-Engines.com ranking within the last year than any of the other 341 systems they monitor. This led DB-Engines to name Postgres the DBMS of the Year for 2017.

There is more than one way to deploy Postgres in the cloud, however, and it is important to understand the benefits and limitations of each Postgres option in order to forge a comprehensive migration strategy. A few approaches to consider include:

Option 1: Self-Supported Postgres

Perhaps the most straightforward option is simply to run a copy of the PostgreSQL server on a compute instance in a public cloud. This approach is analogous to running the software in your own data center. You can configure the software as you like, use any extensions your application may require, log into the virtual machine directly, and tune any and all database and operating system parameters for your workload. On the other hand, it doesn't provide the kind of "self-service" experience that a DBaaS platform would provide or the level of enterprise support from a commercial database vendor that you are likely accustomed to receiving from Oracle.

Option 2: Amazon RDS and Aurora

Amazon RDS and Aurora provide a self-service alternative to provisioning Postgres compatible databases in AWS. To simplify deployment of RDS and provide the elasticity of Aurora, Amazon has modified core PostgreSQL code to optimize their use of the AWS cloud infrastructure. Of course, the fact that these are Amazon proprietary versions of PostgreSQL leveraging AWS capabilities means that

EDB POSTGRES™

they are only available on Amazon's cloud. And, since Amazon Postgres DBaaS offerings are forks of PostgreSQL, many common extensions are not available.

These DBaaS options are appealing to developers in particular, offering fast, easy deployments without the need for operational IT support.

This "hands-off" approach does come with some downsides, of course. Amazon DBaaS offers the same vanilla Postgres configurations to everyone. It simply must be this way in order for Amazon to operate AWS economically at their incredible scale. It also creates rigidity and strips away some of the flexibility and extensibility that makes Postgres so capable. DBAs and developers are limited in the configurations and tuning they can perform on the database to optimize the system to meet performance, storage, and availability requirements. You can't even log into the machines where your database servers are running. And, if you can't tune the Postgres engine to optimize for an Oracle migration or take advantage of features that enhance Oracle compatibility, you may be left with a failed migration.

Option 3: EDB Postgres Platform

EDB Postgres™ Advanced Server (EPAS) is one of two databases available as part of the EDB Postgres Platform. EPAS enhances the capabilities of the open source database PostgreSQL, to deliver many enterprise features not available in self-supported PostgreSQL, some of which are shown in the table below.

The EDB Postgres Platform's EPAS also extends PostgreSQL with Oracle-compatible functionality that can be deployed either on-premises or in the public cloud. It also comes with a complete set of tools to manage the database, deliver high availability, manage backups, etc. that will be familiar to Oracle DBAs. EDB Postgres provides compatibility with native PL/SQL, the programming language that runs on Oracle® database products, and includes automated migration tools to help save time and minimize uncertainty and risks in the migration process.

	Self-Supported PostgreSQL	EDB Postgres
Password Management	Not available	EDB Password Profiles
Authorization	PostgreSQL RLS	EDB Virtual Private Database
Auditing	Limited audit capabilities	EDB Postgres Advanced Server (EPAS) Audit with DML auditing for INSERT, UPDATE, DELETE, TRUNCATE by user and database, syslog integration, etc. Manage audit logs separately from the server logs.
SQL Injection Attacks	Not available	EDB SQL/Protect
Encryption at Rest	Do-It-Yourself	Proven full-disk encryption procedure Extension of pgCrypto to support secure key management
24/7 Support	Do-It-Yourself	Enterprise level SLA support with direct access to Postgres community leaders
HA/DR	Multiple open source tools	EDB Postgres Management Tool Suite: EDB Postgres Failover Manager EDB Postgres Backup and Recovery EDB Postgres Enterprise Manager
Data Redaction	Do-It-Yourself	Custom Data Views EPAS 11: Built-in data redaction
Secure Configuration Best Practices	Do-It-Yourself	EDB Postgres Advanced Server Secure Technology Implementation Guidelines (http://iase.disa.mil)

EDB POSTGRES™

EDB Postgres Advanced Server can be provisioned a number of ways: 1) manually, as with the self-supported PostgreSQL, 2) through the EDB Postgres Ark DBaaS platform for a self-service experience similar to RDS, or 3) through the EDB Managed DBaaS Service, a fully managed, "white glove" DBaaS service for Postgres delivered in the Amazon public cloud.

If you are currently considering moving your Oracle workloads to the cloud, Amazon RDS and Aurora may have your mindshare. However, the EDB Managed DBaaS option allows you to experience the same ease of deployment and elasticity as with Amazon, but without the trade-offs in performance tuning, configurability, customization, and control. Finally, the EDB Postgres Platform is supported by the EDB team of Postgres experts 24x7 and backed up by EDB's professional development organization, staffed with many PostgreSQL community members.

Together, these capabilities make migrating to EDB Postgres substantially less risky and time consuming for existing Oracle users whether migrating to on-premises, cloud, or DBaaS environments.

In the end, it is important to understand the benefits and limitations of each Postgres option in order to forge a comprehensive Postgres strategy that best meets all your business goals and objectives.

CHOICE 3: YOUR CLOUD

Certainly, Oracle has a cloud, but even though Oracle does a good job of building a cloud designed to run Oracle® databases well, there are other cloud platforms to consider. AWS, Microsoft Azure, and Google are the leaders, and each has its strengths. Comparing the options, here is how they stack up on page 7.

Learn more about EDB Postgres at www.edbpostgres.com

EDB™
POSTGRES

CHOOSE YOUR CLOUD

Option 1: Oracle Cloud

Since this guide is about moving Oracle workloads to the cloud, it is natural to consider the Oracle Cloud as an option. As mentioned above, it is well optimized for running Oracle database software. Of course, the database is only one piece of the overall application environment to be moved to the cloud and typically not the piece that is leading the decision for that move. The availability of application development tools, regions in which the cloud operates, pricing and any number of other considerations usually come ahead of optimizing database performance. In addition, if you are interested in reducing your dependence on Oracle as a vendor in your migration to the cloud, moving from on-premises Oracle to Oracle Cloud will only increase your commitment.

Option 2: AWS

AWS offers a portfolio of services for almost anything you might want to use on a cloud, and has a strong culture of creating APIs for all of it. If you want to script the cloud, orchestrate, and connect services, AWS makes that more accessible. They are rapidly innovating. Of course, many of these services are only available on AWS so you need to be careful in choosing which APIs you adopt if you want to avoid getting locked in.

Option 3: Microsoft Azure

A strong alternative to AWS, Microsoft Azure might be a more ideal fit for organizations already built on a .NET ecosystem. They have a large network of partners who can help you architect and build solutions. Microsoft is also the most experienced of the cloud vendors when it comes to working with businesses. They have a long history of offering quality of service, terms and conditions, and SLAs that can meet the expectations of SMBs and larger enterprises consistently.

Option 4: Google

Google's infrastructure offers massive scale and responsiveness, which they have created for their own purposes to enable search and support video. When operating in Google's cloud, you'll benefit from their investment from a price-to-performance standpoint. They may over-provision you beyond what you pay for. They do have fewer services, but the services they offer are innovative, such as AutoML, TPU, and Video Intelligence for machine learning, and in the data space, Google Genomics, Spanner, and Bigtable. Also, unlike Amazon and Microsoft, they have a long history of contributing to open source. Because of this, many of their services can be replicated on other environments using the projects they've created, like Kubernetes, Google Cloud Spanner, and CockroachDB, reducing the risk of lock-in.

EDB™ POSTGRES

CHOICE 4: YOUR TOOLS

Once you've decided where you're going, it's time to figure out how to get there. Migrating a database application to a new platform can be challenging and it takes significant forethought. This is especially true when moving to a very different environment like the cloud. Because of this, it can be useful to approach it in a number of phases to make the task manageable. Luckily, there are various tools available to support your efforts in each of these steps.

Migrate Schema

When migrating between different types of databases, you may need to change the schema in some way to accommodate the new database. In moving between different relational databases, this is not generally a complex process, since most of the data types will be common between them and many that are not will have simple mappings. There may be exceptions, such as the way geospatial or GIS data is represented is typically specific to a particular relational database. In some cases, certain database vendors simplify this task even further by providing a data dictionary, which is compatible with another leading vendor, (e.g. MariaDB and MySQL or EnterpriseDB with Oracle.) Of course, when you move out of the relational world, for example to a NoSQL system, this mapping can become significantly more challenging as the fundamental data models can be very different and may require a full application rewrite.

A popular tool currently for helping migrate a database schema from one relational database to another is the AWS Schema Conversion Tool (SCT). AWS SCT provides a project-based user interface to automatically convert the database schema of your source database into a format compatible with a target Amazon RDS instance. If schema from your

source database can't be converted automatically, AWS SCT provides guidance on how you can create equivalent schema in your target Amazon RDS database.

Of course there are limitations in the schemas that AWS SCT can handle. Oracle has a very rich set of capabilities that go beyond the standard SQL supported by PostgreSQL. In fact, if you're only using standard SQL, you may not even need a full blown schema migration tool. A simple text editor would likely be sufficient to replace any Oracle-specific types with PostgreSQL equivalents.

Beyond AWS SCT, you might look to Microsoft's Database Migration Service, but SQL Azure is the target of their migrations. Similar to Oracle, Microsoft seeks to keep you in their ecosystem.

Migrate Data

Once the schema has been mapped, data can be transformed and transported into the new data store. The actual transport of the data can also be quite challenging, especially in cases where there is a large volume of data to be moved and where the data needs to be constantly available with minimal downtime during the move. Automated migration tools can be a huge timesaver here and heterogeneous replication can help reduce or eliminate downtime when cutting over to the new database.

In this area, there are a lot of potential tools to choose from, including a few offerings from AWS. Like SCT, AWS offers a data migration tool called

EDB POSTGRES™

Amazon Database Migration Service (DMS). This provides a way of streaming data into your AWS-based data stores such as RDS, DynamoDB, Aurora, and RedShift. If your data set is too big for a real time migration, Amazon offers off-line tools like Snowball for petabyte-scale data sets and Snowmobile for even more extreme data sets.

Outside of the AWS ecosystem, you could look to some of the same tools that you may be used to using on-premises. There are real time replication tools like HVR's Data Replication, Attunity Replicate, or even Oracle GoldenGate. EnterpriseDB offers EDB Replication Server and the Migration ToolKit (MTK) for this task as well.

For extremely large data sets that require batch mode operations, traditional ETL comes into play. There are a lot of options from the long time market leaders, Informatica to the open source leader, Talend and lots of options in between from too many vendors to try to list here.

Migrate Application Logic

This is a step that many overlook when thinking about migrating a database. This may not be difficult if the migration is between relational databases and the application uses only standard SQL interfaces to access and manipulate the data. Moving from Oracle to a NoSQL database, on the other hand, likely means a full rewrite of the code that interacts with the database. Even in the case of a move from Oracle® to another relational database, however, it can be challenging if the application was built to take advantage of non-standard proprietary features in PL/SQL.

Unfortunately, this is probably the area that offers the least help to people migrating databases while at the same time having the potential to have the biggest impact on the success of the overall

migration project. The AWS SCT provides a bit of help in this area by recommending changes to SQL statements, but can't help much in translating Oracle stored procedures, implementing specific Oracle features or converting the use of Oracle specific connectors like OCL or PRO*C.

EnterpriseDB can provide some relief in this area with its database compatibility for Oracle including PL/SQL execution, OCI and Pro*C compatible connectors, and numerous other features. The EDB Postgres Platform provides compatibility with native PL/SQL, the programming language that runs in Oracle database products, and includes automated migration tools to help save time and minimize the risks rewriting database code that may have taken thousands of man-hours to perfect and has been running without error for years.

The EDB Postgres Migration Tool Kit migrates Oracle, SQL Server and MySQL data to PostgreSQL providing online/offline schema/data migration with flexible customization and fast parallel data loading.

Test, Test, Test

Once you've migrated the schema, data, and application, you'll want to run a thorough set of tests to ensure that the application behaves in the same way it did before the move. When doing this, don't forget the operational characteristics of the application like performance, availability, scalability, network speeds, security etc. As described above, the cloud operates in a very different way than machines on-premises. This leads to very different operational characteristics. Any migration should take these changes into account in determining the cloud environment for deploying the application to ensure that it meets the expectations of the existing users.

EDB POSTGRES™

CHOICE 5: YOUR PARTNER

Moving your Oracle workloads to the cloud is an achievable goal and an experienced partner can be invaluable in getting there. It can be complex, but with the right approach, you'll mitigate risk, modernize your architecture, and be better prepared for the future.

EDB has helped many of our customers achieve this goal, and our knowledge on this subject has been earned through real-world experience inside diverse customer environments. We can make your transition from Oracle to Postgres less stressful and more successful at every stage of the journey.

We'll gauge your readiness: We'll start by identifying the applications and databases that you seek to move to the cloud. Then, we will conduct a production readiness assessment and Architectural Health Check. With a deep understanding of how you currently scale, ensure security, and manage your databases and applications, we can make sound recommendations about which workloads are best suited for the cloud, or how you'll need to adapt them for the cloud.

We'll help you make a plan: We'll rank your workloads and help you build a plan to stage your migrations. By selecting low-risk, high-value workloads first, we can accelerate your move to the cloud and earn you some early wins.

We can tune your workloads: We can provide ongoing, valuable support and services to ensure the ongoing health and wellness of your cloud deployments. We offer pre- and post-deployment performance optimization services. Our expert consultants remotely evaluate your environment, review your PostgreSQL database parameters, and can make adjustments according to your workload

Learn more about EDB Postgres at www.edbpostgres.com

EDB
POSTGRES

and use case. We will guide you on using the proper instance type and server size to meet your current needs for cost and performance, and help you plan for the future.

We offer premium management services: With the EDB Managed DBaaS Service, you can extend Postgres in the public cloud with Oracle-compatible functionality. EDB Managed DBaaS allows you to experience the same ease of deployment and elasticity as with Amazon, but without the trade-offs in performance tuning, configurability, customization, and control.

The EDB Managed DBaaS Service is a fully managed, "white glove" DBaaS service for Postgres offering proactive 24x7x365 database and infrastructure monitoring by certified Postgres specialists—offering a level of support that AWS does not.

Our EDB Managed DBaaS service combines our RemoteDBA offering with on-demand database provisioning and management capabilities. Additionally, the remote DBaaS operations team manages the AWS cloud infrastructure on your behalf. EDB provisions and manages a dedicated DBaaS management environment in AWS for each DBaaS customer.

Your users get the benefit of having our team deploy fully managed instances for them and can also choose to perform self-service provisioning of unmanaged database instances, for development use cases, for example. Additionally, EDB's consulting services are included for on-boarding activities and migration assessments.

Learn more about EDB Postgres at www.edbpostgres.com

EDB™
POSTGRES

MIGRATE ORACLE WORKLOADS WITH A TRUSTED PARTNER

This journey to migrate Oracle® workloads to the cloud might be complicated, but we are here to help. EDB has the experience and expertise to help you configure and tune your cloud deployments, assist with implementing your migration, and then provide the long term support and services needed for your success.

ABOUT EDB

EnterpriseDB (EDB) provides enterprises with the products, resources, and expertise required to confidently run large scale and highly available deployments of Postgres across on-premises, hybrid, multi-cloud, and DBaaS environments with reliable enterprise-class vendor support, services, and expertise.

The EDB Postgres™ Platform provides compatibility with native PL/SQL, the programming language that runs on Oracle database products, and includes automated migration tools to help save time and minimize uncertainty in the migration process.

EDB is an innovator and major contributor to the Postgres community, serving 20% of the Fortune 500 and 15% of Forbes Global 2000 companies worldwide.

EDB is based in the Bedford, Massachusetts with offices around the globe.

Call our nearest location or email info@enterprisedb.com

US +1 781-357-3390 or 1-877-377-4352	**UK** +44-20 3740 6778
GERMANY +49 322 210 97906	**FRANCE** +33 975 187 082
	NETHERLANDS +31 (0)20 808 0937
SWEDEN +46 844 683 476	**JAPAN** +81-50-5532-7038
POLAND +48 223 079 848	**KOREA** +82-2-6007-2500
AUSTRALIA +61-2-8019-7055	**INDIA** +91-20-66449600/601

EDB POSTGRES