

SAP on Google Cloud: Migration strategies

Overview



Contents

About this document	3
Introduction	3
Migration strategies	4
Move to cloud	4
Option 1: Lift and shift	5
Option 2: Rebuild and reinstall	6
Recommendations for all move approaches	6
Migrate to cloud	6
Transform	9
Reimplement (greenfield migration)	10
Planning and preparation	11
Sizing the target system	11
Cleaning up the source system	11
Archiving and deleting data	12
Recommendations	13
Summary	13
Further reading	14
Appendix 1: Database log shipping mechanisms	15
Appendix 2: Lift-and-shift migration scenarios	16
Homogeneous OS migration scenarios	16
Homogeneous database migration scenarios	17
Appendix 3: Heterogeneous migration scenarios	19

About this document

This document is part of a series about working with SAP on Google Cloud. The series includes the following documents:

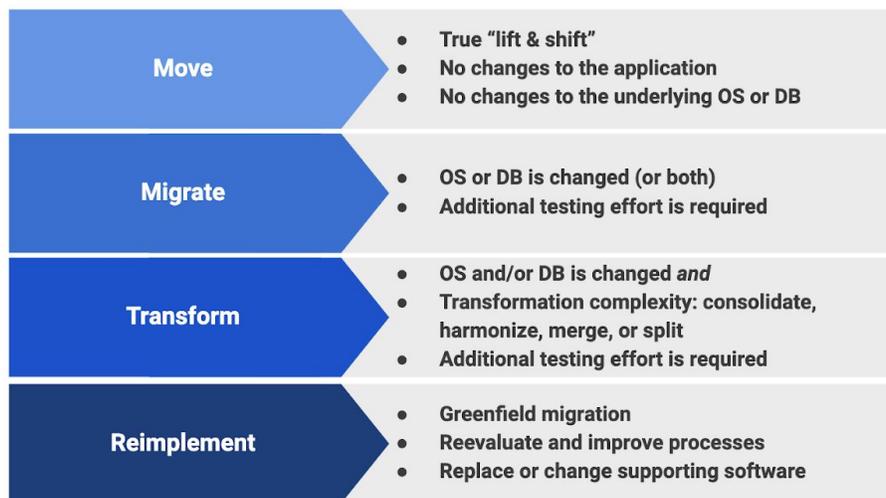
- [High availability](#)
- Migration strategies (this document)
- [Backup strategies and solutions](#)
- [Disaster recovery strategies](#)

Introduction

This document describes the most common migration strategies, use cases, and options that you use when you move SAP systems to Google Cloud. Depending on requirement for the source and target systems, different scenarios apply and determine how you move or migrate the systems to Google Cloud. This document is intended for SAP technical architects.

The following diagram summarizes the scenarios. Which methodology you choose depends on your business and technical requirements.

Move | Migrate | Transform | Reimplement SAP to Google Cloud



The four scenarios are the following:

- **Move (homogeneous migration).** You move an application to the cloud but make no changes to the application or to the underlying databases.
- **Migrate (heterogeneous migration).** As part of the migration, you change the operating system, the database, or both.
- **Transform.** As part of the migration, you might change the underlying operating system and database, but you also make significant changes to the application.
- **Reimplement.** You reimplement an application from scratch and migrate only relevant data from the source system to the target system.

A typical SAP system is made of a number of layers that include application servers, central services, databases, and supporting services. Each layer might require a different migration strategy. For example, migrating an application server from SUSE Linux to SUSE Linux is a homogeneous migration. But if the underlying database moves from Db2 on AIX to Db2 on x86, the move is a heterogeneous migration. Therefore, you might need to consider a combination of approaches when you plan the migration. In many cases, a move or migration might be only the first step in a longer-term transformational program.

Migration strategies

This section provides details about migration strategies for the different scenarios described earlier.

Move to cloud

In a move, you move an application to the cloud but make no changes to the underlying applications or underlying databases. There are two move options:

- Lift and shift, in which all components are moved as is.
- Rebuild and reinstall, in which you create an environment to restore relevant configuration files and databases to.

When you move an application to the cloud, you need to consider the differences between the source and target environments. A migration might initially look simple, such as moving a Windows VM on-premises to a Windows VM in the cloud. However, the application might require access to technologies that are specific to the source environment and are unavailable or function differently in the target environment—for example, the application might rely on NFS or SMB file shares, or require access to a local NTP time server.

Option 1: Lift and shift

You can consider a lift-and-shift strategy under these circumstances:

- If the status of the source systems is known and documented, and the system conforms to SAP's best practices.
- For very large landscapes where a rebuild-and-reinstall strategy requires too much time and effort.
- For databases that will experience a low volume of updates and database performance is less critical during the migration period.

Before you decide on a lift-and-shift approach, you need to consider characteristics of the source system. Any operating system, database, or application to migrate might have configurations that are native to the source environment.

Examples include the following:

- Hardware-specific kernel parameters.
- Hard-coded network configurations such as entries in `/etc/hosts` files.
- Database file layouts that are tuned to the underlying storage subsystem.
- Hardcoded links to file shares.

Google Cloud provides a number of accelerators to assist with the move to the cloud. For typical lift-and-shift operations, Google Cloud provides [Migrate for Compute Engine](#) (formerly known as Velostrata). This uses streaming technology to minimize migration time, and it provides rightsizing recommendations before you begin, to help you choose appropriate instance types. Migrate for Compute Engine integrates with VMware vCenter to provide a single control interface for managing VM migrations.

When you're considering Migrate for Compute Engine, you must make sure of the following:

- The source environment is running on bare metal or on VMware.
- The target environment is a supported combination of OS/DB and source environment specific configurations.
- Mitigations are documented and tested.

Migrate for Compute Engine is not recommended if you need to migrate a zero-downtime production system. For a complex, three-tier architecture, we recommend a backup/restore and recovery using log shipping. If you're using SAP HANA, we recommend HANA system replication.

Option 2: Rebuild and reinstall

The second approach to move to the cloud is to rebuild the source environment in the cloud and tune the new environment to the target infrastructure. Advantages of this approach are the following:

- You can install the latest supported OS and application executable versions without impacting any source systems.
- You can eliminate legacy undocumented workarounds.
- You can configure the size of the target environment for performance and cost.

This might initially seem like substantial effort, but you can take advantage of accelerators offered by Google Cloud. For example, there are a number of [deployment manager templates](#) that can assist in automating the deployment of Google Cloud infrastructure. By using these templates, you can reduce the time for installing a fully clustered database environment from days to minutes, and you can deploy multiple environments in parallel. You can customize these templates to configure a fully automated rollout that meets your needs. Additionally, you can create [custom virtual machine images](#) for both manual and automated rollouts. These images can contain any features you need for your target environment, such as pre-installed monitoring and security tools.

After the core environment has been deployed, you can restore databases to the target architecture. If you use native database replication techniques, such as [database log shipping](#) or HANA system replication, you can keep system downtime to a minimum. This approach also lets you create a fully isolated test environment on the target platform without impacting the source environment.

Recommendations for all move approaches

Whether you use the lift-and-shift approach or the rebuild-and-reinstall approach, we recommend that you reinstall the OS and database executables. We also recommend that you use database backup to restore the data or HANA synchronization for critical migrations to Google Cloud. This is especially true for environments that have a long maintenance history, where the source system state might not be fully documented.

Migrate to cloud

In a heterogeneous migration, you change the operating system or database or both. For example, you might migrate:

- From RISC-based HPUX, Solaris, or AIX to an x86-based OS like Windows or Linux.
- From Windows to Linux or Linux to Windows.
- From SAP Sybase ASE to SAP HANA.

In general, there are few or no changes to the application itself. For these scenarios, you can use Google Cloud [deployment templates](#) to accelerate the process of building the Google Cloud compute resources.

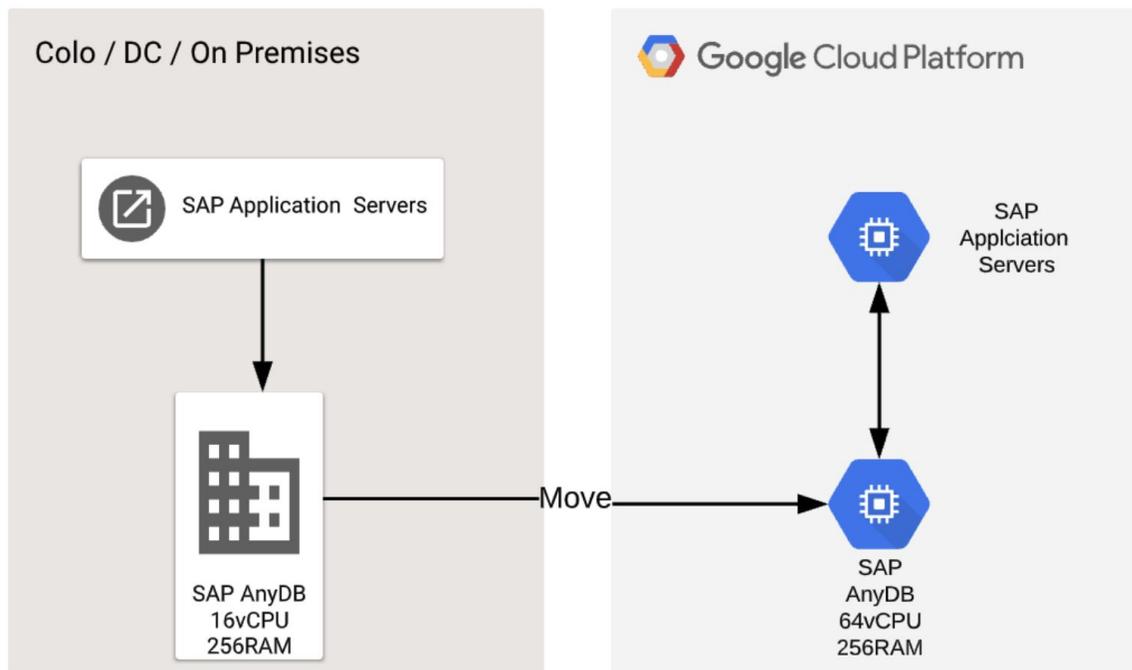
For a full list of operating system and database migration scenarios, see [Appendix 2](#).

For heterogeneous migrations, if HANA or Sybase ASE are the target databases, the SAP-supported method is to use the [SAP DMO tools](#). For other targets or for older SAP systems that are not supported by DMO, we recommend the classic [heterogeneous migrations](#) approaches documented by SAP.

In some cases, the future system architecture is supported on Google Cloud (for example SAP HANA) but the current architecture is not supported. In this case, Google Cloud provides a path forward such as [MPS](#) and [PowerCloud](#), which can host Oracle-based and AIX-based systems to ease the transition without the need to purchase additional on-premises hardware during the migration period.

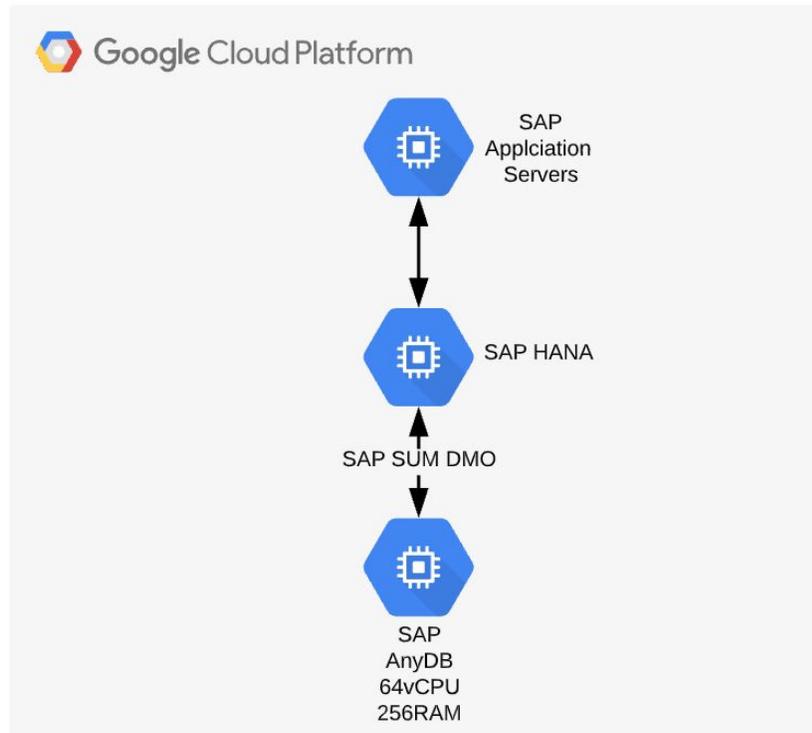
Consider a system for which you want to migrate the database from on-premises Microsoft SQL Server to SAP HANA on Google Cloud. In this case, a typical approach is to create a large system at the source with many CPUs to accelerate the database export to the target by using SAP migration tools. However, you might not want to procure a new system in the source environment because of the time and expense it requires. You can take advantage of the virtual nature of Google Cloud and use a two-step approach.

The first step is to use a lift-and-shift approach to build a temporary system with more vCPUs on Google Cloud, as shown in the following diagram:



For a production environment, you use a data recovery approach to build a temporary landscape on the larger infrastructure. You should build the target HANA database in advance and should consider using the Google Cloud deployment templates. At cut-over, the final logs are shipped to this temporary database and the heterogeneous migration can commence.

After the cut-over, you verify that the target system reconciles with the source and operates as expected. You can then switch off the source system to ensure that no further updates are made to the source database.



The second step is to migrate from the temporary database to the final database using increased CPU and memory capacity. When the OS and database migration phase has completed, you can shut down and delete the source temporary databases on Google Cloud and scale the target database down to its required size.

Transform

In a transform migration, you might change the underlying operating system and database, but the migration might also require significant changes to the application. This can involve an upgrade to the application, merging applications, merging company code, splitting systems, or upgrading and transforming to S/4 HANA. Google Cloud is working with a number of partners such as SNP and GekkoBrain to ease and accelerate transformational migrations.

A transform migration is often a combination of the migration strategies discussed earlier, such as building a core system based on historical data and customizing, with additional major transformational activities.

These transformational activities can include the following:

- **Consolidation:** This involves extracting data from other systems into a single SAP system. Data extraction and manipulation tools are used to read data from the source systems, and standard SAP APIs to import the data into the target system. The goal of consolidation is to lower the operational overhead of maintaining multiple systems and to give end users a seamless experience.
- **Harmonization:** Harmonization is used for improving the overall quality of the data within SAP landscapes by deduplicating data. For example, it might mean consolidating charts of accounts, converting new GLs, reorganizing profit centers, or merging company code or master data into a single location. There are a number of approaches to harmonizing. You might harmonize the data within the same system, in which case a migration is required as described earlier. Or you might build a new system and import the data into the system using SAP APIs.
- **Split:** The system to be migrated is split into multiple target systems. This might be the approach if a single system is too large for current hardware to support, if legal carve-outs are required for a change in laws, or if there are partial or complete entities.

For these kinds of migrations, downtime is less of a concern, because the target system will likely be built ahead of time, before the master, transactional, and configuration data is loaded during cutover.

Google Cloud is able to accelerate a transformational migration program with partners:

- [SNP Group](#) provides an optimized path to digital transformation and has completed over 10,000 projects with customers. The SNP Group solution gives you a deeper understanding of the SAP landscape that's targeted for migration to Google Cloud. It also lets you selectively migrate your SAP landscape.
- [Gekkobrain](#) develops analysis and automation software for SAP performance optimization and migration before S/4HANA or cloud migration. They also provide a transparent overview of your HANA and S/4HANA readiness so you know exactly how much effort is needed to migrate.

Reimplement (greenfield migration)

In a greenfield migration, you fully reimplement the system, such as S4 HANA. Reimplementing the application is often an opportunity to reevaluate and improve business processes and to leave behind legacy code, customizations, and data.

A reimplementation of SAP is generally not a like-for-like reimplementation. Instead, you undertake a full review of current business processes, taking advantage of functionality that's

available on Google Cloud. You generally deploy a new landscape, which includes new development, new quality assurance, and new production systems. The SAP application that you deploy is likely to change. For example, you might replace the SAP portal with SAP Fiori, and replace other applications such as CRM with a SaaS-based system.

For the greenfield migration case, the automated deployment templates provided by Google Cloud can accelerate the process of installing and configuring the target system.

Planning and preparation

This section provides guidance for planning a migration and suggestions for how best to prepare for the migration, regardless of which approach you use.

Sizing the target system

Accurate sizing of the target system in the cloud is key to the success of a migration project. This is not only to make sure the environment performs as expected, but to help you make sure that you realize the anticipated cost savings of moving to the cloud. When you look to move to Google Cloud, you should consider right-sizing your environment—that is, size it to what's needed now rather than to what you anticipate in three to five years. This is a key differentiator between running on a fully virtualized cloud environment versus running on an on-premises system, or running on a cloud where you commit to a VM configuration for a predefined period of time. This is especially relevant to non-production systems, which are often oversized and underutilized. Consider also when the environments need to be available; you can realize significant cost savings by shutting down environments when they are not in use.

When you're sizing the environment, it's often tempting to implement the target system at the same size as the source environment. In the case of a new system, you might be tempted to use the output of the SAP quicksizer, which is based on an anticipated growth rate of three (or more) years. However, you can get significant cost savings by considering current system usage and workload patterns.

Cleaning up the source system

Before you perform any migration, regardless of migration type, it's important that housekeeping jobs are run and completed in the source SAP system. When they run, SAP systems store a considerable amount of data that's not directly related to the business processes being performed. This can be log data that's stored in tables, as well as temporary data such as IDocs, especially in the case of SAP Business Warehouse (BW). Conducting a proper cleanup of the landscape helps to reduce the system size, which has an impact on the downtime of any migration. It also reduces storage and memory requirements in the target systems and lowers the risk of error during the conversion process.

Before you make the decision about migration types (especially for homogeneous lift-and-shift migrations), you should undertake an analysis of the current system state, including the following:

- **Understand performance characteristics and constraints of the source and target systems.** When you migrate to SAP HANA, the performance characteristics of the application server landscape will change. With a conventional disk-based database, the application servers spend a considerable amount of time waiting for the database to return the requested data. For SAP HANA, the wait time is considerably reduced. As a result, the application servers run at higher utilization for shorter periods of time. Therefore, you should reconsider the number of work processes, and the performance characteristics of the target processor types. For example, if the source application servers are running on the Intel Sandy Bridge CPU architecture and the target application servers will be running on the Intel Cascade Lake architecture, you don't need to size the virtual machine with the same number of vCPUs as the source.
- **Understand the primary interfaces.** As noted in the introduction, a system migration project is primarily an integration test project. Host names will change, IP addresses will change, additional and new firewalls will be in place. You should make a comprehensive list of all interfaces. You should identify hard-coded interfaces and any user-initiated interfaces, such as directory structures that are used for spreadsheet uploads and downloads. We recommend that you use virtual host names both on-premises and in Google Cloud to allow easy interface management.

Archiving and deleting data

Data archiving and deletion is a major key to the success of any migration, especially for SAP HANA. With the price of storage becoming cheaper and with performance improving significantly, data archiving is seen as an expense that creates overhead with little return on investment. However, with in-memory databases, data volumes are a key factor in infrastructure and license costs. Excessive data volumes also contribute in a major way to downtimes during a migration, especially in the case of a heterogeneous migration.

It's therefore vital that redundant data be moved to nearline or coldline storage (that is, to an archive) and that any blob-based data (scanned documents, images, and so on) be moved to a content store.

A best practice for fast and smart analytics on the archived data is also to load the data into BigQuery.

Recommendations

For a homogeneous migration, we recommend that you build the environment from scratch using deployment templates and restore the database from backup or replicate using database-replication technologies. This ensures a clean environment that's tuned for Google Cloud. It also takes advantage of Google services such as Google DNS and [Google Public NTP](#) without bringing legacy and infrastructure-specific configurations to the cloud that can cause issues. Using the replication methods described in this document, you can execute many test scenarios, such as dress rehearsals, performance tests, and integration tests, in a repeatable and consistent manner without impacting the source system. In addition, this approach can help you to achieve zero downtime during cut-over.

You can use Migrate for Compute Engine if the following are all true:

- Bandwidth is not a constraint.
- The operating system is in a known state.
- All operating system changes have been documented and can be revised to suit Google Cloud architecture.
- The database change rate is low and cut-over time is not critical.

For heterogeneous migrations, if possible, you should perform a lift-and-shift migration first. We recommend this if the combination of OS and database on the source is supported on Google Cloud. That way, you can allocate additional resources temporarily to accelerate the SAP SUM Database Migration Option (DMO) process.

Summary

The correct migration strategy is dependent on the criticality, complexity, and cost of the systems to be migrated.

In general, your approach to migrating to Google Cloud should be based on a clean, new environment that's built and tuned to Google architectures using [Google Deployment Manager](#) with prebuilt and customization deployment templates. These templates help to accelerate and automate the process of provisioning systems that are built to match SAP certification requirements.

For heterogeneous migrations, we recommend the SAP DMO/SUM migration strategy. However, you can accelerate the process by moving the system to Google Cloud first (if homogeneous migration applies) and then performing the DMO/SUM migration with appropriately sized infrastructure.

Because of the virtualized nature of the Google Cloud and per-second billing, having a parallel environment during the migration period should not be a cost concern. The parallel environment can be started and stopped when required, rather than having a full parallel migration environment sitting idle between test phases.

Further reading

- [Migrate for Compute Engine](#)
- [SAP DMO/SUM migration](#)
- [Database Migration Option of Software Update Manager 2.0](#)

Appendix 1: Database log shipping mechanisms

- [SQL Server Log shipping](#)
- [DB2 Log shipping](#)
- [MaxDB Log shipping](#)
- [Sybase ASE Log shipping](#)
- [HANA system replication guide](#)

Appendix 2: Lift-and-shift migration scenarios

Homogeneous OS migration scenarios

Source OS	Source chipset	Target OS	Target chipset	Migration options	Recommended
SUSE	x86	Redhat	x86	<ul style="list-style-type: none"> Rebuild and reinstall Backup/restore SAP directories 	Rebuild and reinstall
SUSE	x86	SUSE	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine 	Depends on testing strategy and downtime requirements
RedHat	x86	RedHat	x86	<ul style="list-style-type: none"> Rebuild and re-install Migrate for Compute Engine 	Depends on testing strategy and downtime requirements
Windows	x86	Windows	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine 	Depends on testing strategy and downtime requirements

Homogeneous database migration scenarios

Source DB	Source OS	Source chipset	TargetDB	Target OS	Target chipset	Migration options
Db2	Redhat	x86	Db2	Redhat	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
Db2	SUSE	x86	Db2	SUSE	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
Db2	Redhat	x86	Db2	SUSE	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall
Db2	SUSE	x86	Db2	Redhat	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall
Db2	Windows	x86	Db2	Windows	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
Oracle*	Redhat	x86	Oracle	Redhat	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
Oracle*	SUSE	x86	Oracle	SUSE	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
Oracle*	Redhat	x86	Oracle	SUSE	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall
Oracle*	SUSE	x86	Oracle	Redhat	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall
Oracle*	Windows	x86	Oracle	Windows	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
MaxDB	Redhat	x86	MaxDB	Redhat	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
MaxDB	SUSE	x86	MaxDB	SUSE	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine
MaxDB	Redhat	x86	MaxDB	SUSE	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall
MaxDB	SUSE	x86	MaxDB	Redhat	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall
MaxDB	Windows	x86	MAXDB	SUSE/Redhat	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall
MaxDB	Windows	x86	MaxDB	Windows	x86	<ul style="list-style-type: none"> ● Rebuild and reinstall ● Migrate for Compute Engine

* SAP workload on Oracle [can be supported](#) with Google Cloud Partners.

Source DB	Source OS	Source chipset	Target DB	Target OS	Target chipset	Migration options
SQLServer	Windows	x86	SQLServer	Windows	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine
SQLServer	Windows	Itanium	SQLServer	Windows	x86	<ul style="list-style-type: none"> Rebuild and reinstall
Sybase ASE	Redhat	x86	Sybase ASE	Redhat	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine
Sybase ASE	SUSE	x86	Sybase ASE	SUSE	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine
Sybase ASE	Redhat	x86	Sybase ASE	SUSE	x86	<ul style="list-style-type: none"> Rebuild and reinstall
Sybase ASE	SUSE	x86	Sybase ASE	Redhat	x86	<ul style="list-style-type: none"> Rebuild and reinstall
Sybase ASE	Windows	x86	Sybase ASE	Windows	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine
HANA	Redhat	x86	HANA	Redhat	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine
HANA	SUSE	x86	HANA	SUSE	x86	<ul style="list-style-type: none"> Rebuild and reinstall Migrate for Compute Engine
HANA	Redhat	x86	HANA	SUSE	x86	<ul style="list-style-type: none"> Rebuild and reinstall
HANA	SUSE	x86	HANA	Redhat	x86	<ul style="list-style-type: none"> Rebuild and reinstall
HANA [†]	Linux	Power	HANA	SUSE/Redhat	x86	<ul style="list-style-type: none"> Rebuild and HANA system replication
HANA [†]	SUSE/Redhat	x86	HANA	Linux	Power	<ul style="list-style-type: none"> Rebuild and HANA system replication

[†] SAP HANA on IBM Power Systems [can be supported](#).

Appendix 3: Heterogeneous migration scenarios

Source database	Source OS	Source chipset	Target database	Target OS	Target chipset	Migration options
Db2	AIX	Power	Any	Linux/Windows	x86	SAP DMO
Db2	HPUX	Itanium	Any	Linux/Windows	x86	SAP DMO
Db2	Solaris	Sparc	Any	Linux/Windows	x86	SAP DMO
Sybase ASE	AIX	Power	Any	Linux/Windows	x86	SAP DMO
Sybase ASE	HPUX	Itanium	Any	Linux/Windows	x86	SAP DMO
Sybase ASE	Solaris	Sparc	Any	Linux/Windows	x86	SAP DMO
MaxDB	AIX	Power	Any	Linux/Windows	x86	SAP DMO
MaxDB	HPUX	Itanium	Any	Linux/Windows	x86	SAP DMO
MaxDB	Solaris	Sparc	Any	Linux/Windows	x86	SAP DMO
Oracle	AIX	Power	Any	Linux/Windows	x86	SAP DMO
Oracle	HPUX	Itanium	Any	Linux/Windows	x86	SAP DMO
Oracle	Solaris	Sparc	Any	Linux/Windows	x86	SAP DMO
HANA 1.0	Linux	Power	HANA	Linux	x86	SAP DMO