



Competitive Database Migration Tools and Methodology

Migration to Microsoft SQL Server 2014 Using SSMA

Technical White Paper

Authors: Sudhir Gajre (DB Best), Badal Bordia (ASPL), Vishal Soni (ASPL)

Technical reviewers: Dandy Weyn (DB Best), Bill Ramos (Indigo Slate), Darmadi Komo (Microsoft)

Published: March 2015

Applies to: Azure SQL Database and Microsoft SQL Server 2014

Summary: Microsoft SQL Server is the world's most deployed database and is included in the Leaders Quadrant for the Gartner's Magic Quadrant for Operational Database Management Systems¹. Large numbers of customers are adopting SQL Server because of its low total cost of ownership; ease of management; support for a wide range of enterprise-grade features; and enterprise-class scalability, availability, and reliability. Microsoft SQL Server Migration Assistant (SSMA) helps organizations to modernize their platforms by migrating their legacy applications and databases from other vendors (including Access, DB2, MySQL, Oracle, and Sybase) to SQL Server. In this white paper, we will see how SSMA is helping to automate the various phases of database migration, such as data assessment, schema and application migration, data migration, and testing. We will also discuss how SSMA significantly speeds up—and reduces the cost of—migration at every step.

¹ SQL Server Blog, "Microsoft a Leader in Gartner's Magic Quadrant for Operational Database Management Systems" <http://blogs.technet.com/b/dataplatforminsider/archive/2013/10/31/microsoft-a-leader-in-gartner-s-magic-quadrant-for-operational-database-management-systems.aspx>, October 2013.

Copyright

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in, or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2015 Microsoft Corporation. All rights reserved.

Microsoft, Active Directory, Microsoft Azure, Excel, SharePoint, SQL Server, Windows, and Windows Server, are trademarks of the Microsoft group of companies.

All other trademarks are property of their respective owners.

Contents

- Why migrate to SQL Server? 4
 - Business considerations 4
 - Technical considerations..... 4
 - Architectural considerations 5
 - Customer evidence—Why migrate to SQL Server? 6
- Database migration approach and process 8
 - Database migration challenges and concerns 8
 - Application and platform migration approach..... 8
 - Database migration approach 9
 - Database and application migration process 10
- Data migration using SQL Server Migration Assistant..... 15
 - SSMA overview 15
 - SSMA migration—Automation of all phases of database migration 15
 - SSMA and database migration considerations 16
 - Azure SQL Database service migration..... 17
- What’s new in SSMA v6.0 17
- SSMA engineering and support 19
- Conclusion 19
- Key resources 20

Why migrate to SQL Server?

For any organization, staying on legacy platforms and technologies can negatively affect the IT budget and restrict the ability to grow the business and adapt to changing market dynamics. To ensure effective overall development, an organization must have a strategy for upgrading and modernizing infrastructure, including applications, databases, and other associated systems. For this reason, application and data platform modernization are considered among the top enterprise IT initiatives. According to a research analyst at Gartner², “*Application modernization and increasing agility will continue to be a solid driver.*”

For any organization, the right approach to modernization will depend upon its business requirements; its current infrastructure, including data and applications; and the behavior of people using these applications and infrastructure. An enterprise needs to visualize the future of its IT assets, and should be able to justify its IT investments from a business perspective. This will help it identify the best approach and prepare a roadmap for an efficient transition to modernization. Staying with legacy applications and database platforms prevents an organization from putting its best foot forward against the market dynamics.

Microsoft SQL Server helps organizations break the traditional barriers of legacy databases, and delivers a modern application platform with mission critical performance, faster insights, and support for cloud-based scenarios. SQL Server 2014 enables high-performance, in-memory, and security technology across a wide variety of workloads, with up to 30x faster transactions and 100x query performance gains. Powered by built-in in-memory technology and reinforced with enhanced security, scalability, and availability, SQL Server 2014 provides support for all types of workloads, including online transaction processing (OLTP), data warehousing, business intelligence, and analytics. SQL Server 2014 is designed to work in a hybrid environment that spans on-premises resources and the cloud. The familiar and easy-to-use tools make it easy for organizations to adopt and grow with this highly performant technology platform.

Business considerations

The **total cost of ownership (TCO)** stands out as a major business consideration for the adoption of SQL Server 2014. SQL Server delivers full enterprise-grade features at a lower TCO—and unlike the products from most other competitors, no costly add-ons are required. SQL Server 2014 supports mission-critical operations and modern app development with enterprise-class capabilities built into a single solution. SQL Server offers an integrated high-availability solution with faster failover and reliable backups, delivering new standards in enabling mission-critical operations. SQL Server 2014–based solutions are easy to deploy, manage, and monitor, which helps organizations reduce management costs, drive incremental revenue from improved systems, and improve IT and end-user productivity while reducing TCO.

Technical considerations

Microsoft SQL Server 2014 is bundled with many new features, making it a compelling upgrade for most organizations. These features enable organizations to scale and transform their businesses, with up to a 30x performance gain for transaction processing using existing hardware and a greater than 100x performance gain for data warehousing. New features include:

- **In-Memory OLTP:** In-Memory OLTP is a memory-optimized database engine integrated into the SQL Server engine and optimized for online transaction processing (OLTP) workloads.

² Gartner Newsroom, “Gartner Says Cloud, Mobility and Open Source Will Drive Application Development Market to Exceed \$9 Billion in 2012,” <http://www.gartner.com/newsroom/id/2131115>, August 2012.

With this feature, certain tables and stored procedures can be moved into the main memory (RAM), which reduces the query response time and hence improves performance.

- **AlwaysOn enhancements:** Introduced in SQL Server 2012, AlwaysOn Failover Cluster Instances and AlwaysOn Availability Groups gain several enhancements in SQL Server 2014, including an increased number of secondary replicas and simplified wizards.
- **Microsoft Azure integration:** A new wizard provides the capability to directly host a SQL Server database in a Windows Azure Virtual Machine. There is native support for SQL Server database files stored as Windows Azure Blobs, providing a dedicated storage location for the databases.
- **ColumnStore Indexes:** New updateable Clustered ColumnStore Indexes help improve data compression and query performance for data warehousing workloads. This feature also supports archival data compression.
- **Buffer pool extension:** Solid-state drives (SSD) can be used as a nonvolatile random access memory (NVRAM) extension to expand the buffer pool when systems run out of memory.
- **Resource Governor enhancements for physical I/O control:** The Resource Governor can now be used to manage and restrict use of physical I/O resources in addition to CPU and memory resources—for example, by setting limits on the amount of CPU, physical I/O, and memory resources that incoming application requests can use within a resource pool.
- **Backup and restore enhancements:** SQL Server Backup to URL now supports backup to or restore from Windows Azure Blob storage service. Backups can be encrypted at the time of backup operation.
- **Improved query performance:** With a completely redesigned cardinality estimator, SQL Server 2014 has an improved quality of query plans and, therefore, improved query performance.

Architectural considerations

In recent years, there has been a dramatic change in both business needs and available hardware resources, which has brought a shift in the architectural paradigm. Now organizations require a stable yet dynamic environment, including a robust platform and handy tools to keep up with the changing needs of the market. Key architectural considerations for SQL Server 2014 include:

- **Modernized architecture:** A modernized architecture offers an agile and competitive environment, which organizations can harness for faster business results. It also increases consolidation and performance, and often provides better management capabilities. Microsoft SQL Server 2014 delivers new ways to efficiently store and use data, supporting most modern applications. With the new in-memory technology, it is optimized for various types of workloads, including online transaction processing (OLTP), data warehousing (DW), and analytics and business intelligence (BI). Its wide range of new features helps improve performance dramatically, while providing a base for hybrid cloud-based environments.
- **Consistent architecture:** Many customers have multiple database technologies implemented in their environments, which is usually a costly affair and requires complicated management efforts. A single database for both on-premises and cloud environments provides a consistency of architecture across the environment. SQL Server 2014 uses the same familiar architecture, the same T-SQL language, and the same APIs across on-premises and cloud environments to connect to the applications. This provides the additional advantage of consistent experience and performance across the entire environment.

- **Simplification and standardization:** A consistency in architecture also brings simplification and standardization of processes. For example, instead of needing several tools to manage updates and maintenance, a single standard process could work for all databases across the entire environment. Similarly, organizations can avoid the complexities of managing multiple licenses from various vendors.

Customer evidence—Why migrate to SQL Server?



“By using SQL Server 2012, we can look at dynamic and near real-time reporting alongside weekly and monthly static reports. Our goal is to improve business agility and optimize our offerings with faster access to analytics.”

Cameron Brill,
Auckland Airport,
Technology Manager

Auckland Airport wanted to replace its existing IBM Cognos and Oracle DW-based solutions because the managers wanted more flexibility and faster access to information. They were looking for a solution that could integrate with various data sources—including commercial business units, aeronautical systems, baggage-handling, engineering systems, and sensitive information from external systems—while enabling self-service BI for its managers.

Solution:

The new solution, based on Microsoft SQL Server, handles 13 million transactions each month from duty-free sales alone, and it includes 100 million rows in the largest table. It uses 25 dashboards and more than 120 interactive charts and scorecards.

Benefits:

- **Improved business agility and faster access to data**, as business managers have direct access to data without having to wait for monthly reports.
- **Enhanced data oversight and compliance**, with easy-to-produce, highly prescriptive audit reports.
- **Self-service BI tools** provide simplified and accelerated reporting.

For more details, see <https://customers.microsoft.com/Pages/CustomerStory.aspx?recid=13485>.



“With SQL Server Migration Assistant for Oracle, we can migrate our databases to SQL Server without any trouble.”

Mr. Keiichi Tanaka,
Senior Leader of the Business
Partner Department,
SRI Systems

Sumitomo Rubber Industries (SRI), Ltd., wanted to migrate their stand-alone workloads, which ran on HP Itanium servers, the HP-UNIX OS, and Oracle database software. It wanted to facilitate international growth and reduce environmental impact by consolidating server computers and standardizing IT. SRI wanted to update its mission-critical infrastructure to support server virtualization, improve data insight, and streamline processes.

Solution:

In addition to migrating its Oracle database to the Microsoft SQL Server using SSMA, SRI replaced its Itanium server computers with 64-bit(x86) hardware from NEC and virtualized servers wherever possible. Engineers plan to develop all new software with the Microsoft .NET Framework or Java. By end of 2016, SRI plans to migrate majority of its mission-critical systems to Microsoft platform.

Benefits:

- SRI forecasts **reduced software license costs** of ¥88 million (US \$1 million) in five years just by migrating its dealer management system to the Microsoft platform.
- The unified Microsoft infrastructure increased collaborative development and innovation across different teams.
- The new enterprise-wide BI solution improved data integrity and data insights.

For more details, see <https://customers.microsoft.com/Pages/CustomerStory.aspx?recid=16398>.



“Not only did we increase the functionality and provide continuity of business operations, we also reduced the cost.”

Fadi Areeshi
General Cars Syndicate,
SQL Server DBA

General Cars Syndicate (Naqaba) is responsible for providing secure and comfortable transportation for pilgrims. With more and more of their ticketing, coordination, and other operations managed by their Applied Systems portal, Naqaba realized the need to improve the reliability and availability of their Oracle-based system. It needed to move to a better database solution that would provide high availability, high resilience, an effective continuity of business and disaster recovery strategy, and lower license fees.

Solution:

The Oracle database was readily migrated to SQL Server 2008 using the SQL Server Migration Assistant (SSMA) for Oracle, which was a very cost-effective solution for migrating and validating the migration of both code and data from Oracle to SQL Server. By combining the witness server with asynchronous database mirroring, this solution delivered a truly redundant, resilient, and high-availability system.

Benefits:

- High performance and faster access to data through tight integration among Microsoft products.
- Minimal conversion was required to the .NET applications, accelerating the project’s time-to-value.
- Achieving significant productivity gains with almost no change in overall cost.

For more details, see <https://customers.microsoft.com/Pages/CustomerStory.aspx?recid=3034>.

Database migration approach and process

Database migration challenges and concerns

Database migration is usually considered a complex task, not just from the technology perspective, but also as a deliberate part of overall business strategy. Here are the key challenges that organizations face when planning for a database migration:

- **Value realization:** Making the right decision about which data to migrate is the foremost concern for organizations. Data storage and management involves huge investments of cost and time—as does data migration. Without proper cost and value estimates, an organization might migrate all data sources to target systems, only to realize later that migrating the not-so-critical databases to newer technologies cost more than the actual value of the data itself.
- **Risk:** In-house database administrators (DBAs) are not accustomed to performing database migration as a part of their regular day-to-day activities, and might find it difficult to identify all the possible risks and dependencies up front. Therefore, even after proper planning and methodologies, there is a risk of exceeding time and cost budgets when carrying out migration projects.
- **Skills:** Efficiently migrating a large or complex database with multiple dependencies requires qualified talent. The involvement of sufficiently skilled personnel can ensure minimal data-quality issues post-migration. In addition, the existing users or developers of the technology must be re-trained to use and understand the new applications and platforms quickly and easily for effective adoption.

Application and platform migration approach

For any organization, the actual approach for migrating applications and platform will vary, as it depends on the existing infrastructure and the business requirements. However, based on the analysis of general patterns, a typical approach can be defined.

- **Business and technical/architectural case:** If an organization is using a small number of database instances, migration can be a comparatively easy task. But organizations usually have large database environments with hundreds of applications and supporting elements bundled together. Database migration in such cases, known as *platform migration*, is very complex and requires specialized skills and procedures. A high-level assessment of the entire environment (most likely using an automated assessment tool) is essential to understand the potential technical challenges. Assessment reports can be used to obtain a rough estimate of the man-hours required to migrate the entire platform. Based on this, a business case can be prepared with a return on investment (ROI) estimate. This will help decision makers analyze and justify the investments in migration.
- **Pilot run—migrate and assess sample applications:** Some sample applications with different levels of complexity—for example, low, medium and high complexity—can be selected for migration to SQL Server. Migration of these samples can provide a fair idea of the actual complexity involved in migrating the entire platform. The assessment reports for these application migrations can help customers to visualize and evaluate the complexity of the overall platform migration.
- **Entire platform migration:** Migration of the entire platform is a complex process, and it requires robust planning and effective execution by skilled engineers. Additional automation tools can be used to help carry out the transition in an effective manner.

Database migration approach

The database and application migration process can be approached like a software development project. The translation of code from the source system to the destination system involves a defined methodology and professionalism. It can be broken down into different phases (Figure 1).

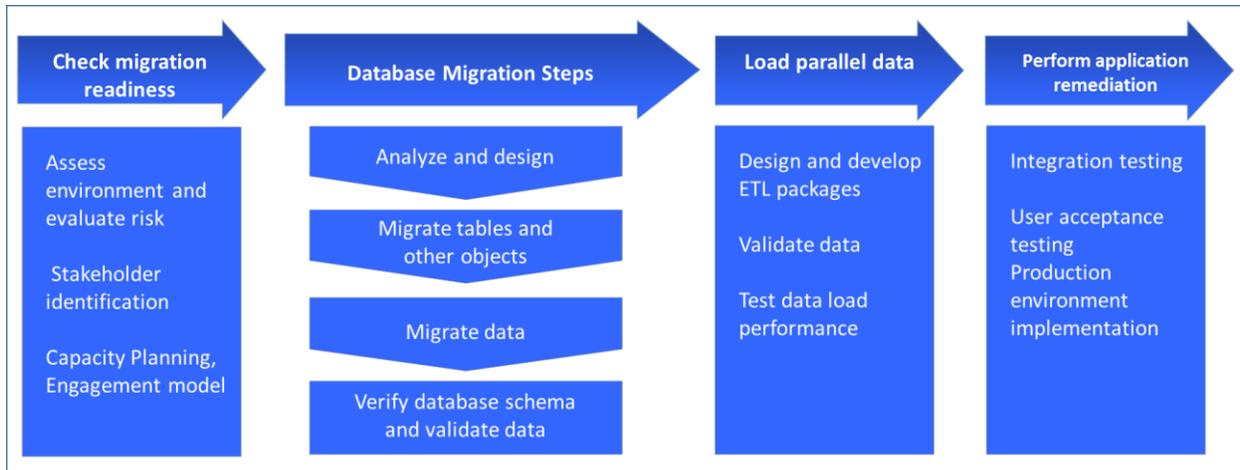


Figure 1. Database migration approach

- **Check migration readiness:** The initial step when preparing for a database migration is to assess the database environment, so that the risks can be evaluated. Assessment reports should include information such as the amount of migration that can be automated and estimates of the manual effort required. Based on the key findings from this assessment report, the organization can proceed with capacity planning, including determining the hardware and software infrastructure. Then the engagement model to be used during the migration process can be defined.
- **Database migration steps:** An organization should analyze the existing database, and then design SQL Server database to provide similar functionality. Next, a prioritized migration roadmap should be prepared, based on business objectives. Once that is done, database objects can be migrated, including tables, views, triggers, PL/SQL packages, Java code, and stored procedures. When the schema is ready in the destination database, data can be migrated. When both schema and data are ready in the destination system, the database schema and the validity of the data can be verified.
- **Load parallel data:** If extract-transform-load (ETL) packages were used to load the data from or to some external data stores, those packages need to point to the new database location. Any data extraction processes from the original database will also need to be replicated within the new SQL Server instance. Load performance can be checked by running some test loads. Once performance has been verified, security and monitoring functions needs to be set up and validated.
- **Perform application remediation:** Finally, testing for data integration with the applications that used data from the original database is required to ensure that the applications work with the new SQL Server instance. This should be followed by user-acceptance testing, and training and knowledge transfer to the respective stakeholders. Production-environment implementation and post-production support activities follow as for any software project activity.

Database and application migration process

The process for database and application migration can be thought of like a software development project. The methodology and professionalism required for the translation of code from system A to system B involves planning, transformation, and tuning, which can be broken down into different phases, as shown in Figure 2.

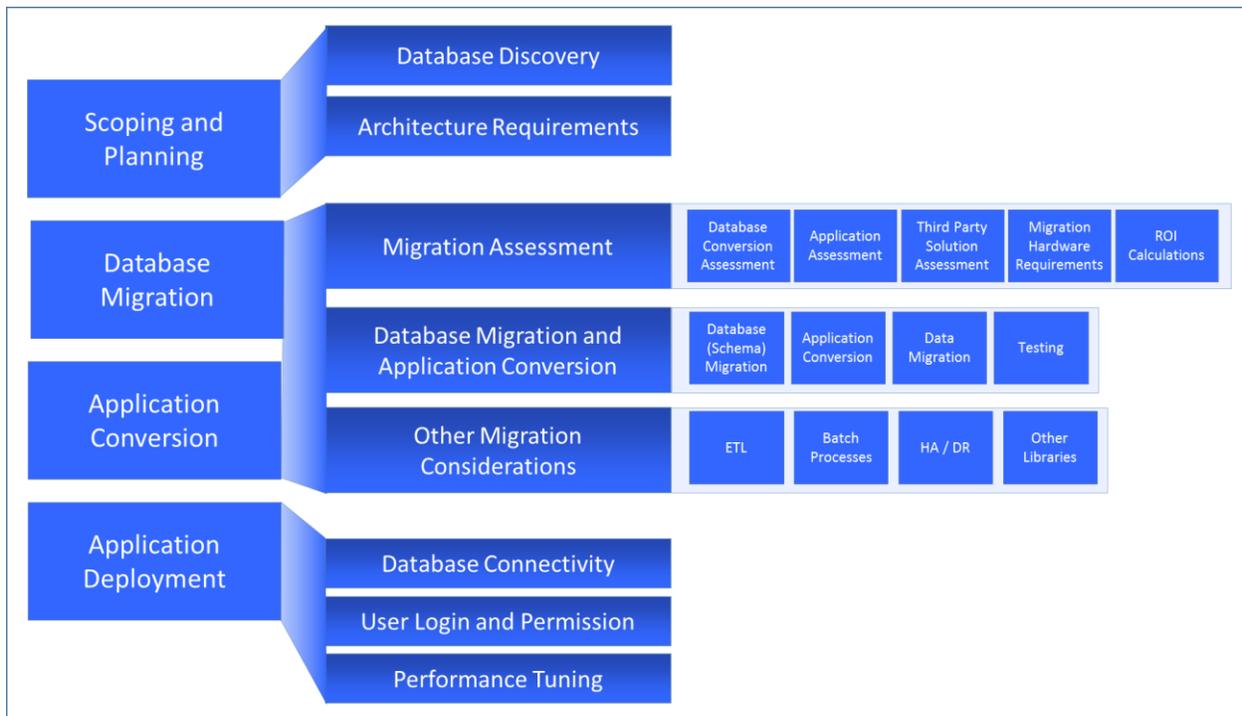


Figure 2. Database and application migration process

- **Scoping and planning:** As with any software development project, the initial requirement for data migration is to define the scope of the project. This step includes identification of all database instances and supporting architecture requirements involved in or affected by the migration.
 - **Database discovery:** Organizations usually have multiple database instances from several vendors spread across their environments. Therefore it is essential to prepare a complete inventory of all software residing throughout the environment when planning a database and application migration. Automated discovery tools like Microsoft Assessment and Planning (MAP) Toolkit can be used for this. With the MAP Toolkit, an organization can perform discovery across its entire environment, and prepare an inventory of the entire database-related infrastructure, including instances of Oracle, DB2, Sybase, and SQL Server. This helps the organization understand the high-level architecture of the entire environment. For example, a customer can know that its environment has 40 percent SQL footprints, 20 percent DB2, and 40 percent Oracle. Based on this information it could evaluate the efficiencies of each database vendor, and identify which database instances it should migrate.
 - **Architecture requirements:** Applications do not run in isolation, but require supporting infrastructure, like a high-availability solution, security, and other features. When migrating an existing database to a SQL Server–based system, the entire architecture

may need to be remodeled. For example, if an organization has an Oracle environment, and it is using Oracle GoldenGate or Oracle Fail Safe for high availability, migration from Oracle to SQL would also mean providing similar functionality using AlwaysOn. There could also be external dependencies on the database systems, like shell scripts running once every 15 days to upload data to an HR system, or feeds that load data to external (partner) applications. These tasks would also need to be either re-engineered entirely (using SSIS) or transformed to PowerShell-based tasks. A thorough evaluation of dependencies can be a critical factor in defining an effective migration approach.

- **Database migration and application conversion:** Before beginning the actual database migration and application conversion, an assessment of the environment can help to identify the actual effort and complexity involved in the migration process.
 - **Migration assessment:** For any organization, running an assessment of the entire environment (including the production environment) can take a long time, and may put the stability of production systems at risk. Tools like Microsoft SQL Server Migration Assistant (SSMA) can perform the assessment without putting the actual environment into risk, as there is no need to run the tool on the production system, or on any system. SSMA can be used offline, on an extract of the schema (including schema, table definitions, and PL/SQL code), and assessment reports can be generated. SSMA can also provide reports about the percentage of automation supported for migrating any other database environment (including Oracle, Sybase, Access, DB2, and MySQL) to an SQL environment, identifying the code which can be directly converted to equivalent Transact-SQL code, and estimating the effort required to convert the remaining code manually. It can provide a general estimate of man-days, man-months, or man-years of effort for an entire platform migration. Based on this, a business case for migration can be framed. A Migration Assessment involves the following key steps:
 - **Database conversion assessment:** Conduct a basic database assessment (Oracle, DB2, Sybase, MySQL, and Access) to see what can be converted, how much time it will take, what can be automated, and what will need manual effort.
 - **Application assessment:** Assess the applications to make sure that the embedded code, dynamic code, connectivity, and other relevant factors residing in the current environment can be migrated to or developed in the destination environment with the same effect.
 - **Third-party solution assessment:** Assess how loosely coupled or tightly integrated any third-party solutions are to the database.
 - **Migration hardware requirements:** The hardware compatibility of the destination system needs to be assessed before performing the migration. The destination system may be either on-premises (SQL Server) on a physical machine or a virtualized environment (like Hyper-V), or it may be a cloud infrastructure (IaaS) or cloud-based service (Azure). The readiness of the hardware needs to be assessed to ensure that the physical environment is ready for migration.
 - **ROI calculations:** Based on the above assessments, the upfront investment in hardware, software, and other resources needs to be identified. Then, based on the estimated savings with the migration, the break-even point for the investment can be computed to justify the ROI.

Once all of the assessment-related activities are complete and reports are obtained, a **business case** can be prepared. This report allows the business decision makers to understand the benefits and the risk components involved with the entire migration process, and helps with the decision-making process regarding some critical applications. It also helps to identify technical obstacles in the data migration process, which can alert the system administrators to potential problems.

To help with the business case, Microsoft provides a **risk assessment** or **risk map**, which provides a graphical representation of the assessment of the overall portfolio of the organization (Figure 3). The risk assessment evaluates all databases, applications, infrastructure, tests, replications, roll-outs, reports, ETL processes, and so on, and categorizes them based on the probability and potential effect of the occurrence of risks. This helps decision makers understand which applications have a higher probability of risk, and which could have a greater effect due to that risk, and aids decisions about which applications or databases should be migrated first.

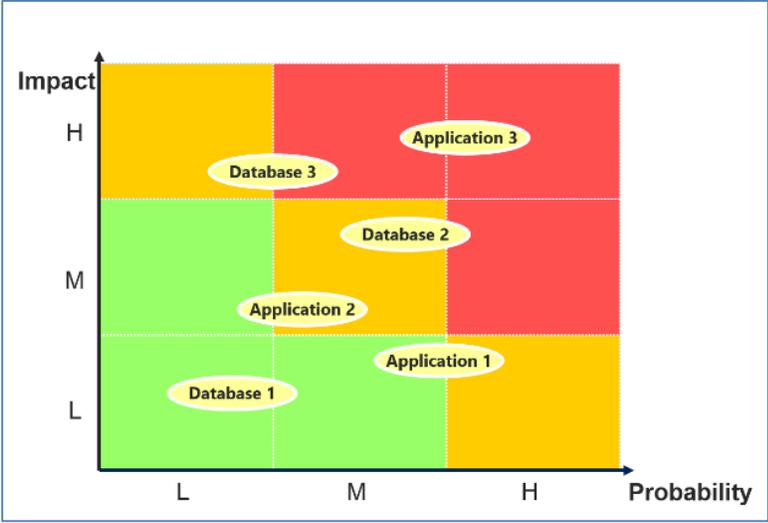


Figure 3. Sample risk map

An **applications migration map** (Figure 4) also provides business leaders, CIOs, and managers an overall view of the complexity involved in the migration. This report is based on simple graphs or PowerPivot charts, where the size of a bubble represents the effort required. The applications migration map can be used to identify the applications with which to start the migration, while considering the efforts involved and the potential risks.

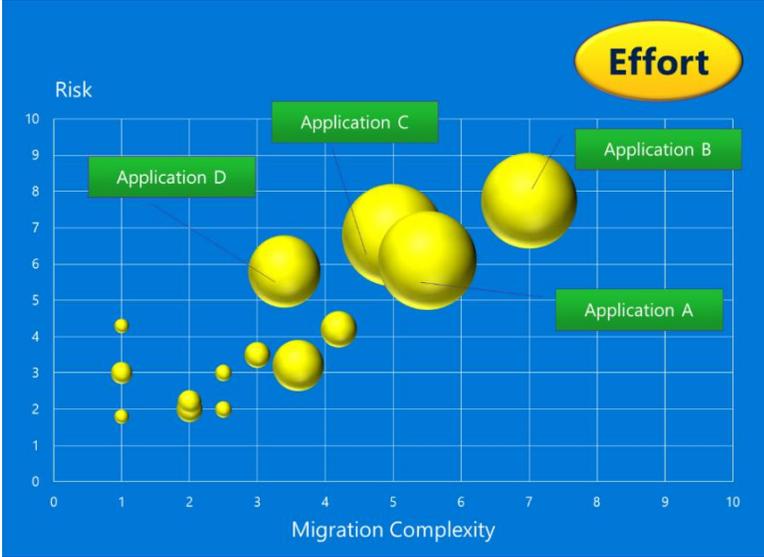


Figure 4. Sample applications migration map

- **Database migration and application conversion:** SSMA can also help organizations automate the actual conversion process. It supports a direct conversion of schema and data migration from source system to destination system. It also supports migration of embedded SQL statements. In any database application, code may not exist solely inside the database—it may also be stored outside the database, as with a PowerBuilder application having embedded PL/SQL code. There could be even a middle tier. All embedded SQL statements needs to be considered for conversion, which SSMA does. The key steps involved in the database migration and application conversion stage are:
 - **Database migration:** This includes the schema translation and the code translation (of packages, stored procedures, functions, and the like).
 - **Application conversion:** The applications accessing the database may not always need be converted. For example, a customer may have a PowerBuilder application. Ideally, this application could be migrated via conversion to .NET, but that approach might be very costly, and the customer may want to keep using the same application instead of spending so much on its migration. That application can be momentarily left alone, and kept running as-is. Thus, depending on the requirements, decisions about handling various applications need to be made.
 - **Data migration:** Before starting the migration, any third-party solution, if required, needs to be integrated. The actual process for migrating the data from one system to the other will vary based on the situation. For example, migration of a large database with 16–20 TB of data may be very different from migrating a 100-GB system. One migration technique is an instant turnover, where the current system is entirely shutdown while the data is migrated to the destination system. Once migration is complete, the new system is turned on. However, this

is not feasible with larger amounts of data (10–15 TB) or if the application is mission critical and cannot be turned off. An alternative approach could be a staged migration, where a snapshot of the existing database system is taken and migrated to the destination system while the current database still running. A mechanism is developed to determine the changes that occur in the database after the migration, and those changes are replicated in the destination database until both systems are fully synchronized.

- **Testing:** Data needs to be tested for validity after it has been migrated. To do this, a quick comparison of the data (including number of records, format, and so on) can be made between the source and the destination systems. SSMA can be used to perform this testing.
- **Additional migration considerations:** Aside from the above-mentioned factors, every database system has some ETL and batch processes going on to enable the flow of data into and out of the system. Also, there could be several additional system and database interfaces in the environment, such as interfaces with high availability and failover (like ODBC or JDBC connectivity). For example, there can be a situation where PowerBuilder could be calling the Oracle or Sybase database using DBLib (a PHP library to create WWW interfaces to databases). Types of libraries which do not exist now, or are not supported today, also need to be considered, and strategies prepared for them.
- **Application deployment:** Just like any software project, database migration requires proper testing, performance tuning, going live, and similar post-production activities. When a database is migrated, the front-end applications need to be connected with the new database. There are several considerations when deploying applications.
 - **Database connectivity:** Connectivity to the database can vary, based on business requirements or technical feasibility. There could be cases where database applications may or may not be using the Active Directory integration for authentication, which must be taken care of during migration or post-migration. Possible solutions could be to migrate to Active Directory (AD), or some alternative option depending upon the specific situation.
 - **User login and permission:** The authentication mechanism can be bound to the database in several ways. For example, there could be a scenario where the user connects to the application via Active Directory integration, then the application connects to the database tier. In this case, the application may have the user ID embedded to make an SQL call, or User ID may be passed as a parameter. Organizations may also use various other authentication mechanisms in their environments, like Windows groups, SQL Auth, Mixed Mode Auth, and Windows Auth, which need to be considered during migration.
 - **Performance tuning:** Currently, almost all databases like Oracle, DB2, Sybase, and SQL server possess roughly similar offerings, but the capabilities could be balanced differently across different products or features. For this reason, some fine-tuning would help to further improve the performance. When switching to SQL Server, there are several ways in which users can assess and optimize performance:

The Microsoft **SQL Server Distributed Replay** feature helps users assess the performance of the system by replaying the captured traces of any mission-critical workload against any new SQL Server environment. This way, users can check for the effect of running a workload in a given environment without having to actually deploy the entire workload in that environment.

SQL Server Extended Events allows users to record the event data (key execution points) generated by the SQL server and use it for various purposes, including application tracing (tools like SQL Trace and System Monitor) and logging (tools like Windows Event Log or SQL Server Error Log).

Within the SQL Server Management Studio (SSMS), there is **Query Execution Plan** feature, which graphically displays the data-retrieval methods used by the SQL Server query optimizer. This helps users understand the performance characteristics of the queries being executed.

There are several additional tools, like **SQL Profiler**, **Index Tuning Wizard**, and **SQL Query Analyzer**, which can be used to optimize the indexes and thus the performance of SQL Server.

Data migration using SQL Server Migration Assistant

SSMA overview

SSMA is a semi-automated tool which helps automate and manage the database migration process. It is a family of products, with specific tools for migrating databases from DB2, Oracle, Sybase, Access, and MySQL to SQL Server and Azure SQL Database. SSMA helps automate various aspects of database migration, including migration assessment, schema and SQL Statement conversion, data migration, and migration testing. This automation helps reduce the costs and risks involved in database migration, while providing better management and control over the entire migration process.

SSMA migration—Automation of all phases of database migration

SQL Server Migration Assistant helps automate the entire database migration process. Each component of SSMA helps in a specific phase of database migration. The key components and their functions are shown in Figure 5.

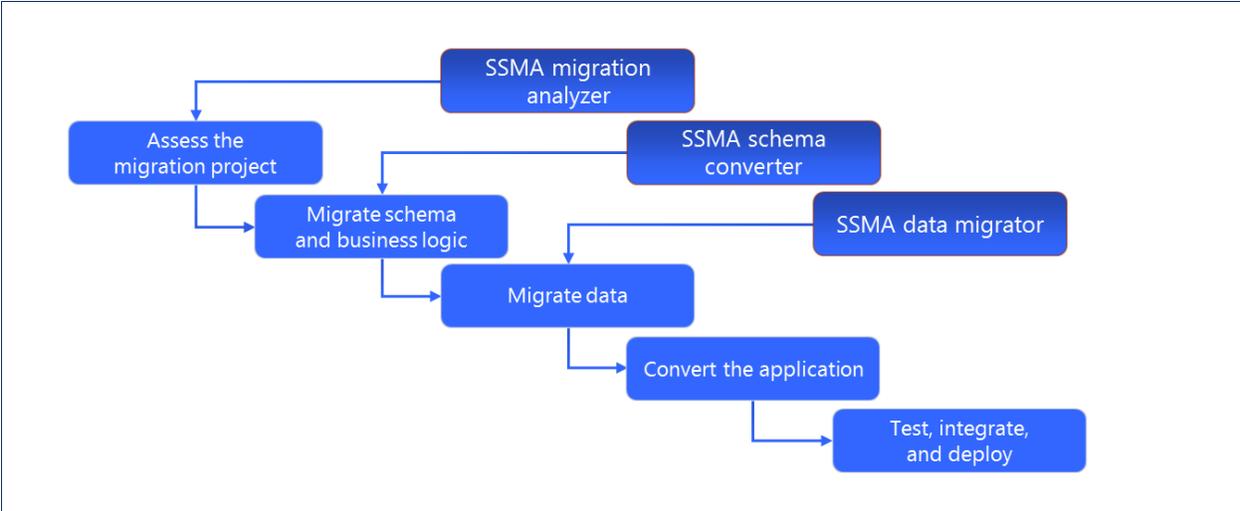


Figure 5. Key components of SSMA

- **Migration Analyzer:** SSMA Migration Analyzer performs an analysis of the source database objects and provides estimates of the time and effort required to migrate all the database objects to destination database (SQL Server or Azure SQL Database). It provides an estimate of the percentage of the database code that can be converted automatically. For the remaining code which it cannot migrate, it can provide an approximate estimate of the labor required to migrate the code manually. To make its assessments, the Migration Analyzer computes approximately 100 specific statistics that comprehensively characterize the database. An assessment usually takes less than a day to complete. The statistics provided by the Migration Analyzer can be saved in an external file in Excel-compliant format to enable the user to work with this data offline, if required.
- **Schema Converter and Data Migrator:** This component is associated with a powerful IDE for cross-platform database developers. It performs the actual database conversion. The Schema Converter usually converts more than 90 percent of the SQL code automatically, and there are instances where it has accomplished up to 99 percent of code conversion automatically. It provides detailed reports about database objects that it cannot convert, along with the specific constructs in the code that cause the problems.
- **Migration Tester:** This component performs migration testing by providing automated validation of the migration process. After all data migration steps are completed, SSMA Migration Tester can be used to verify that converted objects work properly and that all data was transferred properly. SSMA Tester can be used to test the following object types:
 - Tables
 - Stored procedures, including packaged procedures
 - User-defined functions, including packaged functions
 - Views
 - Standalone statements

SSMA and database migration considerations

SSMA helps with various aspects of data migration. SSMA Migration Analyzer performs a deep analysis to obtain estimates of the time and effort required to migrate a database. The SQL Converter helps automate code conversion from other databases to SQL Server.

- **Effort estimation:** SSMA Migration Analyzer can perform a thorough analysis of the source database (to be converted) to estimate the time and effort required to migrate the database. It can also provide details about the percentage of the database that can be converted automatically, and the estimated effort required to migrate the remaining database manually.
- **Schema conversion:** SSMA can automate the conversion of data types, constraints, views, indexes, sequences, table schema, and the like of any specific database to SQL Server data types. SSMA raises appropriate flags for any code-bits which it is not able to convert. For example, for conversion from Oracle scalar data types, it uses the following mappings:
 - varchar2 → varchar
 - char → char
 - number → numeric
 - date → datetime
 - long → text
 - nvarchar2 → nvarchar
 - boolean → smallint

- **Data migration:** Once the schema is ready at the destination database, SSMA can help to migrate the data as well. It migrates the data either one table at a time, or all tables at once. During data migration, the user should ensure that there are no constraints or triggers associated with a table which might prevent the records from being inserted.
- **Business logic:** SSMA can also help convert additional business logics implemented in a database, like triggers, constraints, outer joins, hints, stored procedures, functions, and packages.

Azure SQL Database service migration

- **Extension pack moved to a schema:** In addition to installing SSMA, you must also install database-specific extension packs on the computer that is running SQL Server. Extension packs identify specific code or functions in Oracle or any other language. They support data migration and database-specific providers to enable server-to-server connectivity. The SSMA extension pack adds the databases sysdb and ssmatesterdb to the specified instance of SQL Server. The database sysdb contains the tables and stored procedures that are required to migrate data, and the user-defined functions that emulate source database system functions (like Oracle). The ssmatesterdb database contains the tables and procedures that are required by the Migration Tester component.
- The extension packs are now moved to a schema level (as opposed to separate database), because Azure SQL Database cannot have multiple databases. For more details about using extension packs, see [https://msdn.microsoft.com/en-us/library/hh313165\(v=sql.110\).aspx](https://msdn.microsoft.com/en-us/library/hh313165(v=sql.110).aspx).

What's new in SSMA v6.0

SSMA version 6.0 provides many additional features, both at the functionality level and at the performance level. One important addition is support for Azure SQL Database and improved Azure SQL Database code conversion, which helps organizations attain their app modernization objectives much faster. Support for memory-optimized tables and “well known schemas” markup also helps increase the speed of conversion. The new version also includes support for larger schema migration, with additional enhancements like 25 percent smaller assessment reports, display of object count, and incremental schema load. The SSMA support functions are moved to the destination schema.

The SQL Server Migration Assistant product family includes the following products:

- **SSMA for Access:** SSMA for Access simplifies the database migration process from Access to SQL Server. It automates conversion of Microsoft Access database objects to SQL Server database objects, loads the objects into SQL Server and Azure SQL Database, and then migrates data from Microsoft Access to SQL Server and Azure SQL Database. SSMA for Access v6.0 is designed to support migration from Microsoft Access 97 and higher to all editions of SQL Server 2005, SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, SQL Server 2014, and Azure SQL Database. The Microsoft SQL Server Migration Assistant v6.0 tool for Access is available for download here: <http://www.microsoft.com/en-us/download/details.aspx?id=43690>.
- **SSMA for MySQL:** SSMA for MySQL simplifies the database migration process from MySQL to SQL Server and Azure SQL Database. It automates all aspects of migration including migration assessment analysis, schema and SQL statement conversion, data migration, and migration testing. SSMA for MySQL v6.0 is designed to support migration from MySQL 4.1 and higher to all editions of SQL Server 2005, SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, SQL Server 2014, and Azure SQL Database. The Microsoft SQL Server Migration Assistant v6.0 tool

for MySQL is available for download here: <http://www.microsoft.com/en-us/download/details.aspx?id=43688>

Separately, a detailed technical white paper on Microsoft SQL Server Migration Assistant v6.0 for MySQL is available for download here:

<http://download.microsoft.com/download/A/D/9/AD924E32-93E8-4141-B8BC-01399102DAFB/Migrating-MySQL-to-SQL-Server-2014-and-Azure-SQL-DB.pdf>.

- **SSMA for Oracle:** SSMA for Oracle simplifies the database migration process from Oracle to SQL Server and Azure SQL Database. SSMA automates all aspects of migration including migration assessment analysis, schema and SQL statement conversion, data migration, and migration testing. This product includes the GUI client–based application to manage the migration process as well as the SSMA extension pack to be installed on the destination SQL Server. The SSMA extension pack includes functionalities to emulate Oracle features not natively supported in SQL Server, a tester database to support SSMA Testing features, and an application to facilitate direct server-to-server data migration. SSMA for Oracle v6.0 is designed to support migration from Oracle 9.07.3 or later versions to all editions of SQL Server 2005, SQL Server 2008, SQL Server 2008 R2, and SQL Server 2012, SQL Server 2014, and Azure SQL Database. The Microsoft SQL Server Migration Assistant v6.0 tool for Oracle is available for download here: <http://www.microsoft.com/en-us/download/details.aspx?id=43689>

Separately, a detailed technical white paper on Microsoft SQL Server Migration Assistant v6.0 for Oracle is available for download here: <http://download.microsoft.com/download/F/C/A/FCA37580-53F5-46DB-A981-E08018661F09/Migrating-Oracle-to-SQL-Server-2014-and-Azure-SQL-DB.pdf>.

- **SSMA for Sybase:** SSMA for Sybase simplifies the database migration process from Sybase Adaptive Server Enterprise (ASE) to SQL Server and Azure SQL Database. SSMA automates all aspects of migration including migration assessment analysis, schema and SQL statement conversion, data migration, and migration testing. SSMA for Sybase v6.0 is designed to support migration from Sybase ASE version 11.9 and higher to all editions of SQL Server 2005, SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, SQL Server 2014, and Azure SQL Database. The Microsoft SQL Server Migration Assistant v6.0 tool for Sybase is available for download here: <http://www.microsoft.com/en-us/download/details.aspx?id=43691>

Separately, a detailed technical white paper on Microsoft SQL Server Migration Assistant v6.0 for Sybase is available for download here:

<http://download.microsoft.com/download/E/5/2/E522B67E-FA7F-4149-9C31-3EAD725B3A1D/Migrating-Sybase-ASE-to-SQL-Server-2014-and-Azure-SQL-DB.pdf>.

- **SSMA for DB2:** SSMA for DB2 simplifies the database migration process from DB2 to SQL Server and Azure SQL Database. SSMA automates all aspects of migration including migration assessment analysis, schema and SQL statement conversion, and data migration. This product includes a GUI client–based application to manage the migration process. The GUI client will install functionalities in SQL Server or Azure SQL Database to emulate DB2 features not natively supported in SQL Server or Azure SQL Database. SSMA for DB2 v6.0 is designed to support migration from DB2 on z/OS versions 9.0 and 10.0 and DB2 on Linux/Unix/Windows (LUW) versions 9.8 and 10.1 to SQL Server 2012, SQL Server 2014, and Azure SQL Database. The Microsoft SQL Server Migration Assistant v6.0 tool for DB2 is available for download here: <http://www.microsoft.com/en-us/download/details.aspx?id=45296>.

Separately, a detailed technical white paper on Microsoft SQL Server Migration Assistant v6.0 for DB2 is available for download here: <TBD>

SSMA engineering and support

The [SQL Server Resources](#) provide important information about migration from other relational databases to SQL Server. Use this site to download the latest versions of SSMA, find links to migration-related discussion forums, and read the latest information shared by the SSMA team on their blogs. The SSMA product team is available to answer your questions and provide limited technical support. Contact the team at ssmahelp@microsoft.com.

In addition to the previously listed resources, you can ask a question in the SQL Server community or request help from Microsoft support. The following table links to and describes these resources:

Resource	Description
SQL Server Community	Newsgroups and forums that are monitored by the SQL Server community. This site also lists community information sources, such as blogs and websites. The SQL Server community can help answer questions, although there is no guarantee of an answer.
SQL Server Developer Center Community	This site focuses on the newsgroups, forums, and other community resources that are useful to SQL Server developers.
Microsoft Help and Support	You can use this website to open a case with a Microsoft support professional.
SSMA MSDN Forum	This site focuses on the queries or discussions related to migrating to SQL Server from other databases.
SSMA Blogs	SQL Server Migration Assistant blog is a community to discuss SQL Server Migration using various SSMA tools.

Conclusion

Application and data platform modernization are considered to be among the top enterprise IT initiatives. Organizations often seek to migrate toward a better and more competitive platform like SQL Server. This can help them break the traditional barriers of legacy databases and deliver mission-critical performance with faster insights and support for cloud-based platforms.

However, migration to a modern platform is usually considered a complex task and requires a well-defined methodology and approach to ensure a timely and efficient execution of the platform-transition process. SQL Server Migration Assistant, a data migration tool from Microsoft, efficiently handles the complexities involved in the migration process and helps to automate and manage database migration. SSMA helps organizations migrate their applications and databases from other vendors (including Access, DB2, MySQL, Oracle, and Sybase) to SQL Server in an efficient manner. It helps speed the entire process by automating almost every aspect of database migration, including database assessment, schema and application migration, data migration, and validation of the migrated database. This helps reduce the time, cost, and risks associated with migrating to SQL Server. With SSMA, customers do not have to remain tied to an expensive, sub-optimal database solution.

Key resources

- **Platform modernization:** <https://www.platformmodernization.org/Pages/default.aspx>
- **SSMA blog:** <http://blogs.msdn.com/b/ssma/>
- Free download of the latest **SQL Server Migration Assistant:**
 - SSMA for Access: <http://www.microsoft.com/en-us/download/details.aspx?id=43690>
 - SSMA for MySQL: <http://www.microsoft.com/en-us/download/details.aspx?id=43688>
 - SSMA for Oracle: <http://www.microsoft.com/en-us/download/details.aspx?id=43689>
 - SSMA for Sybase: <http://www.microsoft.com/en-us/download/details.aspx?id=43691>
 - SSMA for DB2: <http://www.microsoft.com/en-us/download/details.aspx?id=45296>
- Free download of detailed technical white papers on **SQL Server Migration Assistant:**
 - SSMA for Oracle: <http://download.microsoft.com/download/F/C/A/FCA37580-53F5-46DB-A981-E08018661F09/Migrating-Oracle-to-SQL-Server-2014-and-Azure-SQL-DB.pdf>
 - SSMA for Sybase ASE: <http://download.microsoft.com/download/E/5/2/E522B67E-FA7F-4149-9C31-3EAD725B3A1D/Migrating-Sybase-ASE-to-SQL-Server-2014-and-Azure-SQL-DB.pdf>
 - SSMA for MySQL: <http://download.microsoft.com/download/A/D/9/AD924E32-93E8-4141-B8BC-01399102DAFB/Migrating-MySQL-to-SQL-Server-2014-and-Azure-SQL-DB.pdf>
 - SSMA for DB2: <TBD>

For more information:

<http://www.microsoft.com/sqlserver/>: SQL Server Web site

<http://technet.microsoft.com/en-us/sqlserver/>: SQL Server TechCenter

<http://msdn.microsoft.com/en-us/sqlserver/>: SQL Server DevCenter

Did this paper help you? Please give us your feedback. Tell us on a scale of 1 (poor) to 5 (excellent), how would you rate this paper and why have you given it this rating? For example:

- Are you rating it high due to having good examples, excellent screenshots, clear writing, or another reason?
- Are you rating it low due to poor examples, fuzzy screenshots, unclear writing?

This feedback will help us improve the quality of the white papers we release.

[Send feedback.](#)