datAvail

DATABASE SERVICES

SQL UPGRADE CHECKLIST

Either type of SQL Server upgrade – in-place or side-by-side – is a serious project with many considerations. A smooth, successful upgrade relies on good planning.

With that in mind, here are some tips you will want to follow in order to make that transition to a new version of SQL Server.

Pick a target: Consider the upgrade type, licensing, operating systems, preferred features and so on to determine the target version and edition of SQL Server and the OS.

Read up: Review all the material about your upgrade published by Microsoft as you can. This may include the upgrade advisor and its "breaking changes" and "behavioral changes" for your version of SQL Server published in the Microsoft Books Online. The latter publication might be more up-to-date than the upgrade advisor.

Baseline: Catalogue resources and resource utilization on the legacy server – get details, or at least a general sense of the current resource use space. This will help in provisioning your upgrade target. Provision your hardware (or perhaps cloud-ware.) Use the baseline as well as other requirements (SQL Server, OS and any other software) to determine your specific hardware needs. Ensure the features you plan to use – SQL or otherwise - have the proper caliber hardware to support them.

Choose a migration method: If it is an in-place upgrade this is moot, otherwise consider detach/attach, backup/restore, log shipping, database mirroring, or another method to move the data from the legacy server to the new instance. Define how you will redirect the application and/or user connections to the new server: change connection strings, DNS, etc. If the target is already up and running, this is a good time to test moving the data and perhaps begin testing the workload on the new server.

Dial in the target: The time to ensure the new server is up to industry standards and all the best practices are in place, is before the upgrade/migration. Script-out legacy server-level objects (jobs, logins, linked server, etc.) and apply them to the new instance (you may need to bring them up to date again at migration time.) Run a health check or the Microsoft Best Practice Analyzer and review the suggestions. Capture the workload on the old system and replay it on the new server using SQL Trace or similar tools. This gives a preview of performance and may bring issues to light – for instance, cases where the new cardinality estimator is detrimental to query performance.

ORACLE DATABASE | ORACLE E-BUSINESS SUITE | Microsoft SQL Server | MySQL | mongoDB | IBM DB2 | SharePoint

SQL UPGRADE CHECKLIST

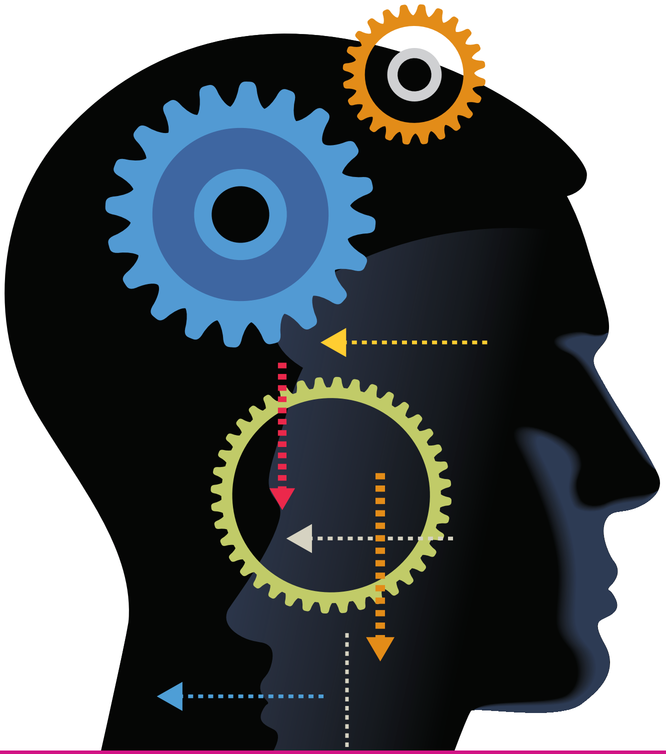
datAvail
DATABASE SERVICES

Do some house-cleaning: As much as possible, bring the old server up to date. Run the Database Upgrade Advisor and review the suggestions. Run a health check or the Microsoft Best Practice Analyzer and review the suggestions. Test the selected migration method with a small testdb – think through any "show-stoppers" that might slow or stop the upgrade. Consider techniques for migrating very large databases if relevant. Reduce high VLF counts. Ensure recent backups. Verify database compatibility levels are as desired and ready for migration. In short, do as much prep work as possible so when the databases are migrated to the new instance they are ready for production.

Document the plan: Write out a detailed step-by-step plan for the actual migration. Include roll back options just in case.

Migration: Follow your plan.

After the migration: Backup databases including system databases. Ensure server logins are mapped to database users (sp_change_users_login.) Run update usage and update statistic commands (DBCC UPDATEUSAGE, UPDATE STATISTICS.) Check database consistency (DBCC CHECKDB.) Run a Datavail Health Check and the Microsoft Best Practice Analyzer and review the suggestions.



If you are still using SQL 2005 or SQL 2008, migrating to SQL 2012 or 2014 may seem a daunting task. Technologies have changed significantly and the gap will expand even more once we reach SQL 2016. We explain some of the rationales and reasons for upgrading in our white paper titled *Upgrading SQL Server to 2012 and Beyond*. We have more detail on upgrade and migration methods on our blog.

Datavail can support your organization's transition to the newest version of SQL Server, regardless of the version you ultimately select. Contact Datavail to discuss a custom SQL Server solution designed for your enterprise and to learn more about how our experts can help with your ongoing SQL Server operations.

SUPPORTED PLATFORMS

Oracle
Microsoft SQL Server
DB2
SharePoint
Oracle EBS
MySQL
MongoDB