Corporate Headquarters
1203 Crestside Drive , Suite 300
Coppell, TX 75019 USA
Toll-free: (866) 304-8221
Phone: (469) 635-6200
Fax: (469) 635-6206

SCALABILITY experts

idera™ TOOLS FOR SQL SERVER™

802 Lovett Boulevard
Houston, TX USA 77006
Toll-free: 1.877.GO.IDERA
Phone: 713.523.4433
Fax: 713.862.5210
www.idera.com

TOP 10 ESSENTIAL TIPS FOR UPGRADING TO MICROSOFT SQL SERVER 2005

1. **IDENTIFY REQUIREMENTS & DETERMINE BEST TIME FOR UPGRADE** — Prior to upgrading, identify business requirements for availability and performance during the upgrade process. Determine how much downtime your business can accept during the upgrade. This will help you choose the best upgrade approach.

2. **DETERMINE UPGRADE MODE** — Study the different modes of upgrading such as in-place upgrades and side-by-side upgrades for the relational engine and other SQL Server components (e.g., DTS and Analysis Services). Based on your business goals and availability, choose the appropriate mode of upgrade and determine the best time to perform the upgrade.

3. **INVENTORY CURRENT ASSETS** — Perform an Architectural Design Review to study your current SQL Server environment and understand your hardware and software resource utilization. Make an inventory of the versions of SQL Server and sub components installed in your environment that you need to upgrade. List all applications and legacy systems that will be affected, including compatibility issues and complexity.

4. **SELECT 2005 EDITION** — Understand the differences between the editions of SQL Server 2005 such as Standard, Enterprise, Workgroup and Express and choose an edition that best fits your business requirements.

5. **EXECUTE UPGRADE ADVISER** — Run the Microsoft Upgrade Advisor tool to identify issues that will cause upgrade to fail. In addition, the tool will identify compatibility issues with running the application T-SQL code in SQL Server 2005. Address all of these issues.

6. **BUILD VERIFICATION SCRIPTS AND DRY RUN** — As part of the pre-upgrade checklist, generate appropriate procedures, scripts and test cases to validate the correctness of the SQL components' configuration and data after upgrade. In addition, before upgrading the SQL Server components in production, perform a mock upgrade in the test environment to resolve any issues that can prevent successful upgrade.

7. **PREPARE SQL SERVER FOR UPGRADE** — Before starting the upgrade, make sure that the SQL Server system is ready for update by turning off all the replication, jobs, trace flags and start-up stored procedures. Check the consistency of the user and system database by running the appropriate DBCC commands.

8. **PREPARE DISASTER RECOVERY PLAN** — Before upgrading, prepare a Disaster Recovery plan to roll back in case of an unsuccessful upgrade. Back up all your database files and if necessary, archive the analysis services data. Script out the database schema and its associated objects, replication settings and database configurations. Save all the scripts and metadata of your DTS packages and Analysis Services in a safe place. Deploy the recovery plan in the testing environment to ensure that you can roll back successfully if needed.

9. **VERIFY PERMISSIONS** — Check to ensure that the administrator has the appropriate permissions to install and upgrade SQL Server and all its subcomponents on the different machines.

10. **EXECUTE AND VERIFY UPGRADE** — Once the upgrade is complete, check the database compatibility level and the configuration of the database components. After validating your upgrade, tackle the post-upgrade issues as advised by the Update Advisor tool and components which cannot be directly upgraded or migrated. Run the sp_updatestats to update the statistics for good performance.

SUMMARY: TOP 10 TIPS

1. Determine your Business Goals and Requirements
2. Understand and pick the Method of Upgrade.
3. Perform an Architectural Design Review
4. Decide on the appropriate SQL Server Edition.
5. Use Upgrade Advisor to determine issues with upgrading.
6. Create a Test Procedure to validate the components after upgrade.
7. Ensure that SQL Server is in a Consistent State for update.
8. Prepare a Disaster Recovery Plan to roll back if necessary.
9. Grant appropriate Permissions to the administrator.
10. After the upgrade, validate the components upgraded by the tools and manually upgrade the rest.