Best Maintenance Tools for PTC Windchill on Microsoft SQL Server 2008

Production-proven content and process management software

Microsoft Corporation
Published: May 2010

Author: Ken Lassesen
Reviewers: Tim Atwood (PTC), Victor Gerdes (PTC), Richard Waymire (Solid Quality Mentors)

Abstract
PTC Windchill is the only Product Lifecycle Management (PLM) solution designed from the ground up to work in an Internet-based, distributed-design environment. Whether you need core product data-management capabilities, optimization of processes to meet industry-specific requirements, or support for global product development, Windchill has the capabilities you need for effective management of global product development teams. Microsoft® SQL Server® provides an ideal database platform for Windchill. With SQL Server as a foundation, Windchill can further reduce the time and costs related to managing product development.

This white paper provides best practices for configuring and running Windchill on the SQL Server database platform. The information in this paper complements the detailed support documentation provided on the PTC support Web site. Implementing these best practices can help you avoid or minimize common problems and optimize the performance of Windchill on SQL Server so that you can effectively manage your resources, reduce operating expenses, increase productivity, and improve employee satisfaction.
Copyright Information

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.

© 2010 Microsoft Corporation. All rights reserved.

Microsoft, SQL Server, Hyper-V, MSDN, and Windows are trademarks of the Microsoft group of companies. All other trademarks are property of their respective owners.
# Table of Contents

**OVERVIEW** ................................................................................................................................. 4

- **Introducing PTC Windchill: A Complete Solution for PLM** .......................................................... 4
- **SQL Server: An Enterprise-Ready Database Platform for Windchill** ........................................ 5
- **PTC and Microsoft: A History of Innovation** ............................................................................. 5
- **Better Together: Windchill on the SQL Server Database Platform** ............................................ 5

**BEST MAINTENANCE TOOLS FOR RUNNING WINDCHILL ON SQL SERVER** ......................... 6

- **Basic Maintenance Tools** ........................................................................................................... 6

**SQL SERVER 2008 MANAGEMENT STUDIO** ............................................................................... 7

- **Database Tuning Advisor** ........................................................................................................... 7
- **SQL Server Profiler** ................................................................................................................... 7
- **Creating a Workload Using SQL Server Profiler** ....................................................................... 8
- **Running a DTA Tuning Session** .................................................................................................. 10
- **Reviewing a DTA Tuning Session** .............................................................................................. 14
- **Index Selection Criteria** ........................................................................................................... 16
- **When to Run Additional Tuning Sessions** .................................................................................. 17
- **Management Data Warehouse (MDW)** .................................................................................... 17
  - **Installing MDW** .................................................................................................................... 18
  - **Built-in Reports** ..................................................................................................................... 26
  - **Advanced Troubleshooting and Analysis** ................................................................................. 29

**SQL SERVER 2008 BEST PRACTICES ANALYZER (BPA)** .............................................................. 29

- **Expected Windchill Exceptions** .................................................................................................. 32
- **Further Securing Your Windchill Database** ............................................................................... 32

**SUMMARY** ..................................................................................................................................... 33

**LINKS FOR FURTHER INFORMATION** ......................................................................................... 36

**TABLES**

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>17</td>
</tr>
<tr>
<td>Table 2</td>
<td>32</td>
</tr>
<tr>
<td>Table 3</td>
<td>32</td>
</tr>
<tr>
<td>Table 4</td>
<td>33</td>
</tr>
<tr>
<td>Table 5</td>
<td>34</td>
</tr>
<tr>
<td>Table 6</td>
<td>34</td>
</tr>
</tbody>
</table>

**FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>27</td>
</tr>
<tr>
<td>Figure 2</td>
<td>28</td>
</tr>
<tr>
<td>Figure 3</td>
<td>28</td>
</tr>
</tbody>
</table>
Overview

For manufacturers, there is constant pressure today to compete in rapidly changing markets. Companies must be innovative in keeping costs low while maintaining high-quality results. The stakeholders involved in the product development process have also changed. Teams today extend far beyond the central engineering department to include globally dispersed cross-functional groups working on hundreds of products with thousands of parts. The result is an enormous amount of data.

To stay ahead of the global competition, companies find that they must create a collaborative environment that brings together engineering, manufacturing, marketing, and sales teams. Product Lifecycle Management (PLM) solutions can help manufacturers achieve this collaboration, while also streamlining operations and keeping costs down.

PLM is the process of managing all phases of product development—from initial concept through end of life. Effective PLM combines information, methodology, and available resources for each phase of a product’s lifecycle, improving a manufacturer’s ability to respond swiftly and effectively to changes, new markets, and competitors.

Introducing PTC Windchill: A Complete Solution for PLM

PTC Windchill provides a complete family of solutions for content and process management, helping manufacturers efficiently control all information assets related to product development while optimizing associated business processes. Windchill is the only PLM solution designed from the ground up to work in an Internet-based, distributed-design environment.

Windchill technology forms a solid foundation for a variety of packages that PTC offers to address data, change, configuration, and process management; product development collaboration; project management and execution; and the release of product information to manufacturing management systems. For example,

- **Windchill PDMLink** consolidates scattered islands of product content into a single information source, which can help bring order to chaotic product development processes such as change management and speed the development of new product configurations.
- **Windchill ProjectLink** creates a virtual workspace that becomes the central access point for a project, enabling team members to collaborate with access to the same information. By automating project management activities, Windchill ProjectLink helps customers better manage all of their programs, project schedules, information, and processes.

As an integral component of PTC’s Product Development System (PDS), Windchill manages all product content and business processes throughout a product’s lifecycle. Windchill connects seamlessly to Pro/ENGINEER for three-dimensional (3-D) computer-aided design (CAD) models, ProductView® for advanced mock-up and interactive visualization, Mathcad® for engineering calculations, and Arbortext® for dynamic publishing.
SQL Server: An Enterprise-Ready Database Platform for Windchill

Microsoft SQL Server provides an ideal database platform for Windchill. SQL Server is a high-performance, integrated database and business intelligence (BI) solution for data management and analysis. This easy-to-implement, easy-to-support foundation provides a multifunctional solution for large-scale online transaction processing (OLTP), data warehousing, and e-commerce applications and a solution for data integration, analysis, and reporting.

SQL Server can help companies manage large volumes of mission-critical data and run software applications—such as PTC Windchill—to optimize their business performance. SQL Server can extract and transform data from a variety of sources, including XML data files, flat files, and relational data sources, and then load it into one or more destinations. In addition to rapid data mining, analysis, processing, and reporting capabilities, SQL Server has built-in features that give you a secure, reliable, and productive data management environment that truly protects your data. With its scalable infrastructure, SQL Server has the capability to grow with your business and keep up with your toughest data challenges.

PTC and Microsoft: A History of Innovation

PTC and Microsoft deliver complementary product development solutions that organizations can use broadly across their infrastructure. With a Microsoft IT infrastructure, you get an open, extensible platform and a simplified user experience. You can take advantage of your existing IT investments for a lower total cost of ownership (TCO). PTC provides full integration with your Microsoft infrastructure, including Microsoft® Office SharePoint® Server, Microsoft® Office Project Server, Microsoft® Office Communications Server, Windows Server®, and SQL Server. The PDS architecture gives you end-to-end solutions for product development and a single source for product and process knowledge.

Together, PTC and Microsoft deliver powerful product development solutions that provide tremendous customer value.

Better Together: Windchill on the SQL Server Database Platform

Running Windchill on SQL Server delivers measureable value by channeling data into manageable, automated processes. This decreases administrative time, improves productivity, reduces costs, and generates greater employee satisfaction. Benchmarking tests confirm that SQL Server scales to meet the performance needs of even the largest enterprise customers, while providing lower initial costs and licensing fees.¹

Benchmark test results showed that Pro/ENGINEER and Windchill 9.1 performed up to 50 percent faster on SQL Server 2005 than on a competitor’s database, with an average performance advantage of approximately 10 percent. These results confirm SQL Server 2005 as a superior database choice for Windchill 9.1. Enhancements in SQL Server 2008 improve performance with Windchill 10.0 even further.

Best Maintenance Tools for Running Windchill on SQL Server

This paper looks at tools that automate basic best practices described in the companion white paper “Best Practices for PTC Windchill on Microsoft SQL Server.” These tools often provide additional analysis and capabilities not covered in the “Best Practices” white paper.

Best maintenance practices usually include the following aspects:

- Corrective maintenance, such as replacing a suspect or failing drive
- Preventative maintenance, such as backing up databases and storing them off-site
- Predictive maintenance, such as reorganizing files during maintenance windows or adding disks before running out of disk space
- Purchasing, such as ensuring that spares for all critical elements are on site or available from a vendor within an acceptable Service Level Agreement (SLA)
- Proactive maintenance, which is a combination of preventative maintenance and predictive maintenance; an example is looking at the expected upper and lower bounds for growth of data files and users in each budget cycle and obtaining appropriate capital budget for hardware and/or software, such as upgrading a server from SQL Server Standard Edition to Enterprise Edition
- Accountability, such as ensuring that someone is designated and available to deal with issues at all times

Remember that you need to validate best practices using current standards. Best maintenance practices for Windchill on SQL Server 2005, SQL Server 2000, or Oracle may not be recommended or relevant for SQL Server 2008. New software and operating system (OS) patches may also alter the maintenance tasks you need to do.

This white paper focuses on SQL Server performance monitoring, maintaining a healthy database, and maintaining a secure system by using tools provided by Microsoft. For more information about SQL Server administration, see the bibliography at the end of this paper.

Basic Maintenance Tools

The tools discussed in this white paper are all free from the Microsoft Web site or part of SQL Server 2008 Installation package.

PTC recommends the installation and use of the following Microsoft or community products:

- Microsoft SQL Server 2008 Management Studio (SSMS)\(^2\), which includes:
  - SQL Server Database Tuning Advisor
  - SQL Server Profiler
  - Management Data Warehouse
- Microsoft SQL Server 2008 Best Practices Analyzer

\(^2\) SSMS installs with SQL Server as an option. SSMS may also be installed without SQL Server.
Using these tools can greatly reduce the time needed to do maintenance, especially when compared to manually performing all the best practices tasks required to keep your system in top shape.

### SQL Server 2008 Management Studio

SSMS is an easy-to-use graphical management tool for SQL Server 2008. This tool includes script editors, graphical tools, and a collection of utilities that can assist with troubleshooting and tuning. To tune your Windchill database, you can use the following utilities:

- Database Tuning Advisor (DTA)
- SQL Server Profiler
- Management Data Warehouse (MDW)

SSMS also provides a wizard to automate maintenance tasks.

### Database Tuning Advisor

A database changes its nature over time and needs to be tuned for optimum performance. The DTA is a wizard utility built into SSMS that uses a query optimizer to:

- Recommend the best mix of indexes for databases
- Recommend aligned or non-aligned partitions for databases
- Recommend indexed views for databases
- Analyze the effects of the proposed changes, including index usage, query distribution among tables, and query performance of a workload
- Provide reports that summarize the potential effects of implementing the recommendations for a given workload

The DTA also lets you customize its recommendations by specifying advanced options such as disk space constraints. (See [Database Engine Tuning Advisor Overview](#) and [Database Engine Tuning Advisor Features](#) for more information about using the DTA.)

Usage patterns and loads for Windchill can vary greatly between customers. This variability prevents extensive pre-tuning of the database before installation. In addition, after you first install Windchill, an atypical workload occurs from loading documents and establishing projects. However, after you have the database loaded and an ongoing pattern of additions, updates, and deletes is established, you have the opportunity to run the DTA and do some tuning.

The key to doing effective tuning is to use SQL Server Profiler with the Tuning template to capture a workload for the DTA to analyze. This method of creating workloads is the simplest and works well for capturing ad hoc workloads.

### SQL Server Profiler

SQL Server Profiler is a standalone tool that captures SQL Server events from a server and saves them to a trace file. You can then replay the events and analyze them to see what is happening on your server. As we noted above, the DTA can also use a captured workload to evaluate the impact of additional indexes on queries.
Let’s look at how to use Profiler and the DTA to perform the following tasks:
- Create a workload
- Run a tuning session
- Determine what recommendations to implement
- Determine when to implement those recommendations

Remember that there can be a great variety of workloads among Windchill customers and installations, so to get the best performance for your specific workload, you need to tune each instance of Windchill separately. Fortunately, this is an easy process using tools such as the DTA.

A best practice is to run SSMS, the DTA, and Profiler from your desktop machine and not on the machine hosting SQL Server.

Creating a Workload Using SQL Server Profiler
You should capture your workload when the system is busy. The captures should include at least these three types of workloads:
- “Opening the week,” which is the first two business hours on a Monday. This period will give you a profile dominated by people checking out documents.
- “Closing the week,” which is the last two business hours of a Friday. This period will be dominated by people checking in or rolling back documents.
- “Typically busy,” which often means between 10-11am or 2-4pm on Tuesdays, Wednesdays, or Thursdays. This period will be a mixture of check-in and check-out activity, as well as other typical Windchill operations such as searching, downloading, running reports, and checking assigned tasks.

If there are other times when the workload is atypical then captures at those times is advised. Some possible examples: closing of a contract, end of year, start of contract, end of month.

The steps for creating a workload are as follows:
1. Open SSMS.
2. On the menu, click Tools and then SQL Server Profiler to open the Profiler application.
3. The resulting Connect to Server dialog box lets you log in to the Windchill instance.
4. From the Trace Properties dialog box that appears, click the General tab and select the Tuning template from the drop-down list.

   ![Microsoft SQL Server 2008 Connect to Server dialog box](image)

   a. You have two choices for where to save the data: *Save to file* or *Save to table*.

   PTC recommends saving to a file, which has the advantage of being easy to forward to PTC for further analysis. Saving to a file also removes load from SQL Server so that your production environment is minimally impacted.
b. To set the end time for the trace, select the *Enable trace stop time* check box and enter the time in the controls to the right of the check box.

5. Next, you need to set the Maximum File Size of the trace. PTC recommends 1GB as a starting point; you can then increase or decrease the file size with experience.
   a. Do not do detail tracing. A single Windchill operation can result in a 2GB trace log. PTC recommends always limiting the maximum file size to significantly less than the available space on the drive.
   b. If the Tuning template produces too much data, you may want to create your own template starting with captures of the following events only:
      - RPC:Completed
      - SQL:BatchCompleted
      - SP:StmtCompleted
   b. For information about building a custom template, see [How to: Create Workloads](#).

6. You now need to determine when you expect the heaviest load in the period. Do you expect it at the beginning or the end of the trace period? The trace file may run out of space, so you need to determine what information you want retained.
   a. If you expect the heaviest load toward the end, select Enable File Rollover.
   b. If you expect the heaviest load at the beginning, deselect the default Enable File Rollover.
   c. If unsure, leave this option unchecked and revisit later.

7. Leave the Event Selection tab with the default values.

8. Click Run. That is it!

If you follow the above steps, it should take you only a couple of minutes to define a trace that will give you the workload you want to analyze. Profiler will automatically record the activity until you stop the recording or reach the specified end time. There is no need to monitor the capture.

Two additional notes to keep in mind about Profiler:
- The longer the trace captures data, the bigger the file produced (up to the limit specified). Use your first run to estimate MB/minute of data capture for future runs of Profiler tracing.
- The more queries captured by your Profiler trace, the better results you will get using the trace data with the DTA utility.

For more information about using Profiler, see
- [How to: Create a Trace (SQL Server Profiler)](#)
- [How to: Correlate a Trace with Windows Performance Log Data (SQL Server Profiler)](#)
- [Mastering SQL Server Profiler – Video Training Course](#)
- [Microsoft’s Patterns & Practices: How to: Use SQL Profiler](#)
- [Running Profiler on a Busy SQL Server](#)
- [SQL Server 2008 Performance Tuning Webcast](#)
- [SQL Server Profiler Tutorial Video](#)

**Running a DTA Tuning Session**

With a workload captured to a file in Profiler, your next step is to use that file as input to the DTA to determine any tuning recommendations. Preparing the DTA session also takes less than
2 minutes, but the DTA will require some time to perform its analysis and return recommendations.

Follow these steps for defining a DTA tuning session:

1. Open SSMS.
2. On the menu, click Tools and then Database Engine Tuning Advisor to open the application.
3. The Connect to Server dialog box will appear. Enter the appropriate login information and connect.
4. The Database Engine Tuning Advisor will then appear.
5. Select the database you want to tune (wcAdmin in this example) and select the Workload file you created. Note that selecting the Windchill (wcAdmin) database will automatically select all 763 tables in Windchill.
a. Next, click the Tuning Options tab. For your first run, select the Indexes, *No partitioning*, and *Keep all existing PDS* radio buttons.

6. Click Advanced Options. Verify the default settings of *All recommendations are offline* and *Max. Columns per index: 1023*, then click OK.

7. Now estimate how long this machine is going to be unused. Typically, running the DTA is an end-of-day activity, so set the time to an hour before you come into work.

8. You’re now ready to click Start Analysis.
9. The screen will immediately start filling up with the captured activity.

10. You will notice a new Progress tab in the DTA. The process can take several hours to complete if the workload contains a large amount of transactions.

11. When the tuning calculations are complete, you will see two additional new tabs: Recommendations and Reports. Make sure you have a latte or equivalent in hand—you will be looking at the results for a while.

**Reviewing a DTA Tuning Session**

Here are some recommendations for reviewing a DTA tuning session.

1. Click the Reports tab to see a summary of the tuning session.
2. At the bottom of the tab, click the Select Report combo box to see a list of available reports.

3. Traverse each of the reports to see what the DTA found. For example, the Statement cost report shows how much performance improvement you may see for various statements (in this example, a few statements are predicted to improve by 94%!)

<table>
<thead>
<tr>
<th>Statement Id</th>
<th>Statement String</th>
<th>Percent Improvement</th>
<th>Statement Type</th>
</tr>
</thead>
</table>
4. Now click the Recommendations tab to see the DTA’s index recommendations.

   a. Scroll to the right until you see the Definition column.
   b. You can then select the index or indexes you want to implement. (The next section looks at how to determine which index recommendations to accept.)

**Index Selection Criteria**

If you are familiar with the database, you can use that knowledge to determine the significance of various indexes. Alternatively, you could simply select all of the DTA’s index recommendations—but be aware that indexes take resources to build and maintain.

If you have no detailed knowledge of the database, its tables, and their relationships, use the following heuristic to help determine which indexes to implement:

- Find the indexes that have the **smallest number of columns** (on the first pass, it will be one column only). In the above example, we have eight (8) indexes that have only one column out of the 17 suggestions, with a net improvement of 4.7%.
- Select those indexes and then click Actions, Apply Recommendations.
- Rerun the workload to see how much the added indexes improved performance. Do not expect the same index recommendations from two different trace logs. The estimations derive from the index statistics used for each query and the number of times a query occurs.

Adding an index may decrease the contribution of an index recommendation from another workload (or the same workload). Existing indexes and index recommendations are often dependent, frequently with complex interactions.

A good rule of thumb is that if the DTA yields a 5% improvement, then add the indexes with the **lowest number of columns** and repeat the workload measurements until the improvement drops below 5%.

**Note:** The DTA will always produce recommendations for **adding** indexes and never for deleting indexes. See **Uncover Hidden Data to Optimize Application Performance** for information about detecting unused or costly indexes, which you may choose to delete. Remember that an unused index consumes resources and results in poorer performance.
This tuning process may span several days, but the time involved should be less than 10 minutes a day. Run the DTA before you leave the office, and review the results and select the new indexes when you come in the next day. The payback to your organization is that users will have faster access to the data they need to do their jobs.

Caution:
- You should not run the DTA when there is any load on the system because the process may place considerable load on SQL Server. Execute the DTA during off hours or maintenance periods.
  - If you have a recent backup copy of the database available then use the backup for the analysis. You will need to manually apply the recommendations to the production database.
- If the size of any of the recommended indexes is large, make sure that you have adequate available disk space to add them. There is nothing like adding an index and having a drive run out of space during the work day!

When to Run Additional Tuning Sessions
If there is a change in your business—for example, ramping up of a new major contract—there may be a significant change in the load on the database. Often this is a good time to perform a tuning. In the course of normal operations, PTC recommends executing the DTA as shown in Table 1 - Recommended Frequency of DTA Execution.

Table 1 - Recommended Frequency of DTA Execution

<table>
<thead>
<tr>
<th>Time Since Installation</th>
<th>Frequency</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>For first 6 weeks</td>
<td>Weekly</td>
<td>Run DTA with default settings</td>
</tr>
<tr>
<td>First year</td>
<td>Monthly</td>
<td>Run DTA with indexes and index views; include filtered indexes</td>
</tr>
<tr>
<td>After 1 quarter</td>
<td>Quarterly</td>
<td>Run DTA with full partitioning</td>
</tr>
<tr>
<td>After 1 quarter</td>
<td>Quarterly</td>
<td>Run DTA with indexes and index views; include filtered indexes</td>
</tr>
</tbody>
</table>

Note: There is rarely a need for full partitioning. For information about rebuilding indexes, see “Best Practices for PTC Windchill on Microsoft SQL Server.”

Management Data Warehouse (MDW)
MDW captures key Windows Performance Monitor (PerfMon) counters and internal SQL Server counters every minute into a SQL Server database. Running MDW continuously results in proactive detection of issues and analysis of reported problems. Prior to MDW, a DBA would typically receive a report of a problem, start running PerfMon, and hope that the problem reoccurs for analysis. With MDW, you effectively have PerfMon always running (at a low sampling rate); you just need to know when the problem occurred to do in-depth analysis.

MDW incorporates many features found in tools targeted for SQL Server 2005, including
- Performance Analysis of Logs (PAL) Tool
- SQL Server 2005 Performance Dashboard Reports
Installing MDW

Because MDW is fairly new, let's walk through it step by step.

1. In SSMS, open Management, Data Collection and then right-click and select Configure Management Data Warehouse.

2. If there is a Welcome screen, click past it to arrive at the Select configuration task dialog box. Select Create or upgrade a management data warehouse and then click Next.
3. On the Configure Management Data Warehouse Storage screen, click New.

4. When the New Database dialog box appears, enter a suitable name. For example if you are setting up a database on a [WcAdmin] database for 2010, enter wcAdminMDW2010 as the database name and click OK.
5. Back at the Configure Management Data Warehouse Storage screen, click Next.

6. The Map Login and Users dialog box will appear. Assuming you are an administrator, there is no need to add a mapping. Click Next.
7. At the Complete the Wizard screen, click Finish.

8. Next, you will see the Configure Data Collection Wizard Progress screen.
9. In the SSMS Object Explorer, notice that a new database (wcAdminMDW2010) has been added.
10. Now, let’s return to Step 1: Open Management, Data Collection and then right-click and select Configure Management Data Warehouse. This time, however, select *Set up data collection* and click Next.
11. On the Configure Management Data Warehouse Storage dialog box, select the new database and click Next. (Note: You must click the Server name ellipsis box (…) to enable the Database name combo box.

12. At the Complete the Wizard screen, click Finish.
13. The Configure Data Collection Wizard Progress screen will appear. When it is finished, click Close.

14. Return to SSMS, and right-click to see new items under Management, Data Collection.
15. Under SQL Server Agent, Jobs, you will see that a new job—mdw_purge_data_[wcAdminMDW2010]—has been created. This job cleans up the database once a day; you may want to disable it if you need to retain the data or want to collect data from multiple days.

16. When you finish the wizard, data collection starts automatically.

**Built-in Reports**

You may have a hard time finding the built-in reports for MDW because they must be used once before they will automatically appear in the SSMS menu. To get started:

- Open SSMS and go to one of the MDW databases.
- Right-click and select Reports.
- Select Management Data Warehouse.
- Select Management Data Warehouse Overview.

A collection of charts and graphs showing server activity will appear, as the example in Error! Reference source not found. shows.
To get more detail for any category, simply click the chart or graph you are interested in drilling down into. *Error! Reference source not found.* shows a sample drill-down report.
The built-in reports have a default charting period of 4 hours, which is sufficient for most troubleshooting. You also have other time intervals to choose from, as Figure 3 shows.

**Figure 3- MDW Reporting Periods**
**Advanced Troubleshooting and Analysis**

MDW’s real strength is that the data is recorded and available in regular SQL Server tables. This allows you to easily query the data from Microsoft Excel or any other tool. For example, comparing PerfMon counters from this Monday against last Monday (or 6 months ago) becomes trivial.

---

**SQL Server 2008 Best Practices Analyzer (BPA)**

The new SQL Server 2008 R2 BPA is scheduled for release in summer 2010. In addition, SQL Server 2005 BPA will run against SQL Server 2008 databases. SQL Server 2005 BPA functionality evolved into the Policy Management Framework in SQL Server 2008 SSMS to allow custom rules. SQL Server 2008 R2 BPA uses Microsoft Baseline Configuration Analyzer (MBCA) 2.0, which we walk through here to illustrate what you will see when the new tool becomes available for download.

For guidance about using SQL Server 2005 BPA, see the white paper “Best Maintenance Tools for PTC Windchill on Microsoft SQL Server 2005.”

1. After downloading and installing the MBCA, you will find it listed under Programs. Click the Microsoft Baseline Configuration Analyzer 2.0 link to launch it.

---

2. When the application appears, you might find the <Select> drop-down box empty—this is typical with the generic engine. You can download additional versions (including SQL Server 2008 BPA when it is released).

3. In the drop-down box, select SQL Server 2008 R2 BPA and then click Connect to Another Computer.

4. From the resulting Enter Parameters screen, select Analyze_SQL_Server_Engine.
5. After its analysis, MBCA will present a report showing errors and warnings.

6. Clicking an item will expand the details of the error or the warning.
**Expected Windchill Exceptions**

The BPA report might show the exception shown in Error! Reference source not found.; do not correct it.

**Table 2 - Expected BPA Violations**

<table>
<thead>
<tr>
<th>Violation</th>
<th>Recommended Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication Mode Violation</td>
<td>Do not take any action—by design</td>
</tr>
<tr>
<td>SQL Login Password Policy Violation by Login[wcAdmin]</td>
<td>Do not take any action—by design</td>
</tr>
</tbody>
</table>

Either correct all other violations or document the reason for the violation. PTC recommends running BPA and MCBA on the frequency shown in Error! Reference source not found..

**Table 3 - Recommended Frequency for Running BPA 2008 and MBCA 2.0**

<table>
<thead>
<tr>
<th>Time Since Installation</th>
<th>Frequency</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>First day</td>
<td>Once</td>
<td>Ensure that best practices were followed for the installation</td>
</tr>
<tr>
<td>After 6 weeks</td>
<td>Quarterly</td>
<td>Ensure that unauthorized changes have not happened</td>
</tr>
</tbody>
</table>

**Further Securing Your Windchill Database**

The National Institute of Standards and Technology (NIST) created the Security Content Automation Protocol (SCAP) program with extensive consolidation of best OS and SQL Server
practices. This program lists various vendor products that have passed certification for checking system security. PTC suggests using a scanner from this list in parallel to BPA.

PTC recommends using Microsoft Baseline Security Analyzer (MBSA) 2.1.1 to ensure a secure Server.

### Summary

The tables below summarize the minimum recommended practices using these maintenance tools. The critical items listed in Table 4 are good items to include as bullets in weekly reports. The suggested items listed in Table 5 are routine maintenance items that should be included in weekly reports when exceptions occur.

#### Table 4 - Summary of Recommended Run Schedule for Maintenance Tools

<table>
<thead>
<tr>
<th>Time from Install</th>
<th>Frequency</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>On install</td>
<td>Once</td>
<td>BPA, MBCA, MBSA</td>
</tr>
<tr>
<td>First 6 weeks</td>
<td>Daily</td>
<td>MDW</td>
</tr>
<tr>
<td></td>
<td>Weekly</td>
<td>Run DTA with default settings</td>
</tr>
<tr>
<td>First year</td>
<td>Monthly</td>
<td>Run DTA with indexes and index views; include filtered indexes</td>
</tr>
<tr>
<td>After 1 quarter</td>
<td>Weekly</td>
<td>MDW</td>
</tr>
<tr>
<td></td>
<td>Quarterly</td>
<td>Run DTA with full partitioning</td>
</tr>
<tr>
<td></td>
<td>Quarterly</td>
<td>Run DTA with indexes and index views; include filtered indexes</td>
</tr>
<tr>
<td></td>
<td>Quarterly</td>
<td>BPA, MBCA, MBSA</td>
</tr>
</tbody>
</table>

You can also consider using additional tools, such as:

- SQL Server Health and History Tool (SQLH2)
  - Performance Collector
  - Reports
- RML Utilities for SQL Server (x64)

However, these tools often have a high percentage of content overlap with reports available from the tools we covered and may not provide significant payback for the time required for you to review the information. In addition, they currently provide only basic SQL Server 2008 support (although full SQL Server 2005 support).

---

4 [http://sqlh2.codeplex.com/](http://sqlh2.codeplex.com/) has the latest version; the old version on Microsoft.com may hang on some machines.
Table 5 - Minimum Frequency of Other Maintenance Tasks

<table>
<thead>
<tr>
<th>At least</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Every 45 minutes</td>
<td>Transaction log backup</td>
</tr>
<tr>
<td>Every 3 hours</td>
<td>SQL Server database backup (Differential)</td>
</tr>
<tr>
<td>Daily</td>
<td>SQL Server database backup (Full)</td>
</tr>
<tr>
<td></td>
<td>Check Database Integrity</td>
</tr>
<tr>
<td>Weekly</td>
<td>SQL Server database backup (Full) and copy to offsite location</td>
</tr>
<tr>
<td></td>
<td>Rebuild Index (does Update Statistics also)</td>
</tr>
</tbody>
</table>

PTC suggests the following task sequence starting at 11pm daily:
1. Check Database Integrity
2. Rebuild Indexes (ALL)
3. Cleanup History

Follow the above tasks with this sequence one or more hours later:
1. Check Database Integrity (DBCC CHECKDB)
2. Backup Database (Full) – with verify backup integrity
3. Maintenance Cleanup Task

PTC suggests a differential database backup and transaction log backup plan such as the one in Table 6 for an 8am-5pm workday.

Table 6 - Full, Differential, and Transaction Log Backup Plan

<table>
<thead>
<tr>
<th>Time</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midnight</td>
<td>DBCC CHECKDB</td>
</tr>
<tr>
<td></td>
<td>Full backup</td>
</tr>
<tr>
<td></td>
<td>Maintenance Cleanup</td>
</tr>
<tr>
<td>8:45</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>9:30</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>10:15</td>
<td>Differential backup</td>
</tr>
<tr>
<td>11:00</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>11:45</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>12:30</td>
<td>Differential backup</td>
</tr>
<tr>
<td>13:15</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>14:00</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>14:45</td>
<td>Differential backup</td>
</tr>
<tr>
<td>15:30</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>16:15</td>
<td>Transaction Log backup</td>
</tr>
<tr>
<td>23:00</td>
<td>DBCC CHECKDB</td>
</tr>
<tr>
<td></td>
<td>Rebuild Indexes</td>
</tr>
<tr>
<td></td>
<td>Cleanup History</td>
</tr>
</tbody>
</table>

You can automate some of the above maintenance items through SSMS. In the Object Explorer, select Server\(\{Windchill database name\}\), Management, Maintenance Plans.
For information about Maintenance Plans, see:

- [How to: Create a Maintenance Plan](#)
- [Maintenance Plan Wizard](#)
- [Setting up a Maintenance Plan to Backup Databases](#)
- [Top Tips for Effective Database Maintenance](#)
Links for Further Information

PTC Windchill
- Microsoft SQL Server: Improved Performance for PTC Windchill
- Windchill and Pro/INTRALINK 9.0 and 9.1 Server Hardware Sizing Guidelines – Microsoft Windows Platform

SQL Server information can be found in Books Online:
- SQL Server 2008 Books Online

SQL Server Books Online also includes best practice information in the following articles:
- Best Practices for Replication Administration
- Replication Security Best Practices
- Best Practices for Recovering a Database to a Specific Recovery Point

See the SQL Server Best Practices portal for technical white papers, the SQL Server Best Practices Toolbox, Top 10 Lists, and other resources.

Following is a list of technical white papers that were tested and validated by the SQL Server development team. These can help you learn more about specific SQL Server topics.

- A Quick Look at Serial ATA (SATA) Disk Performance
- Advanced Troubleshooting with Extended Events
- Backup Compression, Client Redirection with Mirroring, and More
- Best practices for operational excellence
- Best Practices for Running SQL Server on Computers That Have More Than 64 CPUs
- Best Practices for Semantic Data Modeling for Performance and Scalability
- Checklist: SQL Server Performance
- Checksum Problems, Choosing the Correct Recovery Model and More
- Common SQL Server Security Issues and Solutions
- Comparing Tables Organized with Clustered Indexes versus Heaps
- Corruption Recovery Tips, Database Shrinkage Advice and More
- Database Consistency, Temporary Tables, and More
- Database Snapshot Performance Considerations Under I/O-Intensive Workloads
- Description of using disk drive caches with SQL Server that every database administrator should know
- Disk Partition Alignment Best Practices for SQL Server
• FLASH Disk Opportunity for Server-Applications
• Generic Database Security Checklist (Version 8, Release 1.3)
• How to mirror the system and boot partition (RAID1) in Windows Server 2003
• How to use the SQLIOSim utility to simulate SQL Server activity on a disk subsystem
• How To: Use SQL Profiler
• I/O Errors, Database Mirroring, and More
• Improving SQL Server Performance
• Large Transaction Logs, When to Use Repair, and More
• Microsoft SQL Server Database Engine Input/Output Requirements
• Microsoft SQL Server I/O subsystem requirements for the Temp Db database
• Minimize Blocking in SQL Server
• Monitor and troubleshoot storage performance
• Moving, Performance Tuning, Backing Up and Mirroring Databases
• Partial Database Availability
• Partitioning, Consistency Checks, and More
• Performance of WD 250GB SATA Drives + 3ware Controller
• Physical Database Storage Design
• Pre-Configuration Database Optimizations
• Removing Index Fragmentation, Synchronizing vs. Synchronized, and More
• RML Utilities for SQL Server (x64)
• Row-Overflow, Differential Backups, and More
• SQL Server 2008 Management Data Warehouse
• SQL Server 2008 Policy-Based Management
• SQL Server Best Practices
• SQL Server Health and History Tool (SQLH2)
• SQL Server: Recovering from Disasters Using Backups
• SQLIO Disk Subsystem Benchmark Tool
• Statistics Used by the Query Optimizer in Microsoft SQL Server 2008
• Storage Top 10 Best Practices
• Support WebCast: How to Effectively Use SQL Server Profiler
• Technical Note #28: Common QA for deploying SQL Server ...
• Top Tips for Effective Database Maintenance
• Troubleshooting Performance Problems in SQL Server 2008
• Uncover Hidden Data to Optimize Application Performance
• Understanding Logging and Recovery in SQL Server
• Understanding SQL Server Backups
• Unexpected Consistency Checks, Troubleshooting Memory Usage, More
• RML Utilities: An Internal MS Jewel for Performance Analysis
• Cumulative Update 1 to the RML Utilities for Microsoft SQL Server Released

Bibliography
• SQL Server MVP Deep Dives (Manning, 2010), Paul Nielsen, Kalen Delaney, et al.
• Mastering SQL Server Profiler (Red Gate Books 2009), Brad McGehee
• Microsoft SQL Server 2008 Bible (Wiley Publishing, 2009), Paul Nielsen, Mike White, and Uttam Parui
• Microsoft SQL Server™ 2008 Internals: The Storage Engine (Microsoft Press, 2009), Kalen Delaney
• Professional SQL Server 2008 Administration (Wiley, 2009), Brian Knight, Katen Parel, Steven Wort, et al.