



DEMAND TECHNOLOGY
●●●SOFTWARE

Quick Start Guide: Web Portal and PDB for Performance Sentry

VERSION 4.0

Use with

Windows Server 2016 and newer

Windows 10 and newer

Copyright Information

This document constitutes a proprietary development and a valuable work product in the form of trade secrets, know-how, and confidential information which is the exclusive property of DEMAND TECHNOLOGY SOFTWARE, INC. The user shall prevent the disclosure of, or access to the contents thereof by agents, employees, or representatives of the user to any parties outside the user's organization. It shall be given the same protection which the user gives his own confidential material at all times.

This document was prepared to assist users of NTSMF and Performance Sentry, products of DEMAND TECHNOLOGY SOFTWARE, INC. The contents of this document may not be used for any other purpose without written permission. The material contained herein is supplied without representation or warranty of any kind. DEMAND TECHNOLOGY SOFTWARE, INC., therefore assumes no responsibility and shall have no liability of any kind arising from the supply or use of this document or the material contained herein.

This document is an unpublished work fully protected by the United States copyright laws and is considered a trade secret belonging to the copyright holder, DEMAND TECHNOLOGY SOFTWARE, INC.

All rights reserved. This document, in whole or in part, may not be reproduced or transmitted in any form or by any means without the express prior written consent of DEMAND TECHNOLOGY SOFTWARE, INC.

Performance Sentry™ and Performance Sentry VM™ are registered trademarks of DEMAND TECHNOLOGY SOFTWARE, INC.

All other brand and product names are trademarks or registered trademarks of their respective companies.

Table of Contents

Copyright Information	1
Installing the Software.....	3
System Requirements	4
Install the Software.....	4
Using IIS Manager to Change the Application Pool Identity.....	27
Run the Performance Sentry Portal.....	33
Setup Security Administration	34
Viewing Performance Data from Other Windows Servers	42
Viewing Linux Data	44
Using the Performance Sentry Portal.....	45
Default.aspx landing page	45
Daily Performance Charts	46
Box Plot and more controls.....	53
Chart Template Editor	55
Save Chart Definition.....	60
Walkthrough: Modifying an existing chart template and saving it as a new chart template	62
Server Health Reports.....	68
Server Health Report Template Editor.....	70
How All of This Works.....	75
Performance Sentry Performance Datatbase (PDB)	77
How the Performance Data is Stored in the Database	77
View Data Tables	79
How to Transition from Evaluation to Production.....	80

Installation and Overview of Performance Sentry Web Portal

The Performance Sentry Quick Start Guide provides an overview of the components that setup installs, the system requirements for installing the components and the install procedure. There is also a brief overview of the performance data collection process.

Setup installs these components:

- (1) Performance Sentry Collection Service
- (2) Performance Sentry Administration
- (3) Performance Sentry Web Portal and Performance Database

By default, all of the components are selected, but you may change the selection if there are components that do not need to be installed.

Performance Sentry Collection Service runs as a Windows Service 24 x 7, starting when the machine boots and only stopping when the operating system is shut down. It is an agent that runs locally on each machine where it is installed. It gathers the Windows performance counter data, specified in the Data Collection Set (DCS) that it is assigned to use. Performance Sentry includes several pre-defined Data Collection Sets that make it very easy to install and immediately begin collecting valuable performance data.

Performance Sentry Administration is a Windows desktop application that you install which is used to select and modify the Data Collection Sets that determine how performance data is collected by individual Performance Sentry collection agents.

The Performance Sentry Performance Database (PDB) is a Microsoft SQL Server-based repository containing all of the most recent performance data gathered by Performance Sentry collection agents and loaded into the PDB by the Loader utility.

The Performance Sentry Portal is a web-based graphical interface to the PDB. The Portal contains chart and reporting panels for use in performance investigations. It includes a comprehensive set of pre-defined report and chart templates, which are loaded and stored in SQL Tables in the PDB. It also contains editing tools for creating your additional custom charts and reports, based on your corporate needs.

For more detail on each of these components please see the section, “How All of This Works”.

Installing the Software

Setup begins by checking for any system requirements that may require a system restart if they are subsequently installed. These system requirements are listed in the “System Requirements” section below. If there are no pre-requisites required prior to installation, then the setup continues by asking for destination locations for each of the components. If the Portal and PDB component is selected, then setup also checks for the presence of an instance of Microsoft SQL Server. If no SQL Server is installed, then setup will automatically install SQL Server Express Edition during the Portal and PDB component installation.

Quick Start Guide: Web Portal and PDB for Performance Sentry

There are four steps to the full installation process. The Setup program installs the Performance Sentry Collection Service, and it is activated immediately using a default Collection set. With the collection agent running in the background, performance data will be gathered during the remaining steps in the installation process. Setup then installs the Sentry Administration component. Setup will then install the pre-requisites for the Performance Sentry Portal and PDB, which may include SQL Server Express Edition as well as some IIS components.

After all components are installed, the setup routine then loads the performance database (PDB) with sample performance data for a fictitious computer named 'SQLSERVER', and then loads the performance data collected on the local machine during the installation. Finally, setup prompts with a checkbox to open IIS Manager to modify the application pool identity to connect to the SQL Performance Database. Setup will end and if the box was checked, IIS Manager will start and WordPad will open to display instructions for modifying the application pool identity.

System Requirements

System requirements for installation are:

- Windows 7 or newer or Windows Server 2012 R2 or newer¹
- Microsoft SQL Server 2005 or newer²
- Microsoft .NET Framework 4.6³ (if you choose to install SQL Server Express Edition 2016)
- Microsoft IIS with ASP.NET 4.0 (or newer)

NOTES:

1. The Performance Sentry Collection Service can be installed on older versions of the Windows Operating System. Contact support@demandtech.com for compatibility with older systems.
2. If Microsoft SQL Server is not present, then Microsoft SQL Server 2016 Express Edition will be installed as a pre-requisite for the Performance Sentry Portal and PDB component.
3. .NET Framework 4.6 is a prerequisite for installing SQL Server Express Edition 2016. It is pre-installed for Windows Server 2016, but it is not installed, by default, on Server 2012R2. It has been our experience that installing the Framework on Server 2012 most often results in a system restart (reboot). If the Framework is not installed prior to starting the Performance Sentry Portal and PBD setup, you will be prompted to install the Framework, or cancel. If you cancel, you must manually install the Framework. If you continue, the Framework will be downloaded and installed, after which the system may need to be restarted. If a system restart is required, you will have to restart the Performance Sentry Portal and PDB installation from the beginning.

Install the Software

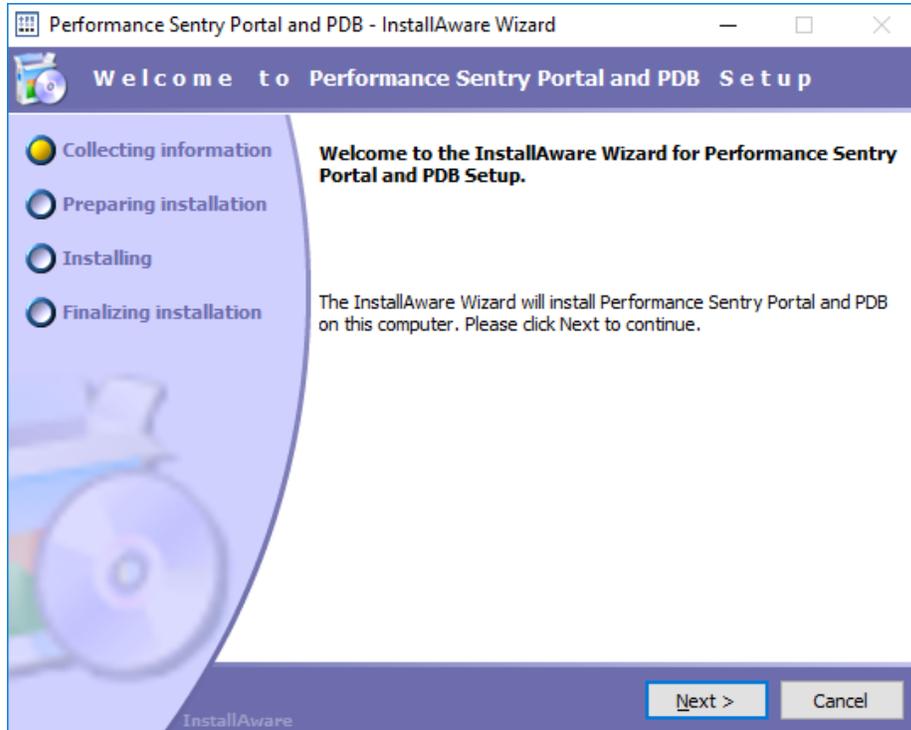
1. Locate the files downloaded from the www.demandtech.com website and select **Performance Sentry Portal.exe**.

After you click on the executable to run it, the InstallAware Wizard dialog box displays a message that the contents of the setup package are being verified.

Quick Start Guide: Web Portal and PDB for Performance Sentry

The Demand Technology Logo splash screen then displays with this InstallAware Wizard dialog box message: **InstallAware is preparing the InstallAware Wizard which will install this application. Please Wait.**

Next, the “Welcome to Performance Sentry Portal and PDB Setup” window displays:

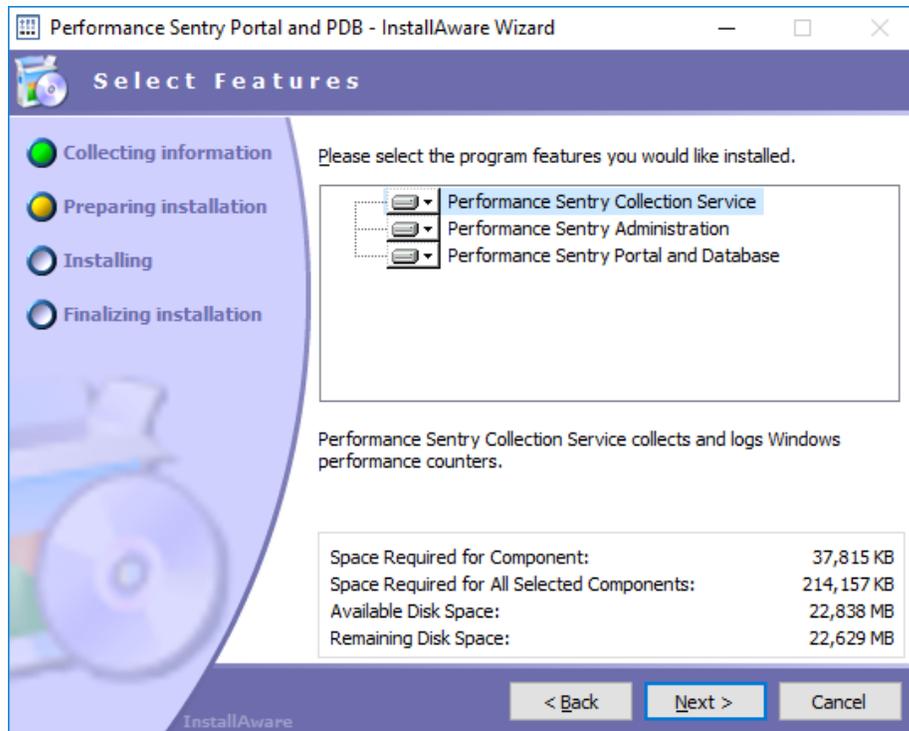


2. Click **Next**.

Quick Start Guide: Web Portal and PDB for Performance Sentry

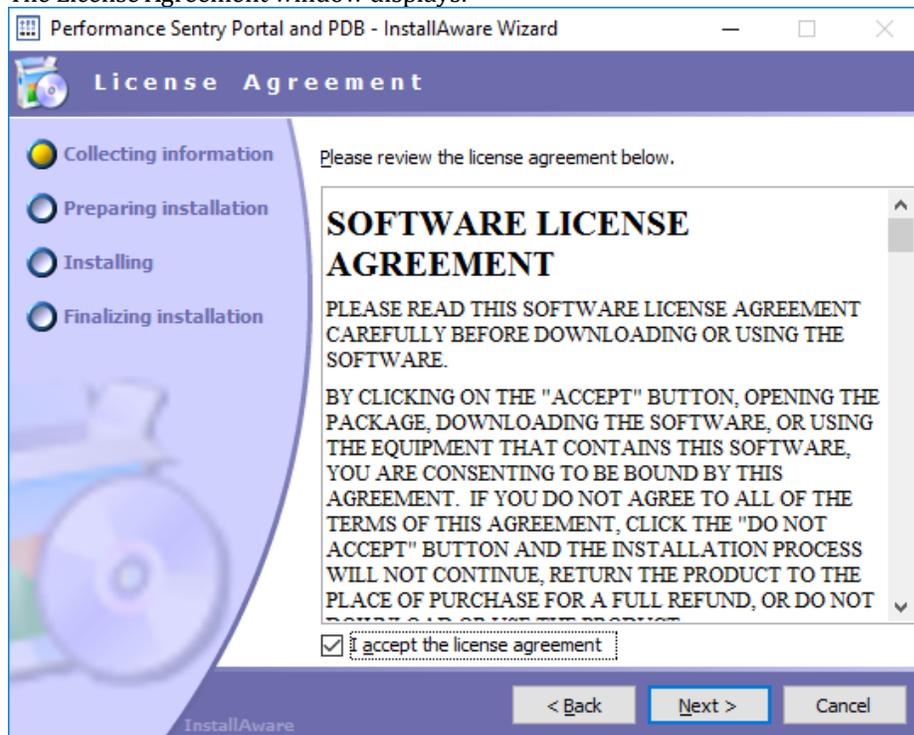
The “Select Features” window displays.

Note: All three components are selected by default. If you have already installed one of the components, or choose not to install it on this machine then click the dropdown icon next to it and select **X**.



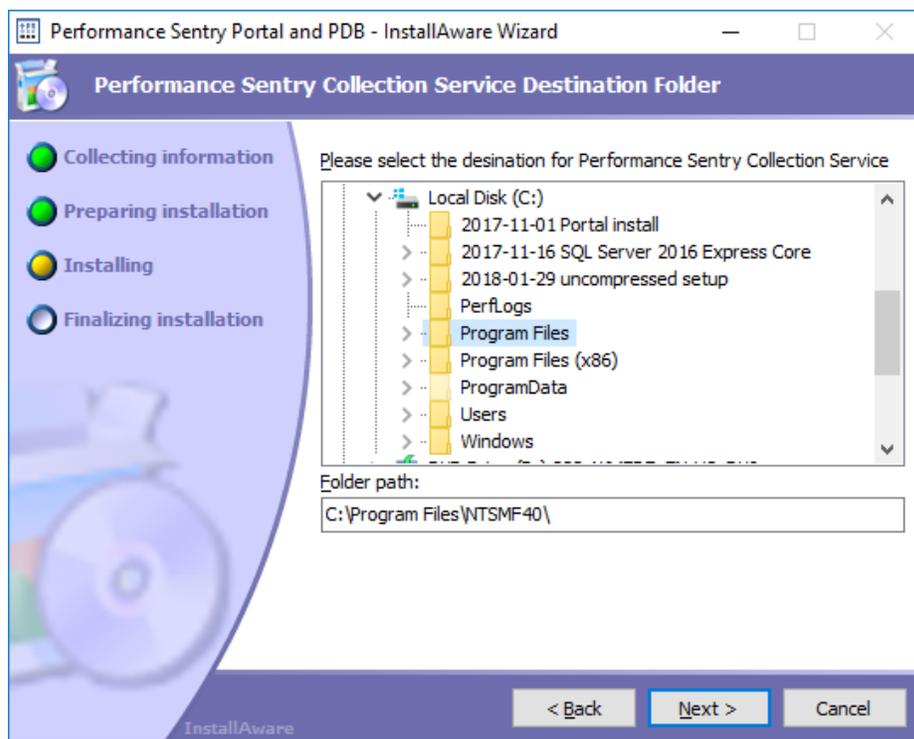
3. Select the components that you wish to install then click **Next**.

The License Agreement window displays:



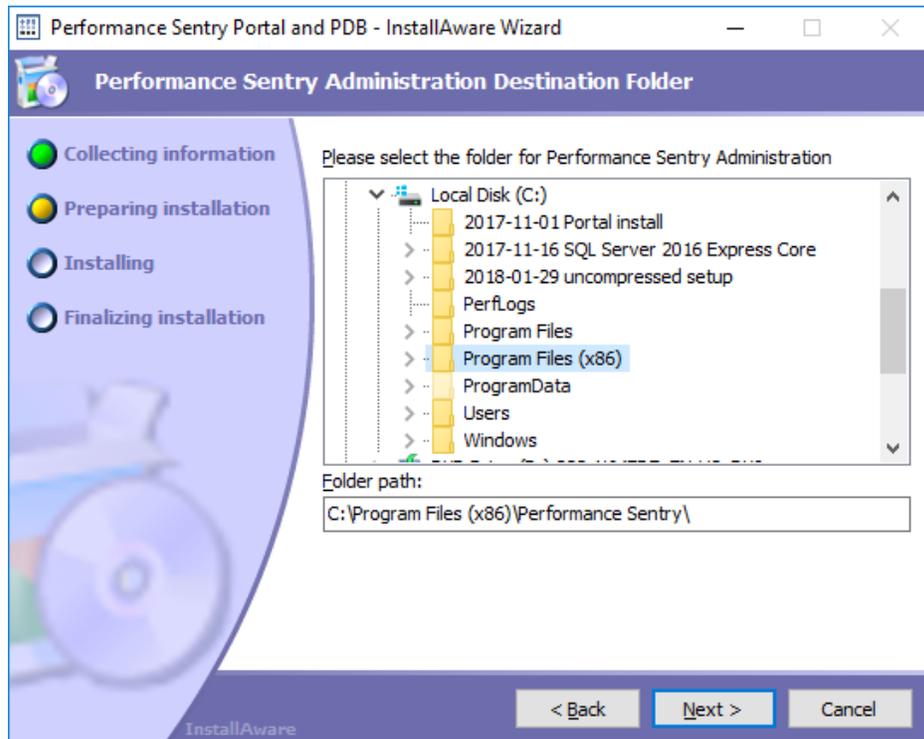
4. Review the End User License Agreement and If you agree with the terms of the Agreement check the box next to 'I accept the license agreement' and Click **Next**.

The Performance Sentry Collection Service destination folder window displays:



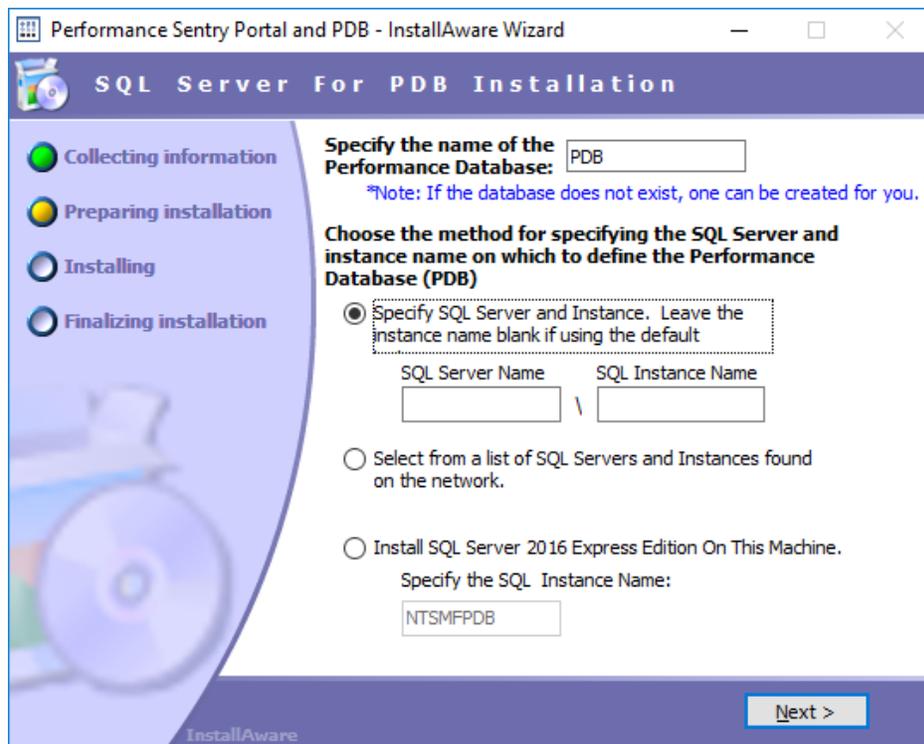
5. Accept the default, or browse to another folder and click **Next**.

The Performance Sentry Administration destination folder window displays:



6. Accept the default, or browse to another folder, select it and click **Next**.

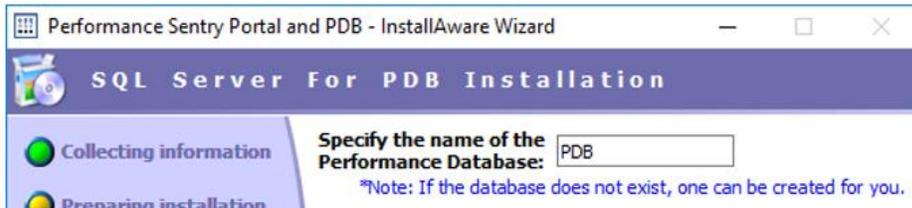
The SQL Server for PDB Installation window displays:



Quick Start Guide: Web Portal and PDB for Performance Sentry

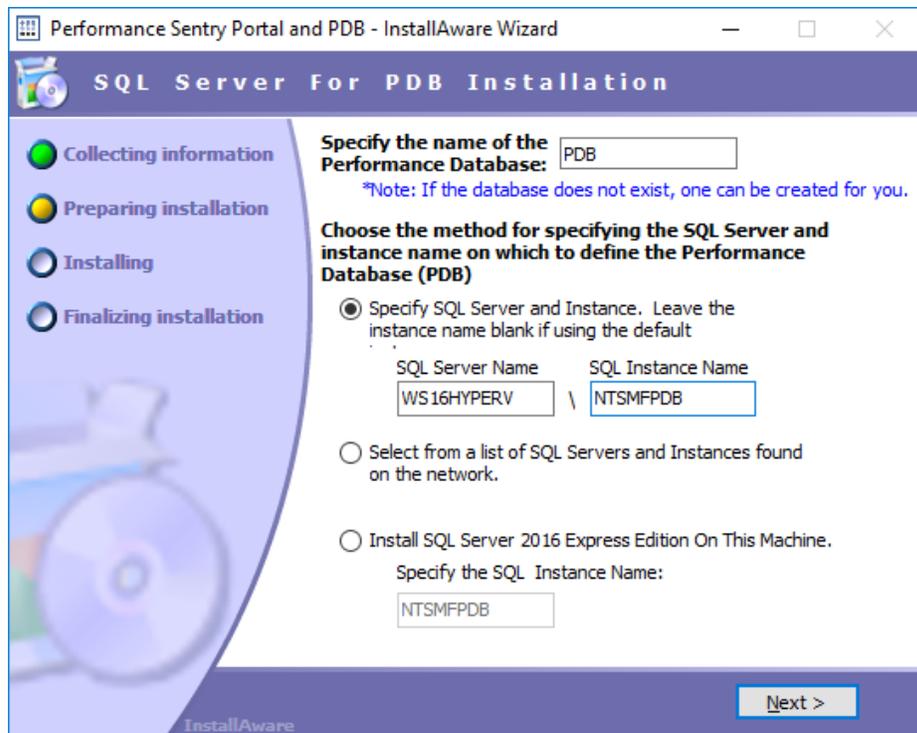
7. You will enter the name of the Performance Database (PDB) and select a pre-installed instance of SQL Server, or elect to install SQL Server 2016 Express Edition.

First you select the name of the Performance Database. The default is "PDB":

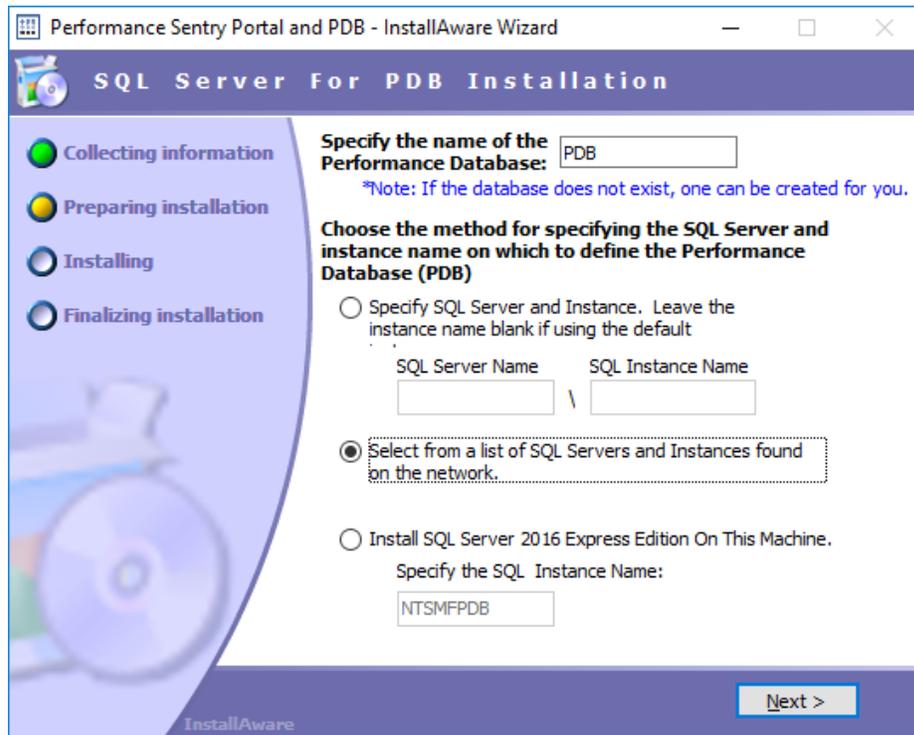


Next you choose from three options. You can specify the SQL Server and instance name where the Performance Database will be defined, or you can choose to browse the network for the SQL Server, or you can choose to Install SQL Server 2016 Express Edition on the target machine.

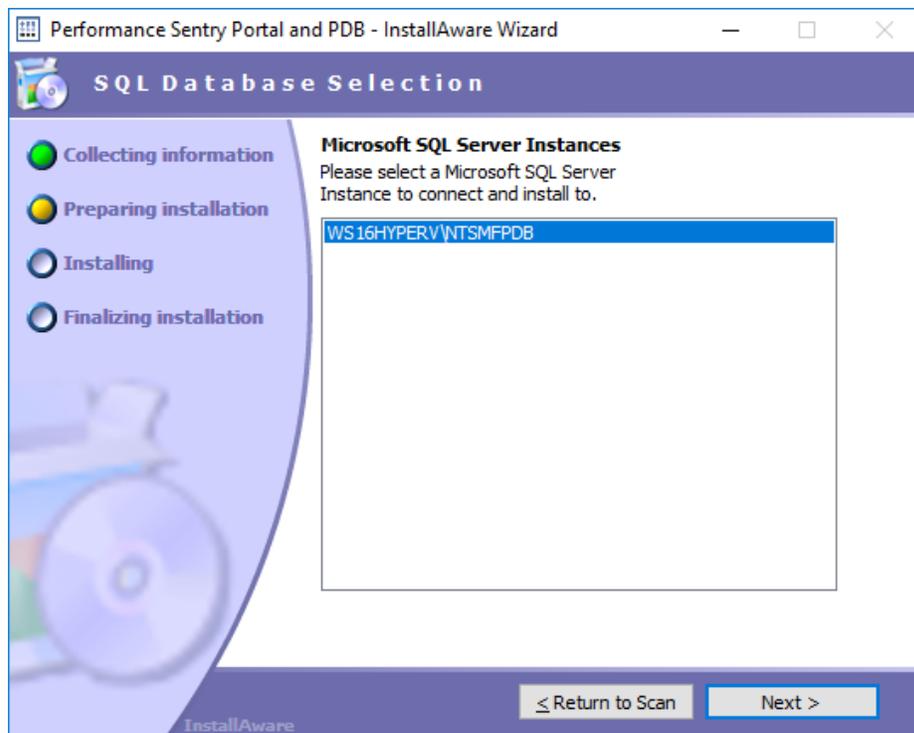
Here is an example of the first option specifying the SQL Server and Instance Name:



Here is an example where you elect to choose from a list of SQL Servers:

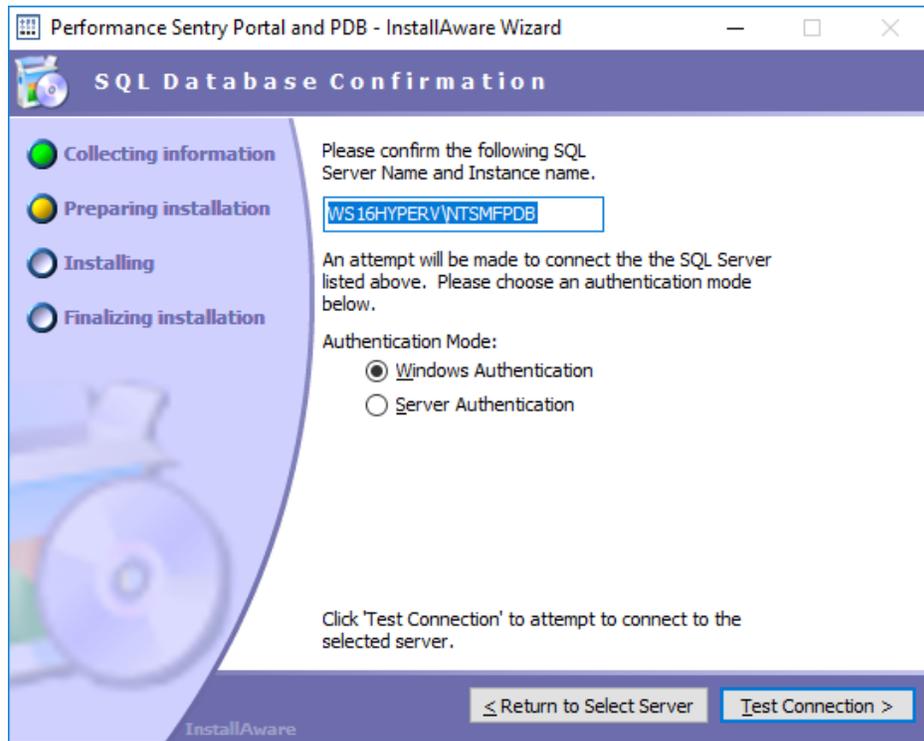


Note: If you choose to browse for a server (the second option), you must have SQL Server Browser Service running. You will be prompted to choose a server from the list:



Click **Next**.

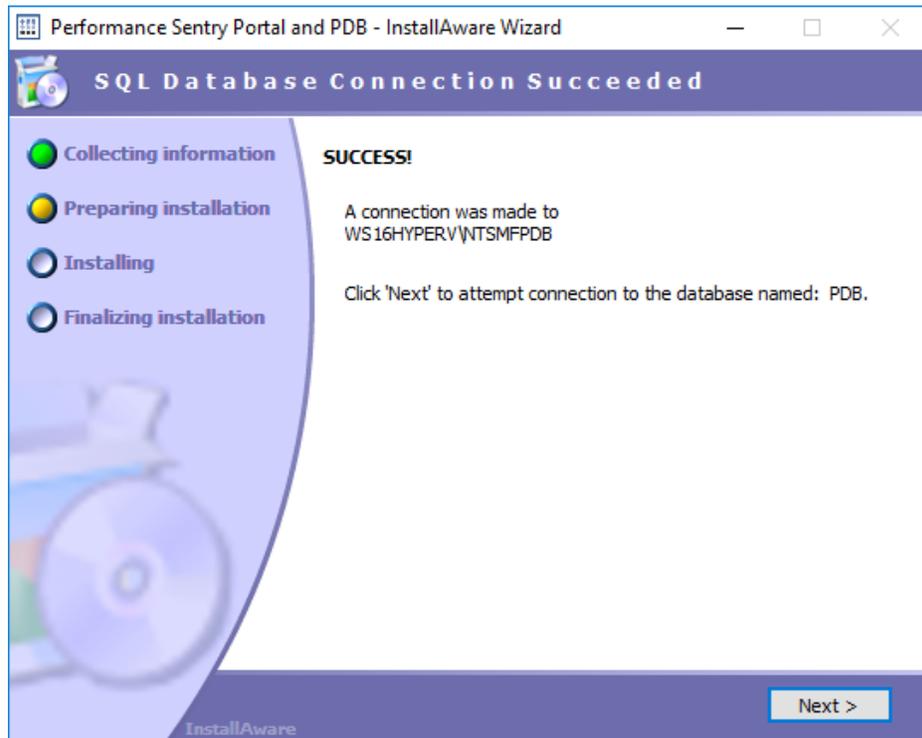
You then choose an authentication mode:



Click **Test Connection**.

Quick Start Guide: Web Portal and PDB for Performance Sentry

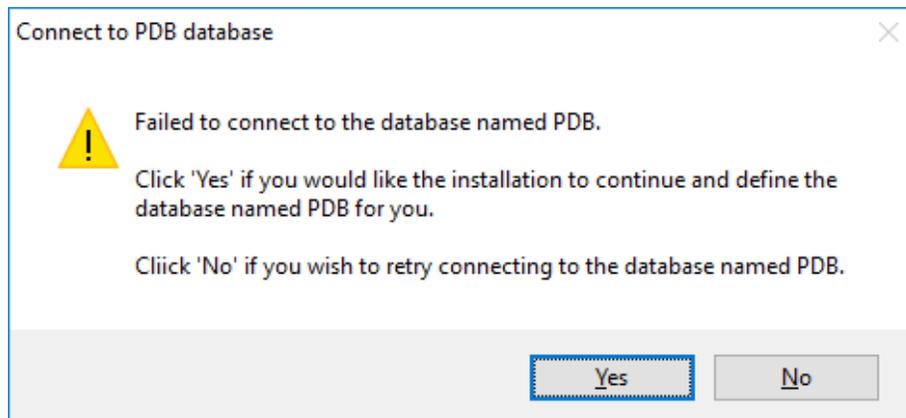
If you specified the SQL Server or chose one from the list, the installation will attempt to connect to the SQL Server and if successful, it will display the following window:



The installation will now attempt to connect to the database previously specified in the 'SQL Server for PDB Installation' window.

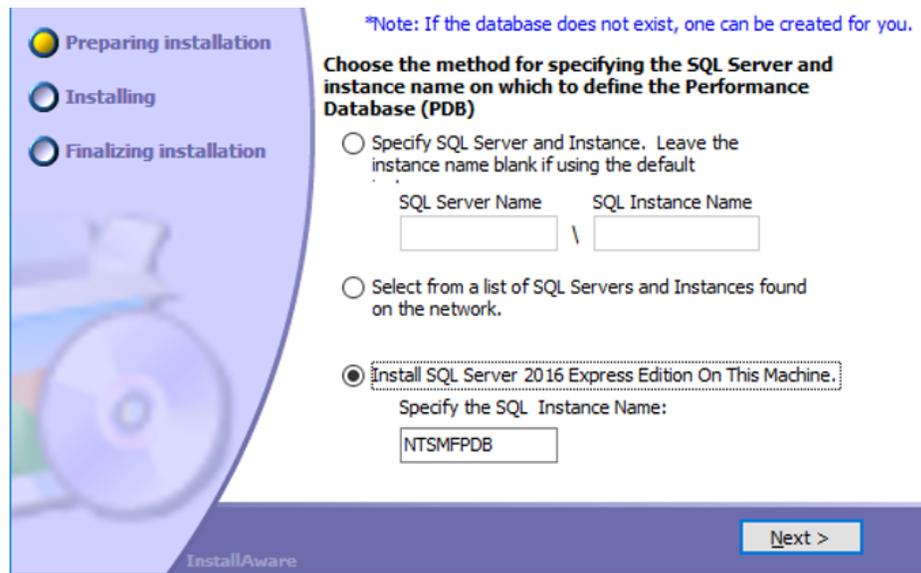
Click **Next**.

If the database is not found, you will receive the following message:

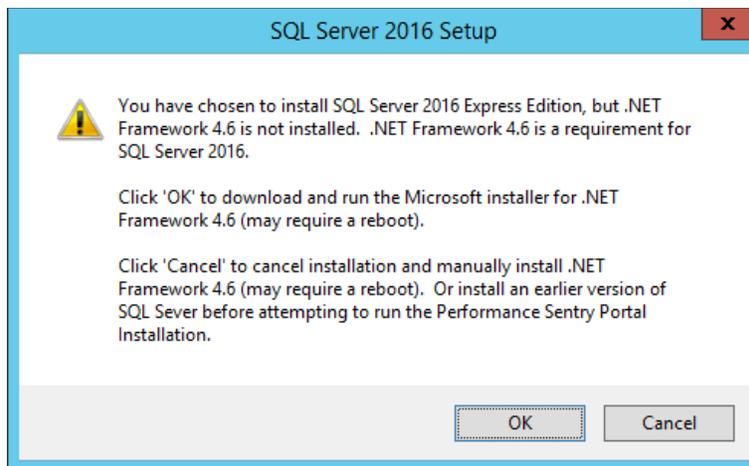


Click **Yes** and the database named 'PDB' will be defined later in the installation program.

If you chose to install SQL Server 2016 Express Edition:



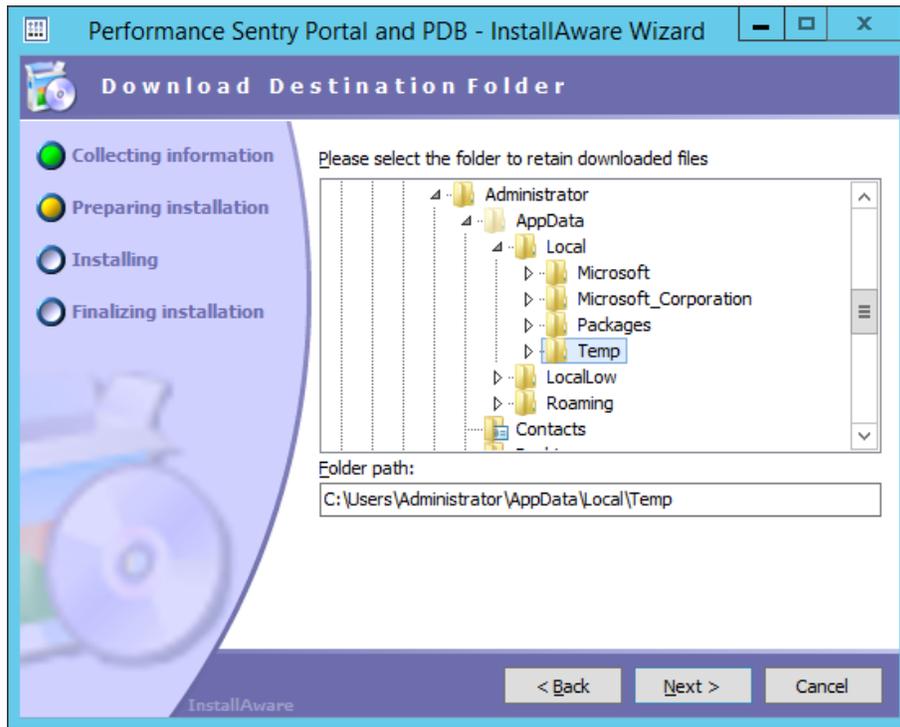
And you are installing to Windows machine that does not have .NET Framework 4.6 installed, then you will receive this message:



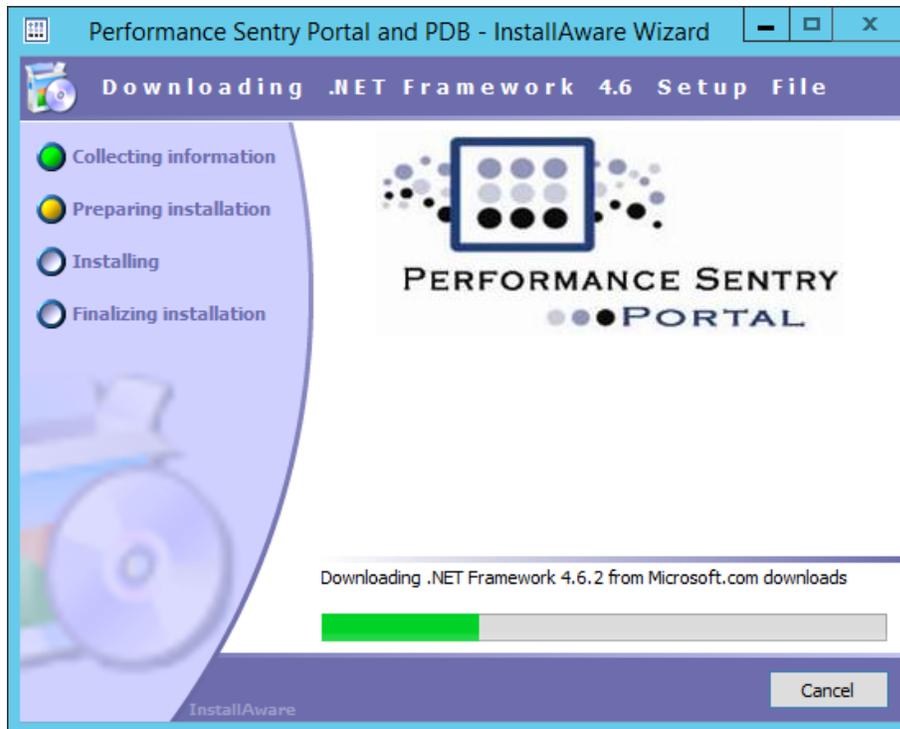
Please Note: It has been our experience that installing the Framework on Server 2012 most often results in a system restart (reboot). Therefore, if you do not want to take the chance of having to reboot your system, click **Cancel** to cancel installation and install .NET Framework 4.6, or install SQL Server 2016 (or previous version), at your convenience.

Quick Start Guide: Web Portal and PDB for Performance Sentry

If you click **OK** then you will be prompted for the folder to download the .NET Framework 4.6 setup:

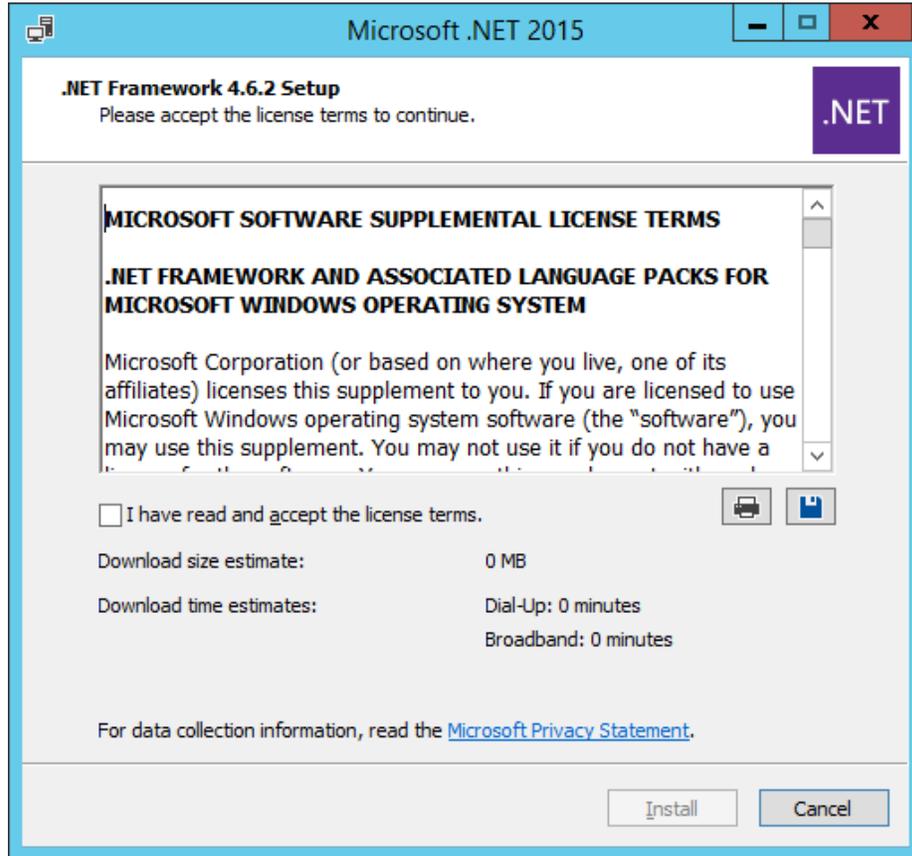


Click 'Next' and the download will begin:

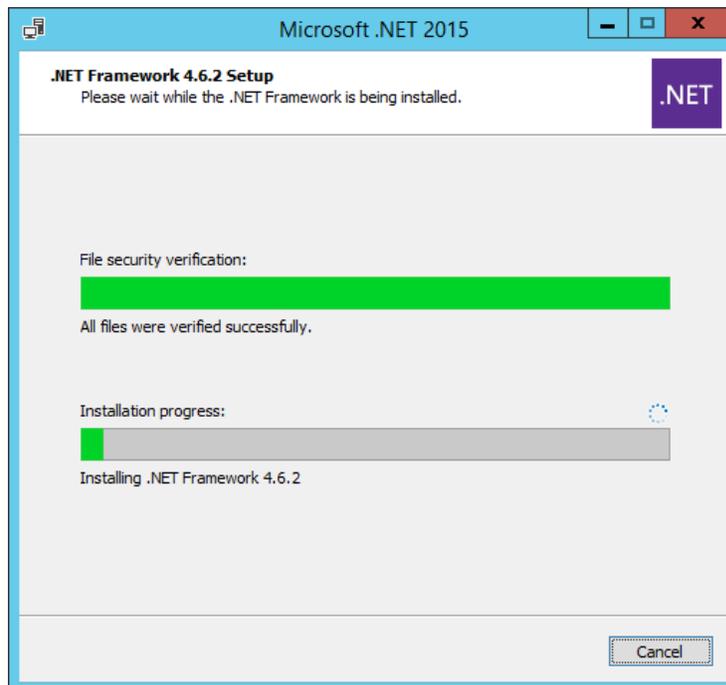


Quick Start Guide: Web Portal and PDB for Performance Sentry

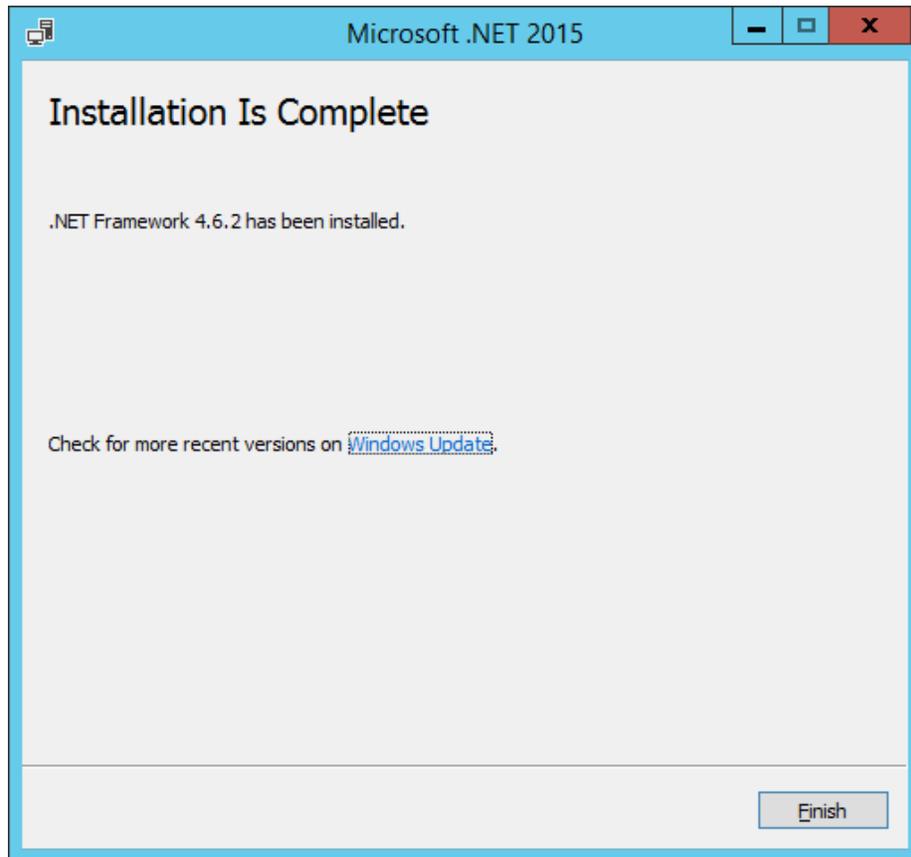
Once the setup has been downloaded, it will automatically start and prompt with the following message. Click the box to accept the license terms, then click 'Install' to begin installation.



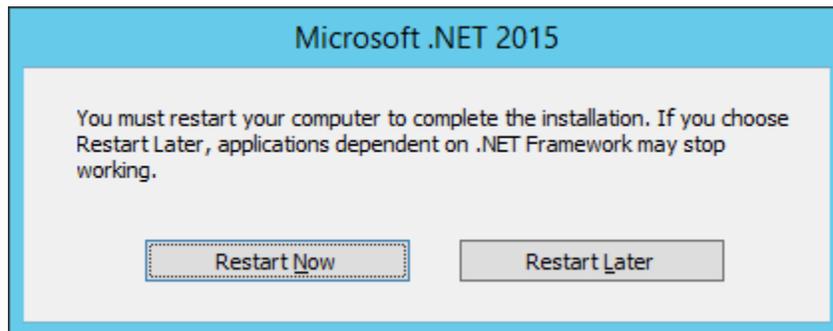
You will see the following progress window:



followed by installation confirmation:



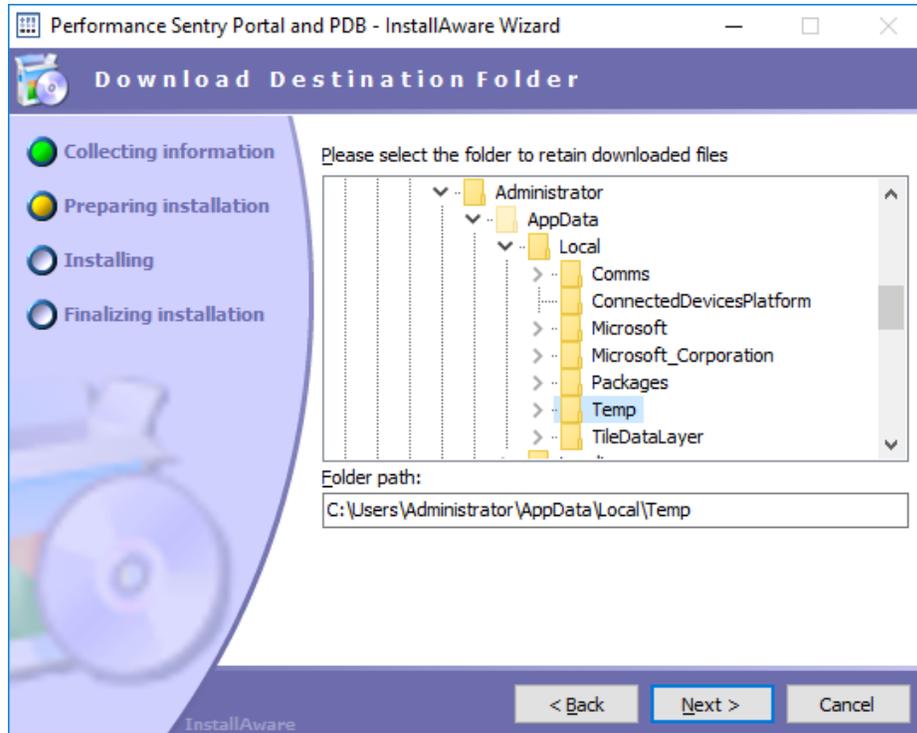
and finally, a restart notification:



Please Note: if you do not restart, then the SQL Server 2016 Express Edition installation will fail (because .NET Framework 4.6 has not been fully installed) and you will have to restart your machine to complete the installation of .NET Framework 4.6, then *restart the Performance Sentry Portal and PDB installation.*

Quick Start Guide: Web Portal and PDB for Performance Sentry

Continuing with the installation and if you elected to install SQL Server Express Edition, you will be prompted for the folder to retain the SQL Server 2016 Express Edition setup file which will be downloaded from Microsoft's download site. By default, the file will be stored in the user's AppData\Local\Temp folder. If this is not acceptable, you may browse to another location to store the download file:

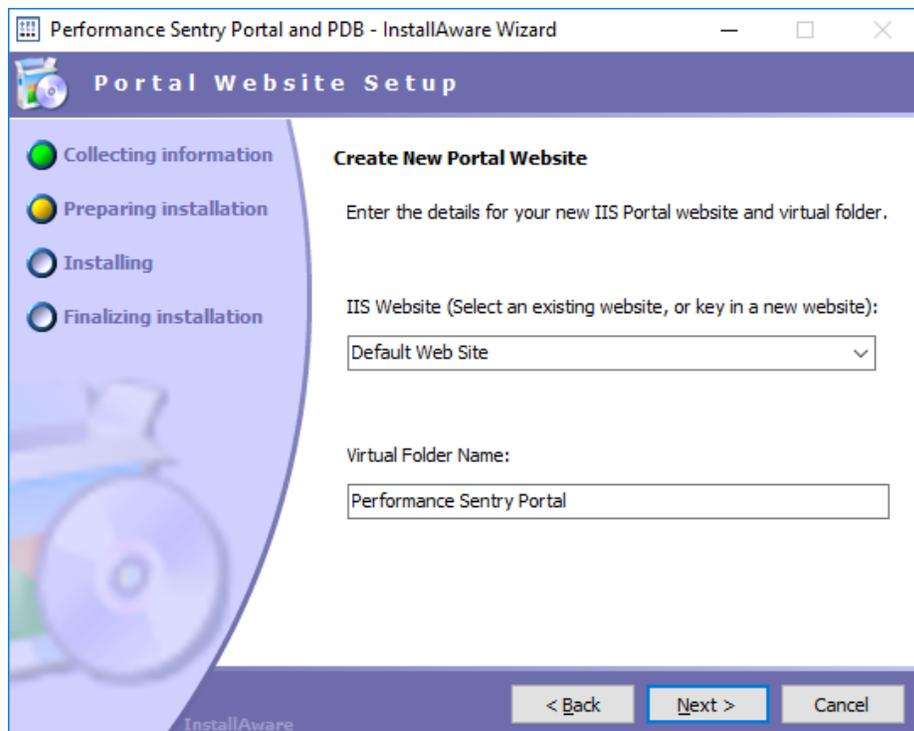


Click **Next**.

The SQL Server Setup file will be downloaded:

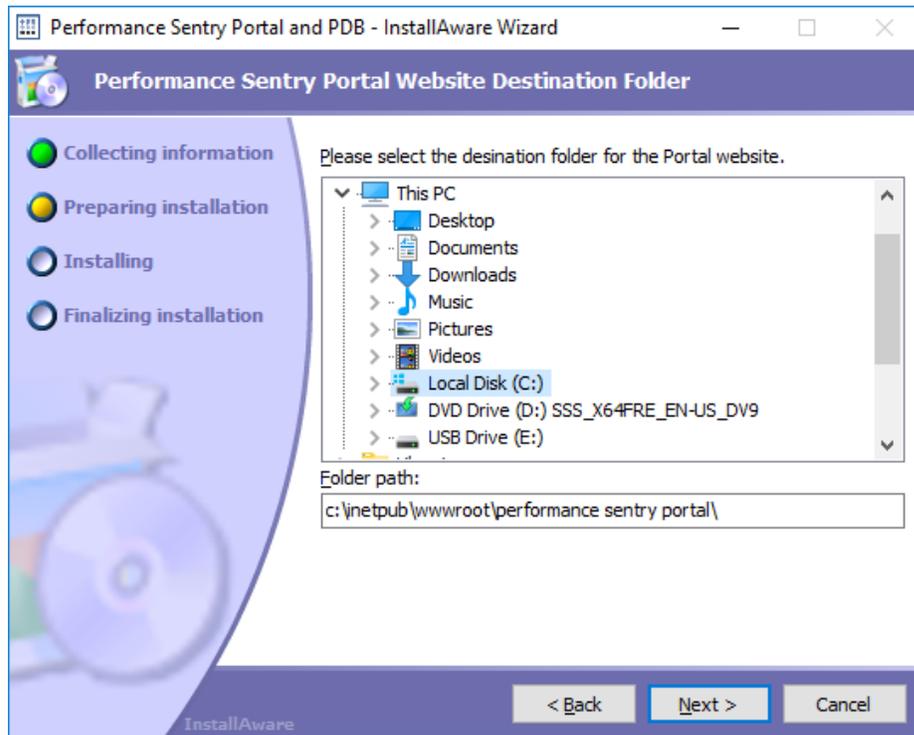


Once the download has completed or you have specified an existing SQL Server to connect to, the Performance Sentry Portal website creation window will display:



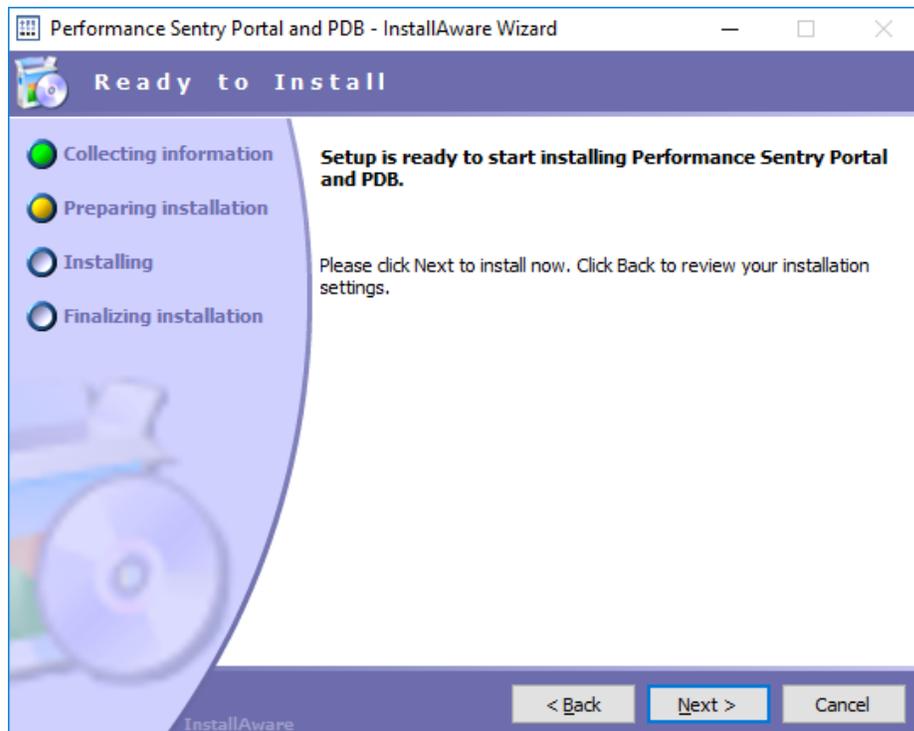
8. Make desired changes or accept the defaults and click **Next**.

The Performance Sentry Portal website destination window will display:



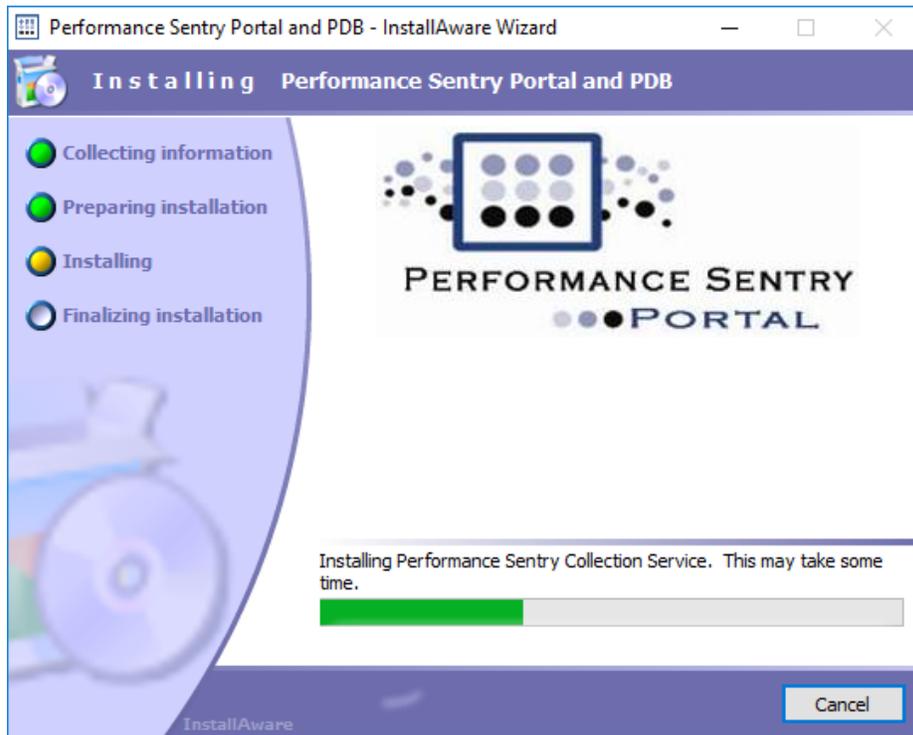
9. Browse to a different folder or accept the defaults and click **Next**.

The Performance Sentry Portal Ready to Install window will display:

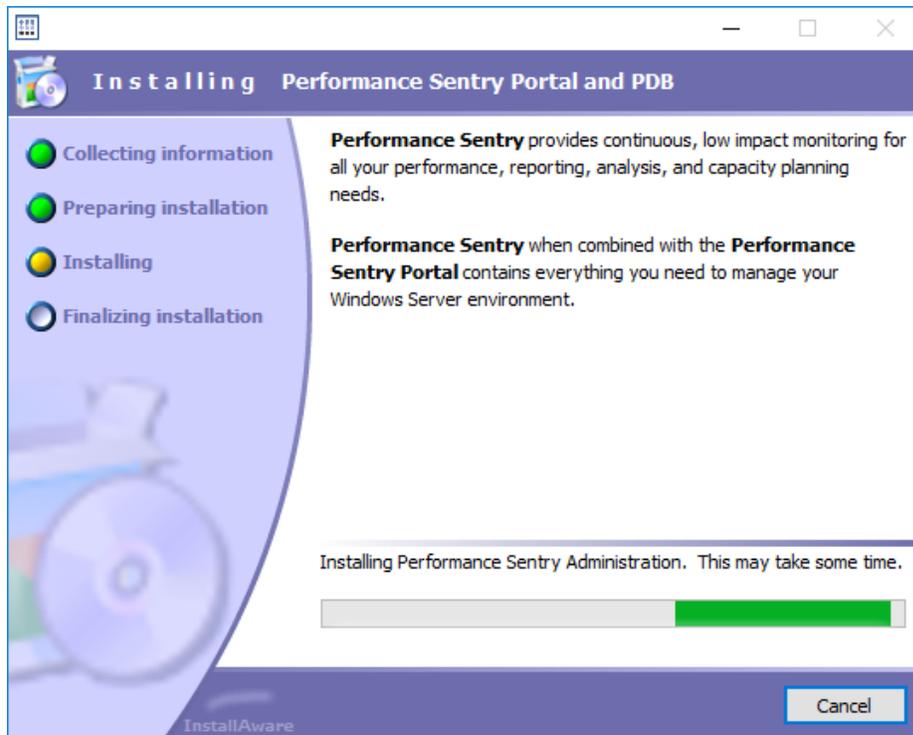


10. All necessary information has been received. Click **Next** to proceed with the installation.

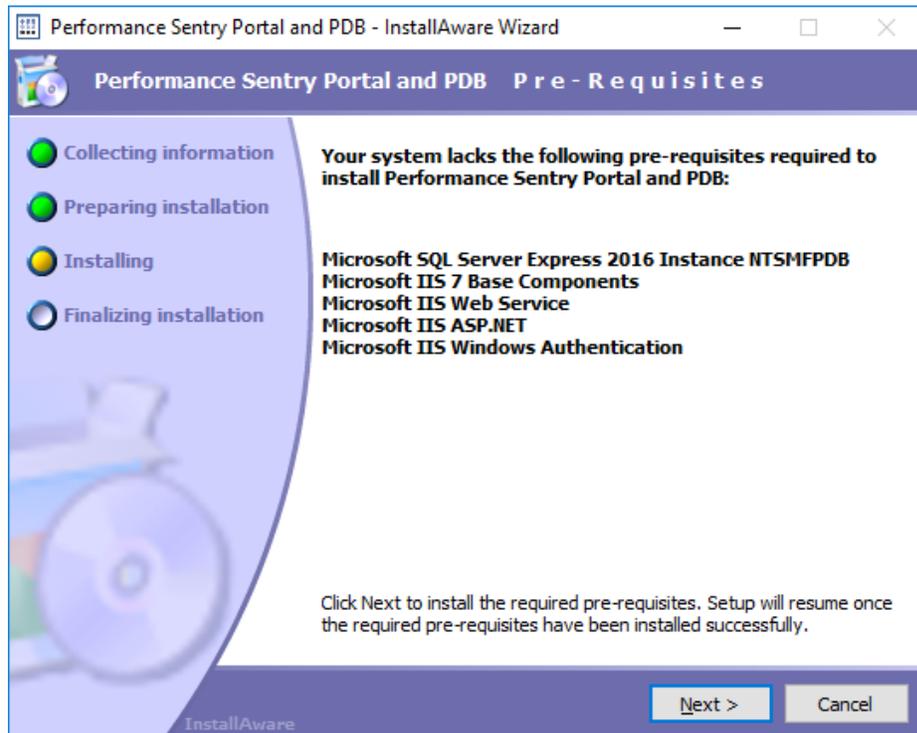
Setup then installs the Performance Sentry Collection Service:



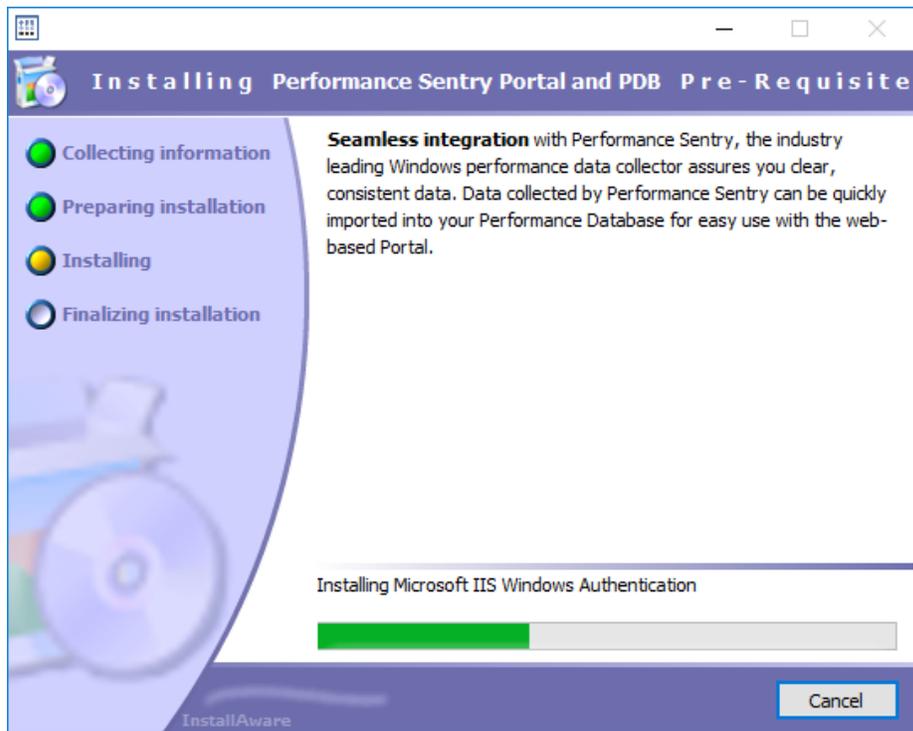
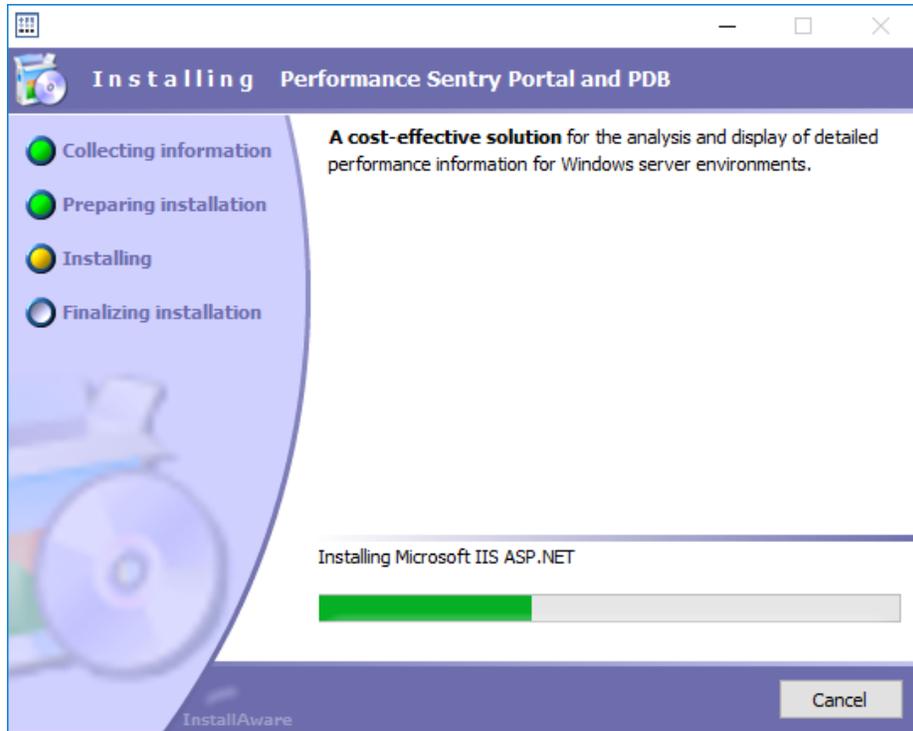
...followed by the installation of Performance Sentry Administration:



Please Note: the pre-requisite window below only displays if Microsoft SQL Server or IIS are not installed

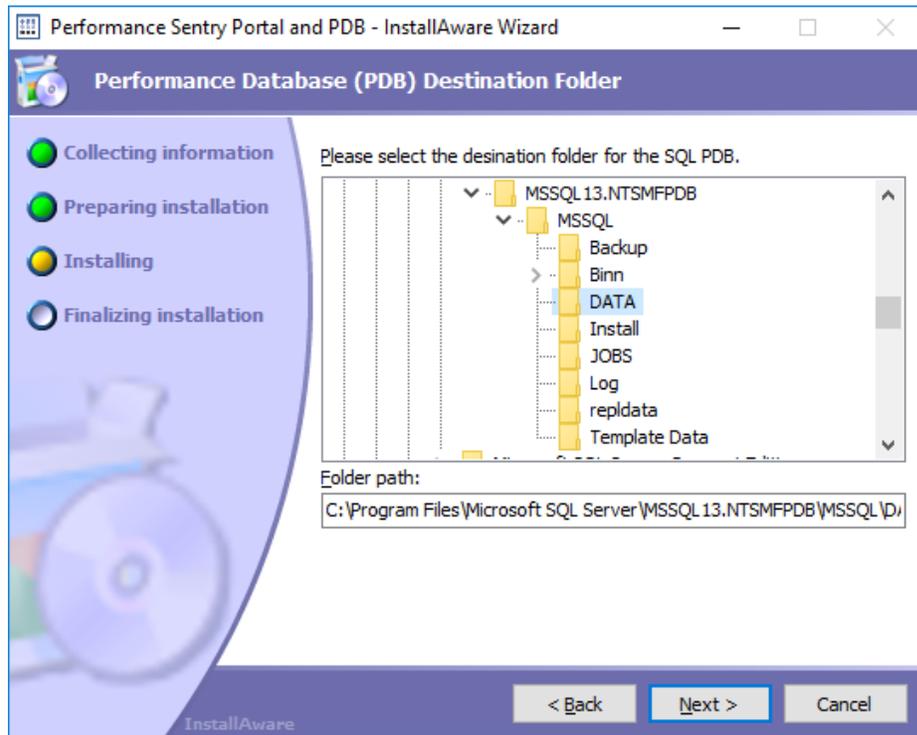


11. Click **Next**.



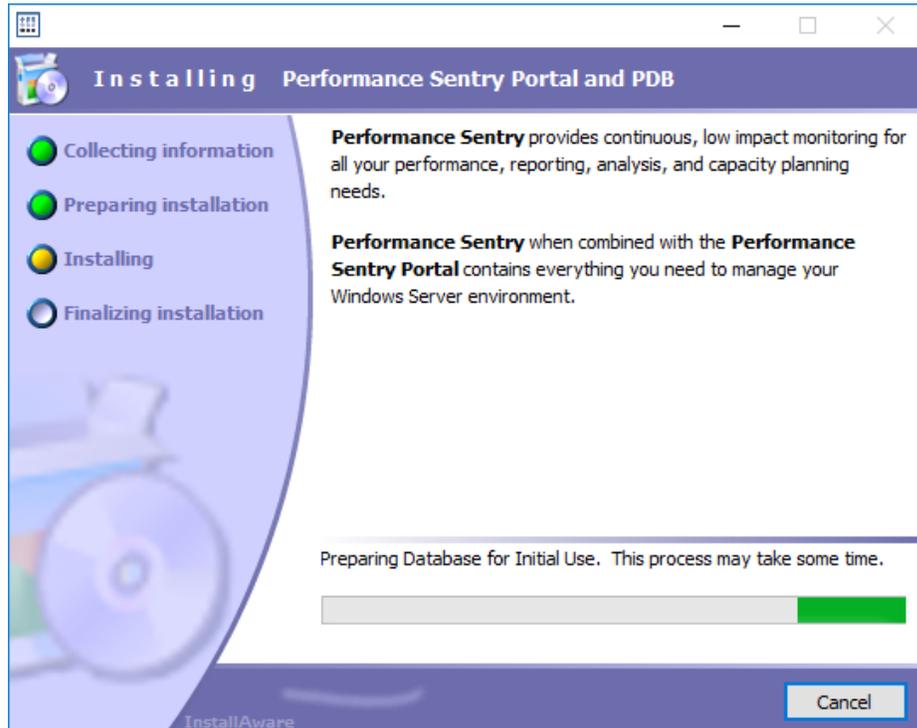
Quick Start Guide: Web Portal and PDB for Performance Sentry

After the pre-requisite applications and the Portal website are installed, the Performance Database (PDB) destination folder window is displayed:

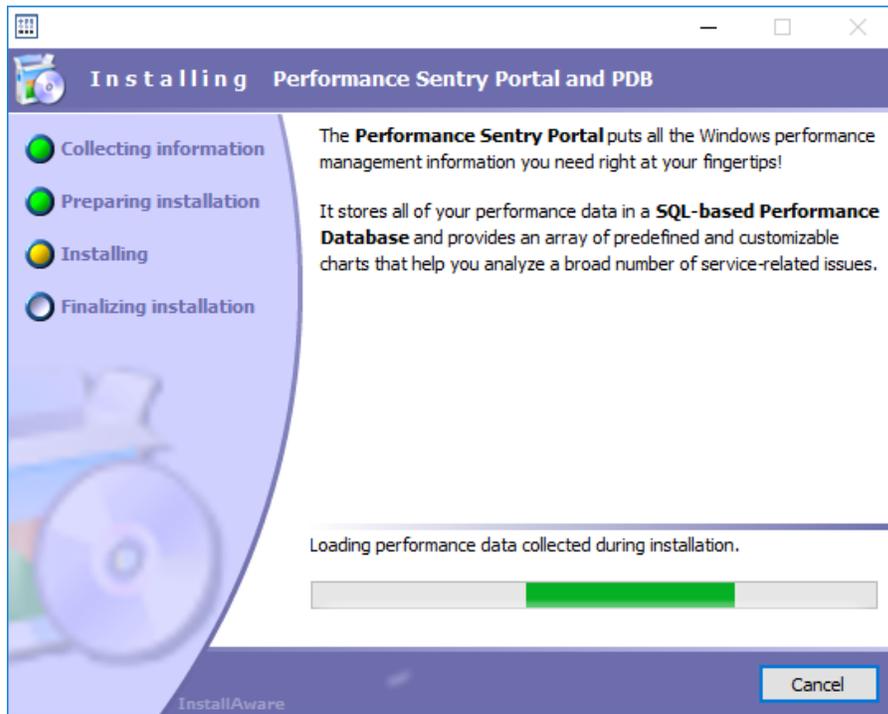


12. Browse to a desired folder, or accept the default and click **Next**.

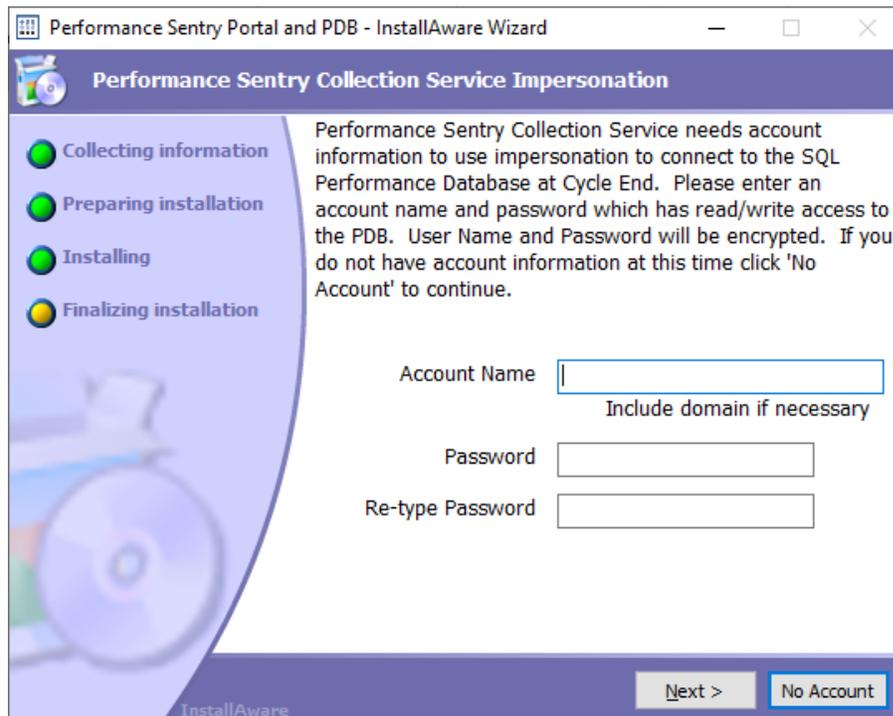
The database will be defined in the folder specified above and sample data from a machine named SQLServer will be loaded into the database:



The performance data collected during installation will then be loaded into the PDB

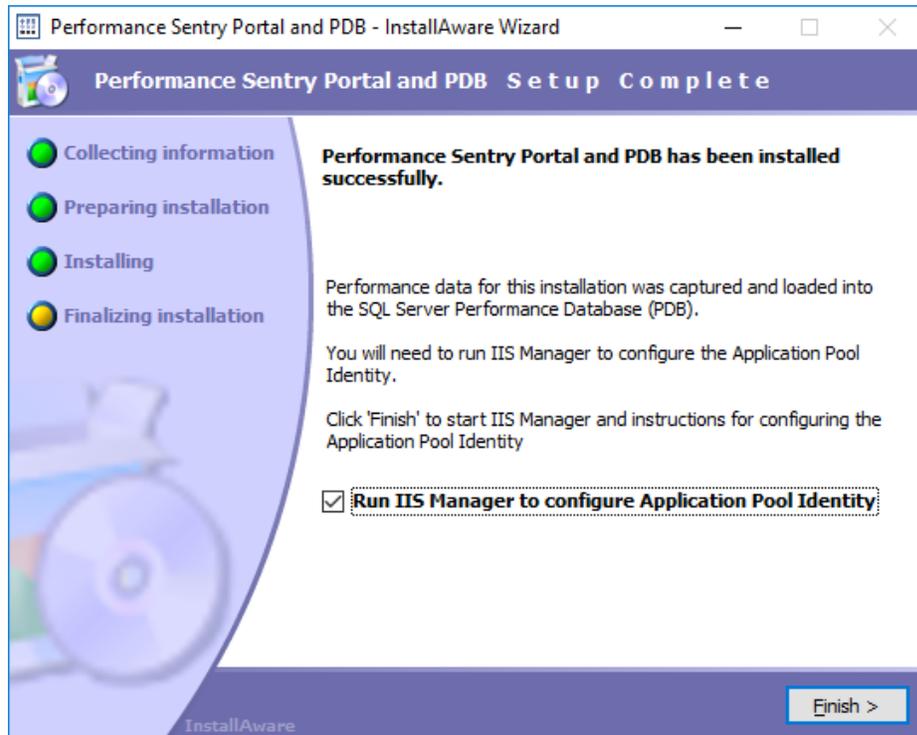


The Performance Sentry Collection Service Impersonation window displays. Please read the text in the following screenshot to understand the requirements. You can read more about impersonation on page 23 of the [Performance Sentry User Manual](#). Enter an Account Name and Password that has read/write permission to the SQL Performance Database specified earlier.



13. Click **Next** after typing the account name and password, otherwise click **No Account** to continue to the next screen without supplying an Account Name. Please note that if you do not supply a valid account name and password, then data collected by Performance Sentry will not be loaded into the SQL PDB on a daily basis at Cycle End. You can enter account information anytime using the Performance Sentry command line interface. Please read the Performance Sentry User Manual for more information regarding the command line interface, impersonation, and Cycle End processing. Contact support@demandtech.com if you have additional questions.

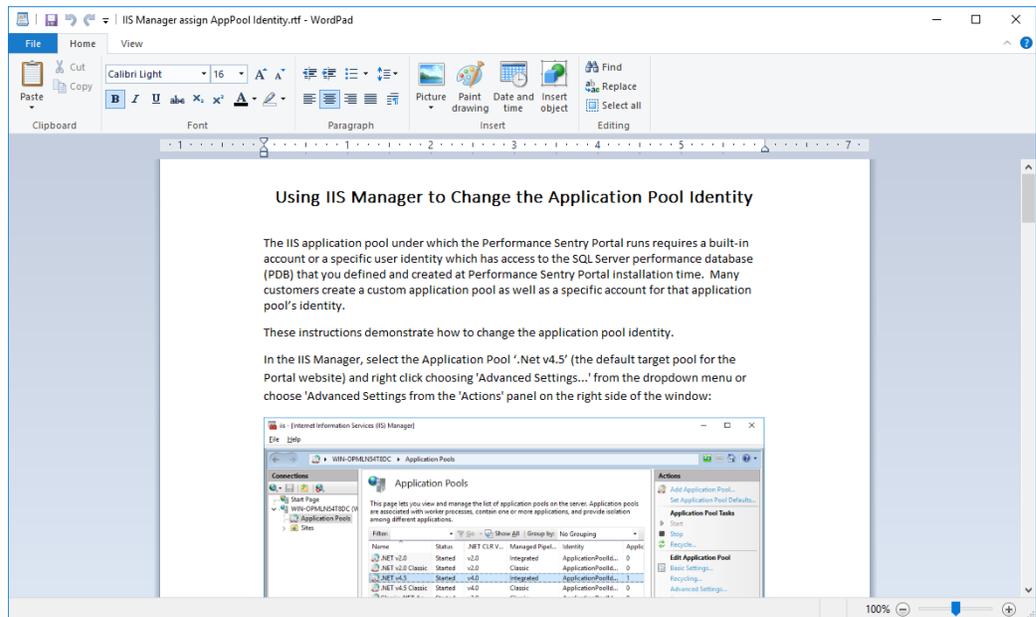
The Performance Sentry Portal and PDB Setup Complete window displays:



Congratulations! You have successfully installed Performance Sentry and performance data has been collected while the installation took place and loaded into the PDB.

Performance data will continue to be collected by the collection service and by default will be loaded into the Performance Database (PDB) at 3 a.m. each morning.

14. Click **Finish** to run IIS Manager to configure the Application Pool Identity. Also, WordPad will launch to display instructions describing how to change the application pool identity to meet your company's security requirements:



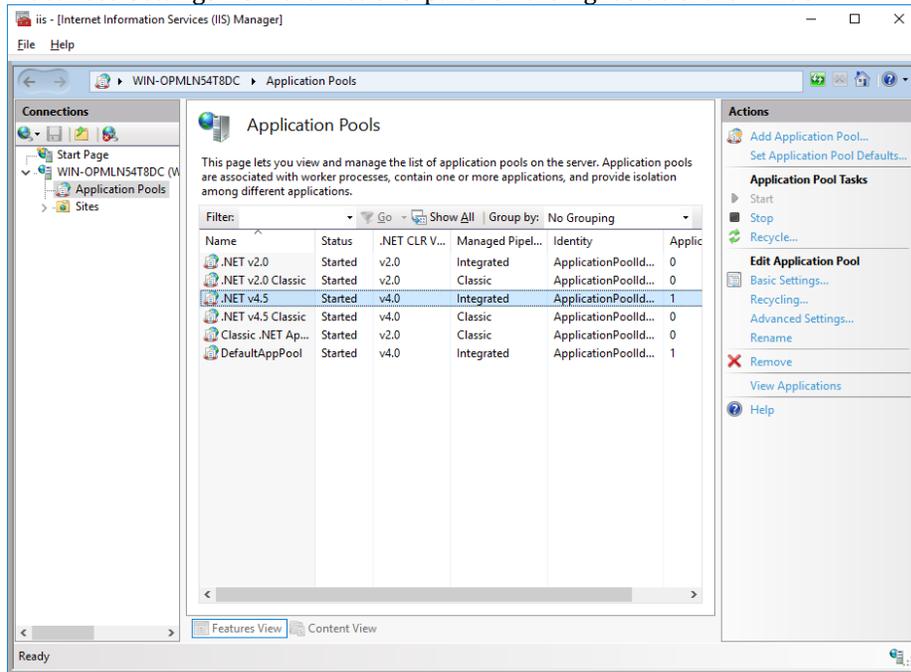
Using IIS Manager to Change the Application Pool Identity

The IIS application pool under which the Performance Sentry Portal runs requires a built-in account or a specific user identity which has access to the SQL Server Performance database (PDB) that you defined and created at installation time. Many customers create a custom application pool as well as a specific account for that application pool's identity.

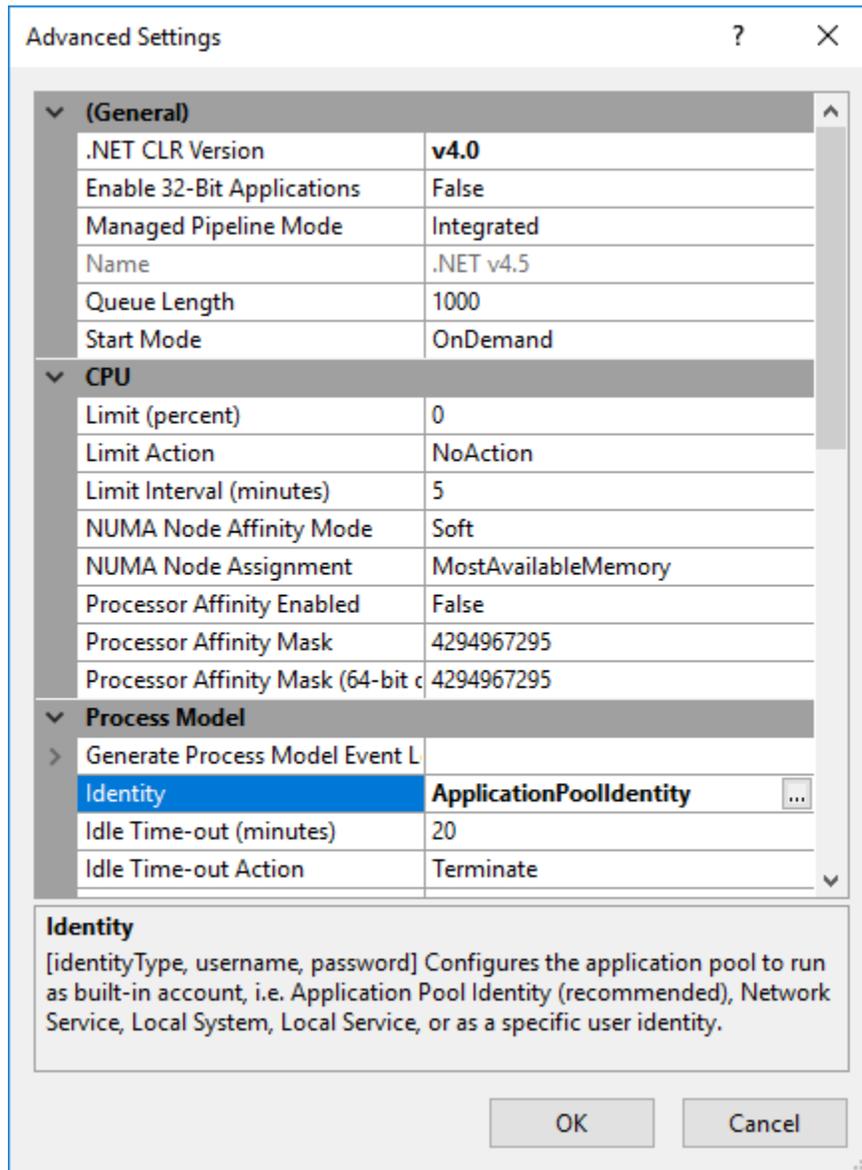
The following instructions demonstrate how to change the application pool identity.

Quick Start Guide: Web Portal and PDB for Performance Sentry

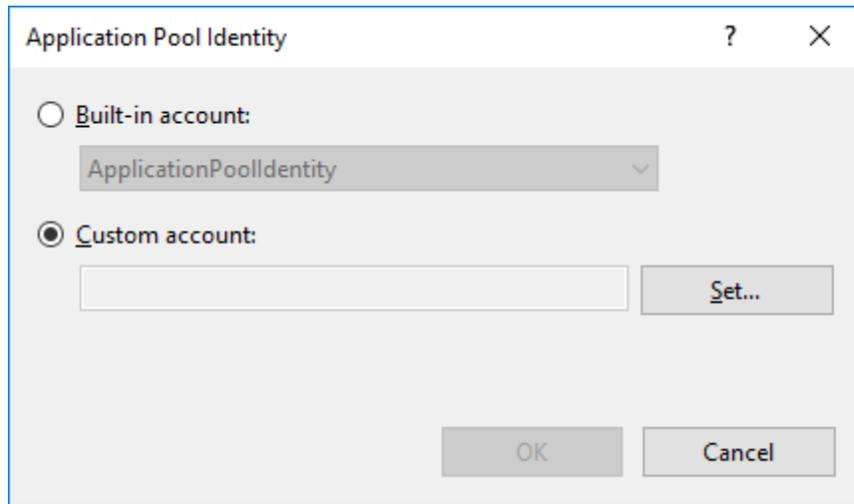
In the IIS Manager, select the Application Pool '.Net v4.5' (the default target pool for the Portal website) and right click choosing 'Advanced Settings...' from the dropdown menu or choose 'Advanced Settings from the 'Actions' panel on the right side of the window:



Highlight 'Identity' under 'Process Model' in the settings window, then click on the box to the right of 'ApplicationPoolIdentity' containing three ellipses (...)



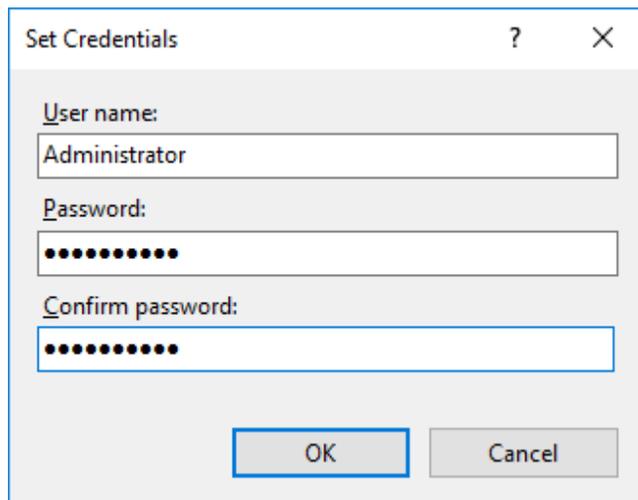
Click on the radio button next to 'Custom Account' then click 'Set'



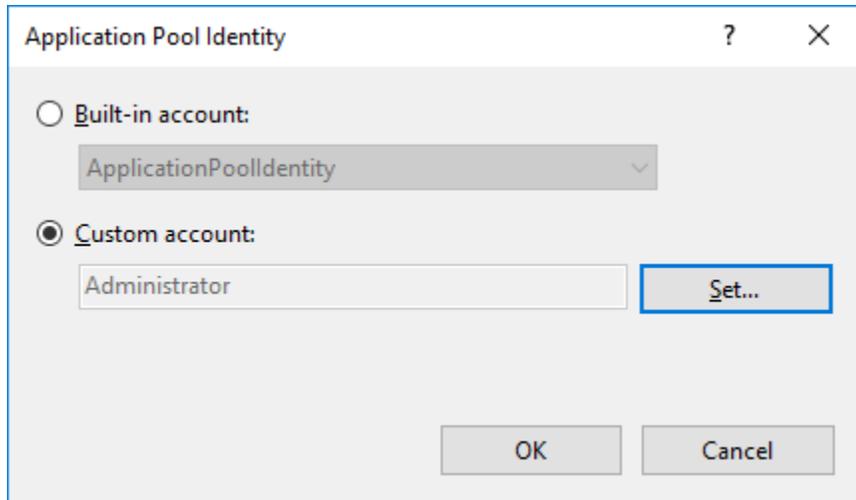
You need to use a built-in account or a specific user identity which has Data Base Owner (dbo) access to the SQL Server performance database (PDB) that you defined and created at Performance Sentry Portal installation time. The Web Portal contains a set of Data Base Administration (DBA) panels - which you can use the web site's built-in security model to control access to - that are used to perform most common PDB DBA tasks, which include maintaining security and setting up the Machine Group definitions that help simplify customer access to specific sets of machines.

A good way to verify that the Web Portal running under IIS will be able to access the PDB is to install the SQL Server Management Studio (SSMS) app from Microsoft (which can be downloaded [here](#)) and try connecting to the PDB using the *same credentials* that you intend to use in the IIS Application Pool settings, that you are configuring to run the Web Portal.

The example below uses the 'Administrator' account. Many customers prefer to create a specific account for the application pool identity like 'NTSMFPDB' that has limited access rights specific to the Performance Sentry Portal website and SQL server performance data base (PDB):



Click 'OK'



Click '**OK**'

You will now see the account in the 'Identity' field of the advanced settings (see next page):

Advanced Settings

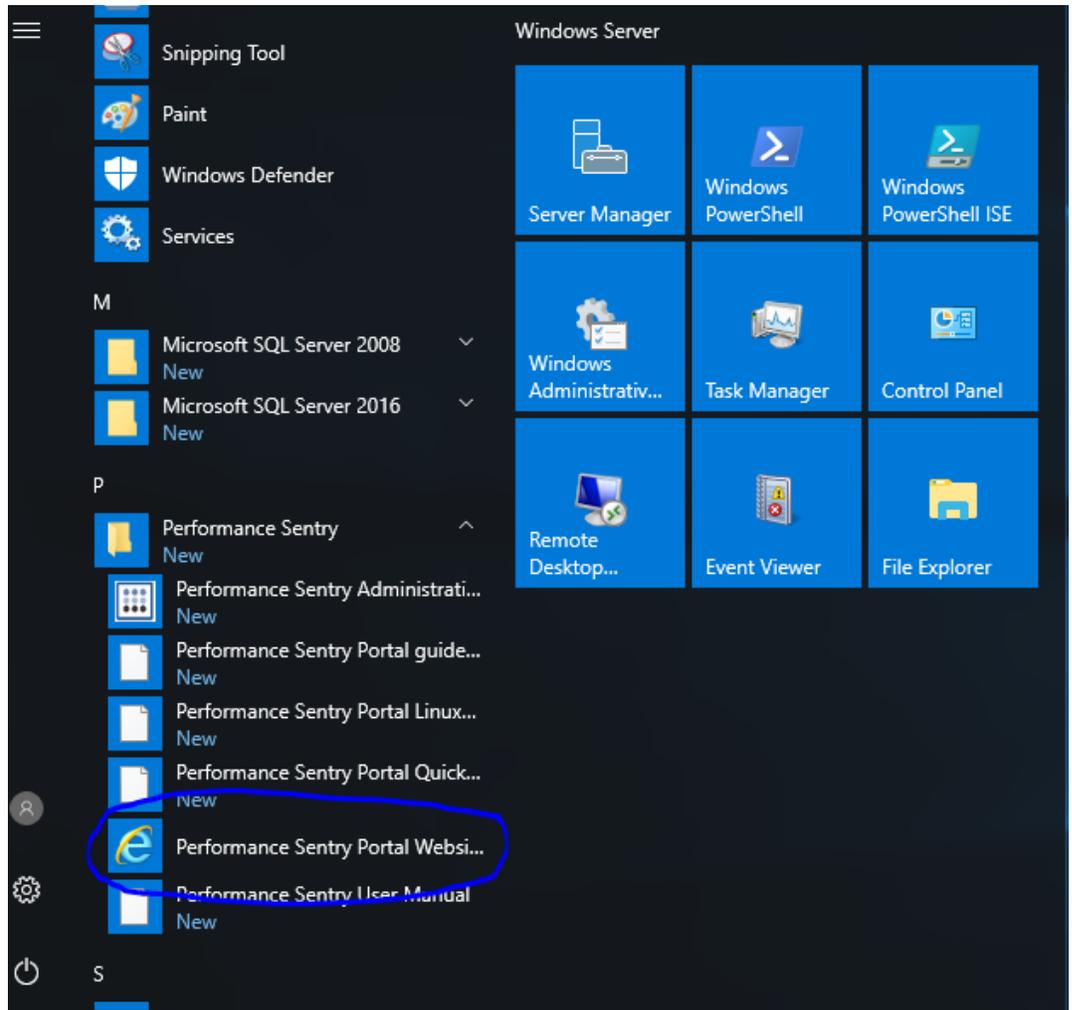
General	
.NET CLR Version	v4.0
Enable 32-Bit Applications	False
Managed Pipeline Mode	Integrated
Name	.NET v4.5
Queue Length	1000
Start Mode	OnDemand
CPU	
Limit (percent)	0
Limit Action	NoAction
Limit Interval (minutes)	5
NUMA Node Affinity Mode	Soft
NUMA Node Assignment	MostAvailableMemory
Processor Affinity Enabled	False
Processor Affinity Mask	4294967295
Processor Affinity Mask (64-bit c	4294967295
Process Model	
Generate Process Model Event L	
Identity	Administrator
Idle Time-out (minutes)	20
Idle Time-out Action	Terminate

Identity
[identityType, username, password] Configures the application pool to run as built-in account, i.e. Application Pool Identity (recommended), Network Service, Local System, Local Service, or as a specific user identity.

OK Cancel

Run the Performance Sentry Portal

You have successfully changed the Application Pool Identity and can now start the Performance Sentry Portal from the start menu:



Setup Security Administration

When you access the Web Portal for the first time, the first page displayed is the Security Administration panel, where you will be prompted register as the Administration of the application, with the capability of controlling who can access the various Portal panels. You will see the following screen, where you are prompted to register your Windows login ID as the web site Administrator.

Web Portal Security Administration

No User Activity Log records found for 2/15/2018.
Default security Profiles successfully initialized.

Set New User Security Policy: (0) No internal security restrictions

Complete Security Administration by Registering REDHOOK\markf as the Web Administrator.

Register New Users:

New User Name: REDHOOK\markf Security Profile: Administrator (1)

Registered Users

User Activity (from Audit trail)

Date	Total Events	Logins	Logins Denied	Access Denied	New User Logins	Security Admin
2/15/2018	0	0	0	0	0	0

Audit Report Date Selection:

February 2018

Su	Mo	Tu	We	Th	Fr	Sa
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	1	2	3
4	5	6	7	8	9	10

Security Audit Detail Report No Filtered User Activity Log records found for 2/15/2018.

Audit Log Filter:

Access Denied
 Successful Logins Blocked
 Logins
 New Registrations
 Security Administration
 All Audit events

Security log maintenance:

older than 35 days

Press the “Register New User” button to complete Security Administration initialization. You should see a confirmation message as follows:

Quick Start Guide: Web Portal and PDB for Performance Sentry

Web Portal Security Administration
 Security Administration setup complete. REDHOOK\markf added as Administrator.
 Use this panel to change the Security Policy and Register new Users.

Set New User Security Policy: (4) Open security: SuperUser access privileges granted to all new Users

Register New Users:

New User Name: REDHOOK\markf Security Profile: Administrator (1)

Registered Users

User Name	User ID	Role	Permission	Modify Permissions
REDHOOK\markf	1	Administrator	7	REDHOOK\markf

User Activity (from Audit trail)

Audit Report Date Selection:

February 2018						
Su	Mo	Tu	We	Th	Fr	Sa
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	1	2	3
4	5	6	7	8	9	10

Date	Total Events	Logins	Logins Denied	Access Denied	New User Logins	Security Admin
2/15/2018	0	0	0	0	0	0

Security Audit Detail Report No Filtered User Activity Log records found for 2/15/2018.

Audit Log Filter:

Access Denied
 Successful Logins Blocked
 Logins
 New Registrations
 Security Administration
 All Audit events

Security log maintenance:

older than 35 days

Notice that the panel displays a table showing each Registered User and the Role associated with that User. You have just been registered as having the Administrator role.

Web site security is governed by one of five security policies:

- **no security:** all Users can access all web pages
- **strict security.** No web access is permitted unless the User is pre-registered. Strict security requires the web site Administrator or Security Administrator to register all Users in advance, assigning them to a security role manually prior to them being able to access any of the web site reporting and analysis functions.
- **relaxed security,** which automatically registers all new Users and assigns them to a Guest role, allowing them access to most reporting and analysis functions
- **moderate security,** which automatically registers all new Users and assigns them to a User role, allowing them access to most reporting and analysis functions

- **open security**, which automatically registers all new Users and assigns them to the SuperUser role, allowing them access to all reporting and analysis functions, including defining machine Reporting groups and assigning which machines are associated with which Report Groups.

The default policy is **open security**, which simply requires defining a web site administrator (which you just did). The open security policy automatically registers all new authenticated Users and assigns them to a SuperUser role that grants them permission to can access to all of the Portal reporting and charting functions. Note that anyone who is not an authenticated User that attempts to access the Portal will have that access denied by normal IIS security.

Under the default security policy, SuperUsers can utilize the Portal for reporting, but are excluded from accessing any web site administration pages, including the Security Administration page. Any user that attempts to access an Administrative web page without being granted the Permission to do so, will denied access to the page and then bounced back to the Portal's main menu Home Page.

To authorize someone who has a SuperUser role assigned, you can use the Security Administration panels to modify manually the security profile assigned to that User to grant them additional privileges or even change their default security Role to Administrator or Security Administrator.

If the open security policy does not satisfy your requirements, you can apply a more restrictive security policy. If security administration is not necessary in your environment, you can change to the **no security** setting to turn off all permission checking.

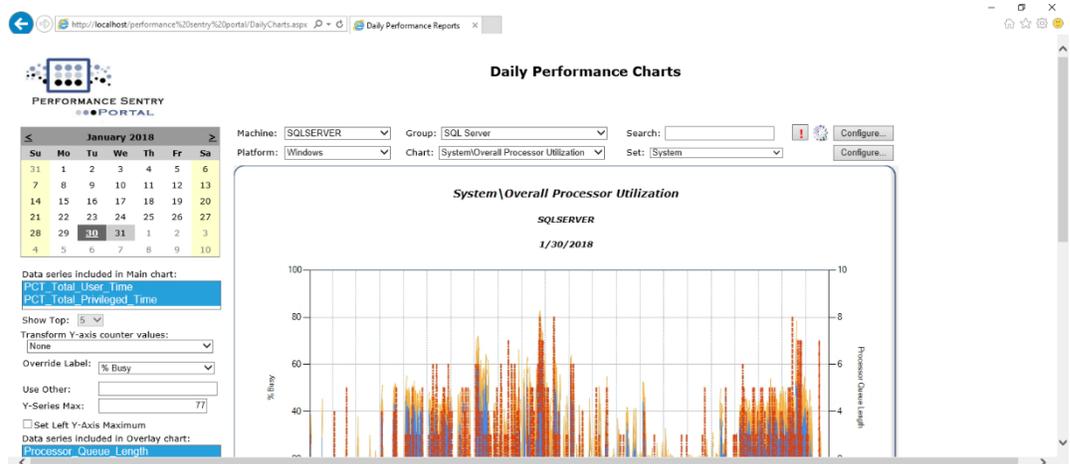
Quick Start Guide: Web Portal and PDB for Performance Sentry

After setting up Security Administration for the first time, click on the Demand Technology Software icon displayed in the upper left-hand corner of every Web Portal page

The Performance Sentry Portal Main Menu displays in your web browser.



Click **Daily Charts** and the System\Overall Processor Utilization chart will be displayed for SQLSERVER, a sample SQL Server machine loaded at installation time:



The Daily Charts can report on activity for a single machine for a single day. The calendar control on the left side of the panel positions itself automatically to yesterday's date or the oldest date associated with the selected machine.

The Daily Charts are designed to be of presentation quality and are rendered by a standard ASP.NET Chart control as .jpg files that you can copy and paste into another application like Microsoft Word or PowerPoint.

Each chart template defines a main chart and an associated box plot chart. The main chart contains one or more performance counter data series, for the date and time period indicated. The main chart is often rendered as an Area chart or, in the case of multiple chart data series that can logically be added together, as a stacked Area. You can change the default chart type, as defined in the chart template, to any of the supported chart types: these include line, point, and horizontal bar charts. The main chart can also display one or more overlay data series, which are plotted against the right-hand y-axis. The x-axis always represents the Time of Day – the performance counters are time series data.

Quick Start Guide: Web Portal and PDB for Performance Sentry

Across the top of the main chart are two rows of controls. The upper set of controls is for selecting the Machine from the PDB that you wish to report on. The **Configure** button on the top row allows you to navigate to one of the Administrative panels where you can define Machine Groups and assign Machines to those groups.

The second row of controls is for chart selection. The **Configure** button on this row navigates to the Chart Definition Editor panel where you can modify existing chart definitions or create new ones.

On the left side of the panel, immediately below the calendar date selection control are a set of controls for changing how the current chart looks. These are temporary changes that do not affect the current chart template definition. You can exclude some of the counter data series from being displayed on the chart, set Y-axis maximum values (which is useful if a few outliers make it difficult to understand what is going on), supply a Y-axis scaling factor, and override the default Y-axis label (which is especially useful if you do apply a Y-axis scaling factor). Once you have set all these controls to your satisfaction, click the **Redraw** button to apply your changes to the current set of counter data.

The left panel also contains a dropdown selection control that allows you to view the chart data using a different chart type.

All the Daily Charts plot counter data on the y-axis against Time of Day measurements on the x-axis. If you want to view the chart data in a different form, you will need use a third-party charting package, such as Microsoft Excel. If you click the “Export Chart Data” you will be able to export the counter data from the current chart to a .csv file that you can easily import into Excel to create, for example, an x-y scatterplot showing the relationship between two or more sets of counter data measurements. Note that the first column of the .csv exported file contains the time stamp in clock timer units that are compatible with Excel. In order for Excel to recognize that these values are clock values, you must

- Select the first column and instruct Excel to apply a date/time Format for the data values, then
- Save the spreadsheet as an Excel file (.xls or .xlsx) to preserve the formatting change

Immediately below the main Chart is a set of controls for manipulating the Time Window for the current set of data being selected for charting.

Finally, in the bottom center of the panel is a Box plot chart for the metric selected showing the distribution of values each of hour of the selected date. *Box plots* – sometimes also called *box and whisker* plots – are low resolution, graphical representations of a distribution: the box is drawn around points at the 25th and 75th percentile values, bisected at the mean and median; the whiskers extend the box to the 5th and 95th percentile values, while outlier observations are plotted discretely.

Besides graphically representing the distribution of counter values observed each hour, each hourly box plot also serves as a selection menu. If you click on the hourly box chart, it will center the Main Daily chart around the selected hour.

For a more in-depth discussion of the Daily Performance Charts facility, see the section entitled “[Using the Performance Sentry Portal](#)” later in this document.

Quick Start Guide: Web Portal and PDB for Performance Sentry

Next, take a look at Server Health reports. Click on the **Server Health** icon on the home page.



<http://localhost:performance%20entry%20portal/HealthReport.aspx>

The 'Server Health Report' page will be displayed.

In the **Server Health Report** screen, you will notice SQLSERVER in the **Machine** dropdown box. SQLSERVER is a sample machine which was loaded at installation and assigned to the 'SQL Server' machine group found in the **Group** dropdown box. To learn more about machine groups, see the document named **Using Machine Groups effectively in the Performance Sentry PDB** for more information on both machine group definitions and assigning machines to machine groups. A shortcut to this document is installed in the Start Menu at installation.

Click on the 'Server' icon below the 'Search' text box to select the SQLSERVER machine listed in the Machine dropdown box and to propagate the date picker with valid dates for that machine:



Quick Start Guide: Web Portal and PDB for Performance Sentry

Choose the previous day's date in the **Select Report Date:** date picker (it should be selected by default) and click the **Build** button to the right of the **Select Report:** dropdown box displaying 'SQL Server Health Report':

PERFORMANCE SENTRY PORTAL

Server Health Report

Machine: **SQLSERVER**
Machine SQLSERVER selected.

Please select the report date and press the Build button to generate.

Select Machine:
Machine:
Group:
or
Search:

Select Report:
 Build

Object	Counter	Instance	Min	Ave.	Median	Max	95%	98%	99%	n
--------	---------	----------	-----	------	--------	-----	-----	-----	-----	---

Select Report Date:

A health report for SQLSERVER is dynamically created and displayed in the browser:

Server Health Report
Machine: SQLSERVER
Machine SQLSERVER selected.
Report Date 1/30/2018; Start Hour: 0; Duration: 24

Select Report:
 SQL Server Health Report [Rebuild] [Configure...] [Navigate...]
 Server-side processing complete!

Select Machine:
 Machine: [SQLSERVER] [v]
 Group: [SQL Server] [v]
 or
 Search: []

Object	Counter	Instance	Min	Ave.	Median	Max	95%	98%	99%	n
System	Processor Queue Length		0.0	0.897	0.0	8.000	5.000	5.000	6.000	1316
Processor	% Processor Time	[SUM]	5.350	211.812	175.320	662.700	447.890	527.870	569.680	1316
PhysicalDisk	% Disk Busy	18 F:	0.027	24.434	20.860	94.930	60.250	77.580	83.450	670
PhysicalDisk	% Disk Busy	10 F:	0.048	21.691	19.130	78.320	53.560	65.250	70.410	699
Memory	Pages Input/sec		0.130	4.890	0.150	1316.440	0.680	15.370	61.100	1316
Process	% Processor Time	sqlservr (2668)	5.080	186.855	147.680	630.340	411.510	481.930	522.970	1316
Process	Working Set	sqlservr (2668)	330170368	336131828	337678336	345219072	339415040	340070400	344420352	1316
Process	Ready Threads	sqlservr (2668)	0.0	0.061	0.0	4.000	0.0	1.000	2.000	1316
SQLServer:Buffer Manager	Page reads/sec		0.0	1031.474	648.750	7404.750	3692.810	4438.440	5085.720	1316
SQLServer:Buffer Manager	Page writes/sec		0.0	1033.207	555.090	7356.580	3733.420	4488.560	5129.910	1316
SQLServer:Buffer Manager	Page life expectancy		68.000	6868.323	3441.000	63943.000	26004.000	35419.000	38127.000	1316
SQLServer:Buffer Manager	Lazy writes/sec		0.0	0.745	0.0	40.870	3.100	9.030	17.170	1316

Congratulations! You have successfully installed and configured Performance Sentry Portal to produce charts and server health reports from the Performance Database (PDB).

Continue reading to learn how process and view data from additional Windows servers. You can skip ahead to the section [Using the Performance Sentry Portal](#) for instructions on how to customize charts and server health reports to meet your needs.

Viewing Performance Data from Other Windows Servers

Once you've seen the depth of performance data that Performance Sentry can collect, you'll want to gather data from other computers as well. How do you do this? You will need to install the collection service on additional Windows servers using the collection service setup routine (CSSetup.exe) that was installed during the Performance Sentry Administration component installation. CSSetup.exe can be found in the following folder:

C:\Program Files (x86)\Performance Sentry Administration v4\collection service files

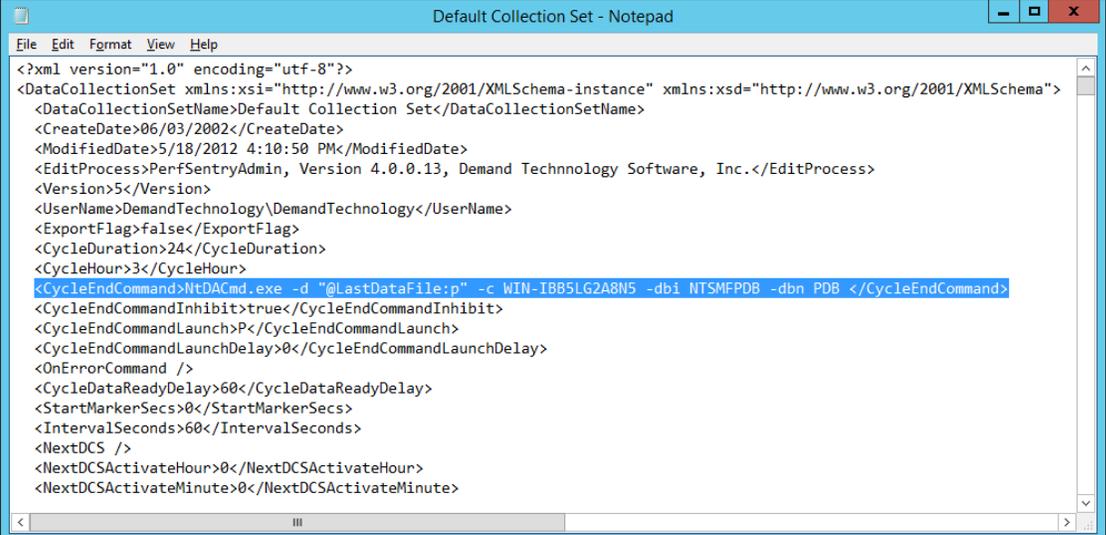
Also in that folder are modified Data Collection Sets (DCSs) which are files containing parameters that are used by the Performance Sentry Collection Service to determine which performance counter values to collect, how often to collect them and what to do with them once they have been collected. 'What to do with them' is the key phrase, as each of these DCSs have been modified to load the data directly to the Microsoft SQL Server Performance Database (PDB) once a day (at 3a.m. by default) using the ntdacmd.exe program with parameters pointing to the newly defined PDB. You can open the XML-formatted DCS files (with a file extension of DCSX) in notepad and examine the cycle end command yourself. It has the format:

```
<CycleEndCommand>NtDacmd.exe -d "@LastDataFile:p" -c computername -dbi
databaseinstancename -dbn databasename </CycleEndCommand>
```

where

computername is the server where the PDB database resides,
databaseinstancename is the SQL Server Instance under which the PDB runs, and
databasename is the name of the performance database (PDB)

Here is a screenshot showing the use of notepad to examine the 'Default Collection Set.DCSX' file with the CycleEndCommand element highlighted:



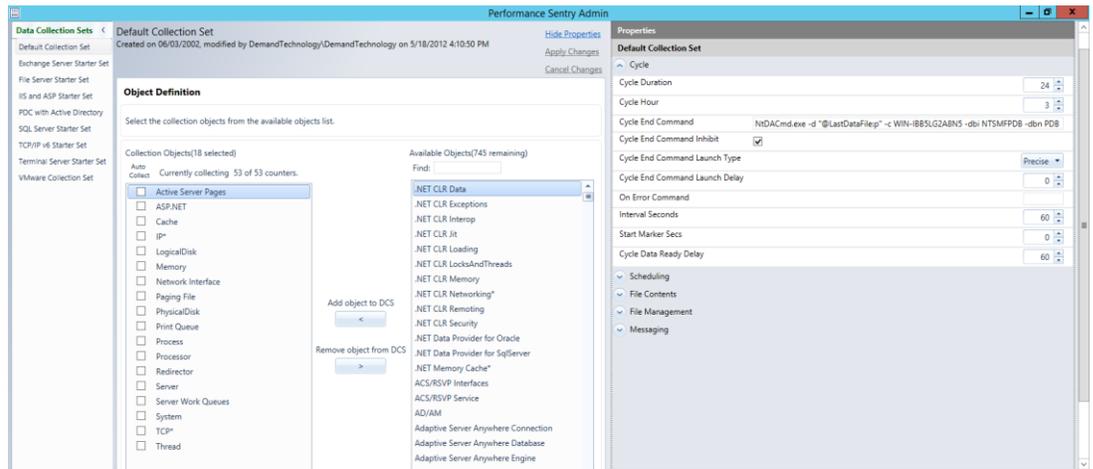
```

Default Collection Set - Notepad
File Edit Format View Help
<?xml version="1.0" encoding="utf-8"?>
<DataCollectionSet xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <DataCollectionSetName>Default Collection Set</DataCollectionSetName>
  <CreateDate>06/03/2002</CreateDate>
  <ModifiedDate>5/18/2012 4:10:50 PM</ModifiedDate>
  <EditProcess>PerfSentryAdmin, Version 4.0.0.13, Demand Technology Software, Inc.</EditProcess>
  <Version>5</Version>
  <UserName>DemandTechnology\DemandTechnology</UserName>
  <ExportFlag>>false</ExportFlag>
  <CycleDuration>24</CycleDuration>
  <CycleHour>3</CycleHour>
  <CycleEndCommand>NtDacmd.exe -d "@LastDataFile:p" -c WIN-IBB5LG2A8N5 -dbi NTSMFPDB -dbn PDB </CycleEndCommand>
  <CycleEndCommandInhibit>true</CycleEndCommandInhibit>
  <CycleEndCommandLaunch>P</CycleEndCommandLaunch>
  <CycleEndCommandLaunchDelay>0</CycleEndCommandLaunchDelay>
  <OnErrorCommand />
  <CycleDataReadyDelay>60</CycleDataReadyDelay>
  <StartMarkerSecs>0</StartMarkerSecs>
  <IntervalSeconds>60</IntervalSeconds>
  <NextDCS />
  <NextDCSActivateHour>0</NextDCSActivateHour>
  <NextDCSActivateMinute>0</NextDCSActivateMinute>

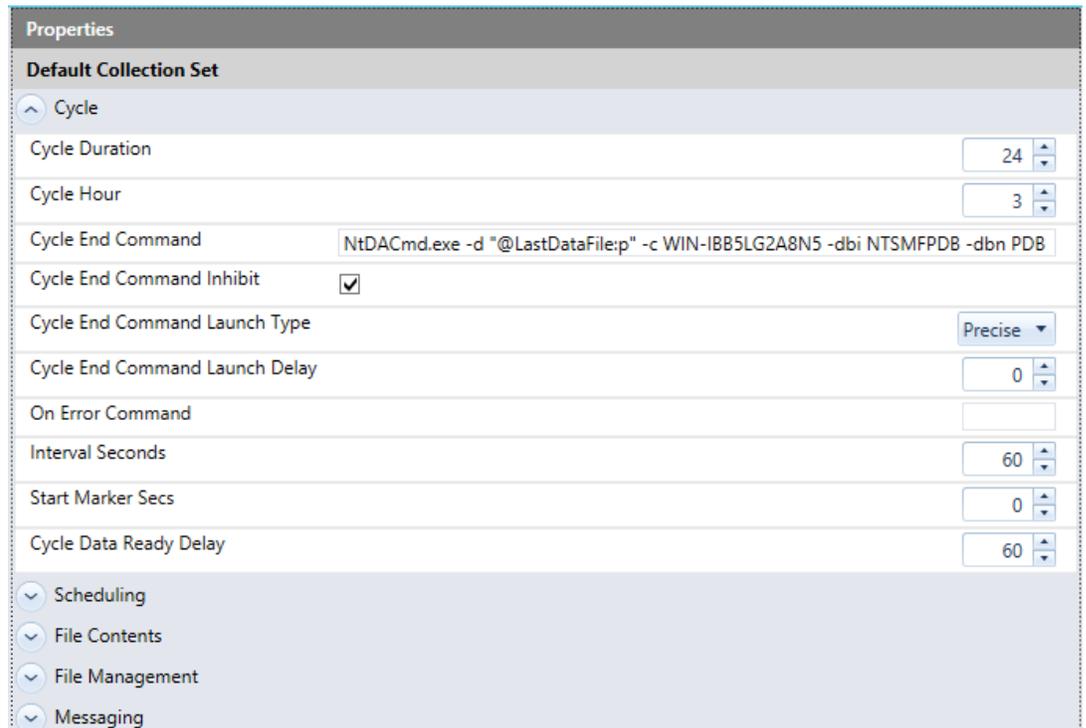
```

Quick Start Guide: Web Portal and PDB for Performance Sentry

You can also view the cycle end command using Performance Sentry Administration and clicking on 'Properties' in the upper right-hand side of the window:



Here is a close-up view of the Properties section:



To install the collection service on additional machines, simply copy the installation executable, CSSetup.exe, and the accompanying DCSX files to the target machine (or link to the folder from the target machine) and start CSSetup.exe. Setup will install the Performance Sentry Collection Service along with the modified DCSX files and will start the Performance Sentry Collection service using the 'Default Collection Set' DCS file.

At cycle end (3a.m. by default) the collection service will close the current data collection file, open a new one, then load all of the data files created the previous day (found in the 'Previous' sub-folder) to the PDB. To learn more about Performance Sentry Administration and Performance Collection Service operation, see the **Performance Sentry User Guide** which can be opened from the Start Menu shortcut or found in the 'Documents' subfolder of the Performance Sentry Administration folder.

Viewing Linux Data

If you would like to view Linux data in the Portal, you must first collect the data and load it into the Performance Database. To collect Linux data you must install open source collection routines on each Linux machine and define an ODBC connection string on the computer hosting the SQL Performance Database (PDB).

Instructions to do this are in the document titled "Quick Start Guide: Linux Performance Data Collection and Processing into PDB."

Using the Performance Sentry Portal

The Performance Sentry web portal provides access to the NTSMF Performance Database (PDB) for both reporting and analysis. It supplies the following the set of reports:

- [Daily Performance Charts](#) is a powerful database query tool that allows you to create and view presentation-quality graphical reports for the analysis of Windows, VMware and Linux performance problems. It is limited, however, to reporting data for a single 24-hour period for one machine at a time.
- Management by Exception reports, using alert data gathered by the Performance Sentry performance data collector.
- A Machine Configuration and Inventory report documents the static hardware and software characteristics of each and every Windows machine for which performance data was gathered and processed through the PDB.

You normally access the Daily Performance Charts and Alerts summary report from the Portal's default.aspx landing page, default.aspx, which was illustrated on page 18 of this document. From the DailyCharts.aspx page, you can also navigate to the configuration options where you can define new chart templates, or modify existing ones, using the Chart Template Editor. Chart templates are stored in the PDB alongside the performance counter data. The [Chart Definition utility panel](#) can also be used to import chart templates, as illustrated on page 19-20 of this document, or export chart templates to copy the templates from one copy of the Portal to another.

From the main Alerts summary report, you can drill into the alerts that fired for specific machines. You can also navigate from specific alerts to the Daily Performance Charts, where a PDB query will be issued automatically for data that is representative of conditions that were current at the time the specific alert was triggered.

The quickest way to navigate to a Machine Configuration report is from the main PDBDataAdministration page, but it can also be reached by selecting the Machine name in several other reports designed for the PDB data administrator.

The Portal also provides tools for PDB data administration, including facilities to monitor the daily update of the PDB, which is usually scheduled to run overnight. It also includes panels for defining Machine Groups, and assigning machines to them, a feature that is designed to help you navigate to specific machines easily.

The sections that follow document the individual web pages that comprise the Performance Sentry Portal.

Default.aspx landing page

The default landing page is a simple graphical menu that allows you to choose between the Daily Performance Charts, the daily Alerts summary report or PDB Data Administrator tasks.



You can return to the default menu at any time by clicking the logo that is displayed in the upper left hand corner of every Portal web page.

Daily Performance Charts

The Daily Performance Charts page, `DailyCharts.aspx`, displays a high-resolution graphical report built from performance counter data stored in the PDB. The web page displays a Main Chart and a Box and Whiskers Plot, plus controls for changing the date and time range of the data that is displayed. If the chart template defines multiple sets of time series data to be displayed in the Main Chart, additional controls can be used to exclude any of these sets of counter data from the Main Chart. The Box plot displays the distribution of a single set of counter values by hour. It is positioned just below the Main Chart. (You may have to scroll down in the web page to see the Box plot in its entirety.)

All charts are limited to reporting on one machine at a time, and the report time window is limited to no more than one 24-hour day. You can narrow the report window on the Main Chart to a period as little as one hour, but the Box Plot is always displayed as a daily view.

Charts are built by querying the performance database to gather a specific set of counter data fields. The PDB queries are generated dynamically, based on instructions contained in a *chart template*, which specifies the PDB data fields to extract. The chart template also contains specific chart formatting instructions, including the y-axis labels to be used.

All counter data is charted in time series. Date and time selection controls allow you to select the date and define a time window for which you would like to view data. The charts can show data for as little as one hour or as long as a contiguous 24-hour period.

Database schema. A basic understanding of how the performance counter data that was gathered by the Performance Sentry collection agent is processed and then stored in the PDB in MS SQL Server Tables will help you make more effective use of the Portal's charting features.

Inside the PDB, SQL Server Tables are built for each machine. These *machine-specific* data Tables, a data Table for each *performance object* included in the collection set, are constructed initially the first time data from a machine is ever processed. Subsequently, when more data for that machine is loaded, the machine-specific data Tables that exist are validated by the PDB Loader program against the current set of format records in the `.smf` collection file to determine if any new data tables need to be built or if existing tables need to be modified to reflect any new performance objects and counters that are available.

Inside SQL Server, each machine-specific Table name is designated *machinename@objectname*. For example, if you have loaded performance counter for a Windows machine named `PROD1`, inside the SQL Server PDB, you will find Tables designated **PROD1@Processor**, **PROD1@LogicalDisk**, etc., that store the counter data gathered for this machine.

Each performance counter gathered is stored in a column in the Table, where the column name identifier is the name of the field. For example, the data for the Available Bytes counter, a field in the Memory performance object in Windows, is stored in a column in the *machinename@Memory* table in a column named **Available_Bytes**.

- ❖ Note that the SQL Server identifiers used for table names and column names cannot contain blank characters. Many other special characters in performance object and counter names like “%” or “#” are also invalid. So, the PDB data loader modifies the object and counter names used in SQL identifiers to conform to the naming conventions that SQL Server adheres to.

Quick Start Guide: Web Portal and PDB for Performance Sentry

Each Row in the machine data Tables stores the counter values for a set of counter data observations made at a specific point in time. A TimeStamp field identifies the precise time at which these counter data values were observed. In addition, there are also date and time columns in each row that identify the Date, Year, Month, Day and Hour the counter data observations were gathered.

A composite field designated as the RecordID is also constructed for each row of data stored in the machine-specific data Table. Together with the TimeStamp field, RecordIDs uniquely identify that row of counter data. The TimeStamp column is used to index the machine-specific SQL Server data Tables to make querying data in the PDB faster.

Some performance objects, like the Memory counters, are singletons – there is just one set of counters that reflects all memory activity in Windows for each measurement interval.

For many other performance objects, like Logical and Physical Disk or Process, you will find multiple *instances* of the counter set stored in the database, reporting measurements for each individual disk or process. In the PDB, there is a separate row for each instance of an *instanced* performance object. A unique Instance name is also stored in the corresponding data row to identify that set of performance counter measurements.

In the case of instances of the Process object, things are a little bit more complicated because the Process Instance name, which is the name of .exe file, is not guaranteed to be unique. Normally, there are multiple instances of the svchost.exe process or w3wp.exe web application worker process running, for example. The Process ID field is unique, however, so the PDB uses the Process name and ID to identify the data rows in the *machinename@Process* data table.

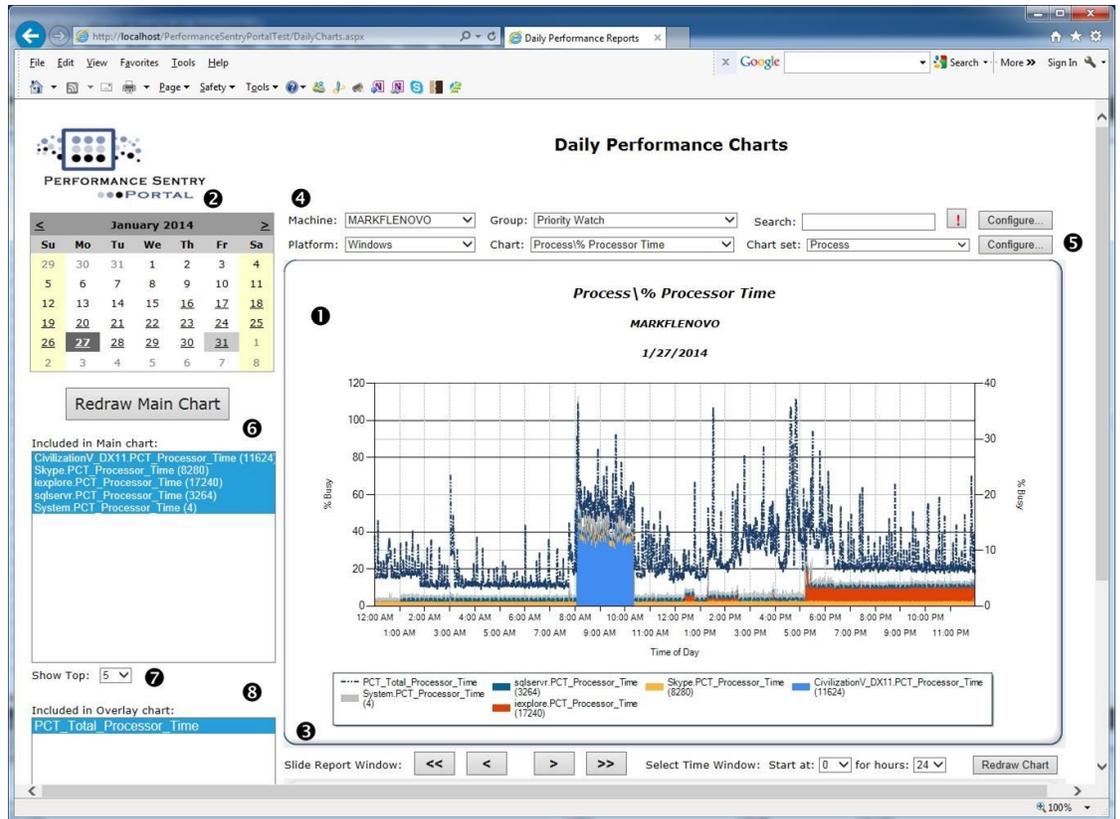
There are also performance objects that require both an instance name and a *parent* instance name to identify a unique set of time series counter data. For example, the Thread object uses an instance name based on the Thread ID and a parent name that is the name of the parent Process. In the case of the Thread object, the parent Process ID is also needed to identify a unique set of counter values.

In this fashion, when Windows counter data from a .smf data file is loaded, each data record the .smf data file becomes a single row of counter data that is inserted in the PDB

For more information on how the Performance Database schema is constructed dynamically, see the section in this document entitled "[How All of This Works.](#)"

With that understanding of the PDB database schema in mind, let's look at how the Daily Performance Charts are constructed and the options available for building charts, including the chart templates that define which counter data fields are selected for charting.

An example Daily Performance Chart is pictured below:



The description of the elements that the Daily Performance Charts web page contains is keyed to the numbers that overlay the example screen shot above.

1 MAIN CHART.

The Main Chart is displayed in the upper right portion of the web page. It is a high resolution graphic built on demand based on

- the counter data fields specified in the chart template
- for the machine selected, and
- the date and time range selected.

The Portal renders charts in the form of jpeg image files. They are built at the IIS web site where the Portal is installed. Image files are then downloaded from the web server to the web page, where they are displayed in the browser. You can easily cut and paste them from the web page to another application, such as an MS Word document or any other document type that can render a .jpg image.

The chart title contains the chart name, which defaults to the performance object/performance counter identifier for the main chart series, the name of the selected machine and the report date. Daily charts that span more than one calendar day are not supported in this release.

Tool tips. Using the mouse, hover over any data point plotted on the main chart to display a tool tip that shows the time of day value and the corresponding data value that was recorded.

Chart formatting. All counter data is graphed using the specified chart type over time. The time of day the counter data was gathered is plotted against the x-axis.

- ❖ Because all the counter data selected for the Main chart is always graphed over time, it is possible to mix and match any counter data from any machine-specific data table in the PDB.

Data values for the Main Chart data series are plotted against the primary y-axis (at left). The maximum y-axis value displayed is determined automatically based on the underlying data series being graphed, but it is possible to specify a y-axis max. You can also specify a logarithmic scale instead of the normal uniform scale for the y-axis. See the discussion of the [Chart Template Editor](#) for more details.

The chart template also defines the initial chart type for the Main Chart data series. It allows you to choose among the following chart types:

- Area
- Stacked Area
- Column
- Stacked column
- Line
- Point

Use stacked area or stacked column charts whenever it makes sense logically to sum the data values for each Main Chart data series. The resolution of the Main Chart plot area shows a maximum of 120 data points across the horizontal time of day axis, enough to represent two full hours of data precisely. Using column charts with reporting windows that are larger than two hours runs the risk of misrepresenting the underlying time series distribution.

The default chart type specified in the chart template for the Main Chart can also be overridden, using the Chart Type dropdown selection menu, located towards the bottom of the web page.

- ❖ Changing the chart type requires the entire chart to be regenerated at the web server.

Overlay series. Optionally, you can add one or more overlays to the Main Chart. Any overlay data series that are extracted are plotted against a secondary axis, so they are a good option whenever you want to report on data where the magnitude of the data values is quite different. For example, adding an overlay series representing the demand page rate works well for a stacked area chart that reports on bytes of physical memory that are allocated.

An overlay series can supply a crucial context that can aid in understanding or help clarify the interpretation of the Main Chart data series. For example, adding an overlay series representing the Processor Queue Length counter to a Main Chart that reports on processor utilization at the system, processor or process level adds significant detail to the utilization measurements. Adding an overlay series that shows the disk IO rate adds significant context to a Main Chart that reports device response time.

Each overlay series is graphed as a dotted line that is plotted against the secondary (right-hand) y-axis. The maximum y-axis value displayed on the secondary axis is determined based on the underlying data series being graphed. Similar to the primary y-axis, it is also possible to specify a secondary y-axis max and to specify a logarithmic scale for the secondary y-axis scale.

Chart legend. A chart legend is generated automatically based on the unique RecordIDs of the data series extracted from the PDB, unless the number of chart data series plotted exceeds the value of the **LegendAutoHideThreshold** setting, which is specified in the web.config App settings file. The factory default value for **LegendAutoHideThreshold** is 16.

2 DATE SELECTION CALENDAR CONTROL.

Clicking on a Day field in the calendar control allows you to change the report date.

Date selections in the calendar control are enabled for selection based on start and last timestamp fields that are maintained in the PDB's Machines table. The Machines table contains one row for every machine for which performance counter data is available.

- ❖ If there are any gaps in the sequence of counter data that was loaded in the PDB, it is possible for a report date to be selected for which no corresponding counter data values are present in the PDB. The Daily Performance Charts will display an error message when there is no counter data present in the PDB for the date selected, instead of simply displaying an empty chart.

When you select a new report date, the Portal web site is notified and immediately generates a new chart using counter data from the new date. The date selected is echoed in the 3rd line of the main chart title. 1

3 TIME WINDOW SELECTION CONTROLS.

The Time Window controls allow you to narrow the report window to intervals as small as one hour and change the report start hour, which defaults to Midnight of the selected date. After setting a new start hour or changing the report time window size, click the "Redraw Chart" button that is immediately to the right of these two controls.

You also click on four buttons that slide the report window that is defined forward one interval, backwards, fast forward by 24 hours, or backwards by 24 hours. Clicking on any of the Time Window slider buttons immediately generates a new chart using counter data from the new time range.

4 MACHINE SELECTION CONTROLS.

The machine selection controls allow you to select a new machine to report using the current chart template. The machine selection controls consist of

- a Machines dropdown list populated from the current Machine Group,
- a Machine Group selection menu, and
- a Search input field that supports auto-completion, based on matching against the first three characters that you typed into the Search box.

There is also an Execute button, which displays a "!" in red, immediately to the right of the Search text input field. To switch to the machine that is named in the Search input text box, click the Execute button.

Machine groups. Machine groupings make it easier to navigate in a large scale PDB that contains data from hundreds or thousands of Windows, VMware and Linux machines. You can define as many machine groups as necessary. Each machine which has counter data stored in the PDB can be assigned to as many as five machine groups, allowing for one machine group assignment for each of the five machine group categories.

If a machine you want to report on is *not* assigned to any machine groups, it will not be available for selection using any combination of the two Machine and Machine Group selection menus. However, you can always use the machine Search capability and create reports for a machine directly.

Or you can press the “Configure...” button at the far right in this set of machine access controls to access the PDB Data Administration panels to assign the machine to one of your machine groups. Try looking for the machine you want in the “Unassigned Machines” report or Search for the machine by name directly. See the document called **Using Machine Groups effectively in the Performance Sentry PDB** for more information on both machine group definitions and assigning machines to machine groups. A shortcut to this document is installed in the Start Menu at installation.

5 CHART TEMPLATE SELECTION CONTROLS.

Chart templates are used to specify which performance counter data to be queried to build the Main and Box Plot charts. Templates also specify additional formatting instructions for displaying the data, including the type of chart to build, and how to label the primary and secondary y-axis.

Chart template definitions are stored within the PDB in two tables, ChartTemplates and ChartSeries.

For convenience, chart templates are organized into sets. If you don't see the chart template you are looking for, change the current Chart set using the dropdown menu.

To modify the current chart template or create a new chart template from scratch, click the “Configure...” button at the far right in this set of chart template access controls. For more information on chart templates, see the section in this document that discusses the [“Chart Template Editor.”](#)

6 MAIN CHART SERIES SELECTION

Many of the database queries defined in the chart template can generate more than one data series to be plotted. Initially, every data series returned from the initial PDB query is plotted. In addition, each data series included in the main chart is identified in the main chart series selection List Box. This control supports selection of one or more items. Initially, when the chart is first built, all items are selected.

If you want to redraw the chart with some of the data series that were extracted excluded, use the mouse to highlight one or more of the original data series. Click the large “Redraw Main Chart” button to redraw the chart, using only the data series that you selected.

7 TOP N SERIES SELECTION

Many chart templates execute database queries on counters that reside in performance objects with multiple instances, such as the Physical Disk or Process objects. Database queries against instanced performance objects can reference specific instances by name, the C: drive, for example, or the dmperfss.exe process. Alternatively, the query can be directed against *any* instance of the object, in which case the database query generated automatically uses a TOP N directive in SQL with an ORDER BY clause based on the sum of the values of the counter specified over the query interval.

Use of the Top N directive is designed to limit the amount of data that a Main Chart displays, but there are situations where that limit may be too severe. By default, the query is for the TOP 5 data series, but you can use this selection box to change the maximum number of counter data series that are requested.

- ❖ Note that when instance names that were subject to Top N processing are added to the Main chart series selection List Box , they are added in sorted order, i.e., top-down, from highest to lowest.

Be aware that it is possible to specify a value for Top N that is greater than the total number of instances of the performance object that existed in the query interval. Changing the Top N value sends a request to the web portal to re-execute the database query immediately.

OVERLAY CHART SERIES SELECTION

Database queries defined in the chart template can generate more than one data series to be plotted as overlay lines. Initially, every overlay data series returned from the initial PDB query is plotted. In addition, each overlay data series included in the main chart is identified in the Main chart overlay series selection ListBox. This control supports selection of one or more items. Initially, when the chart is first built, all overlay data series are selected.

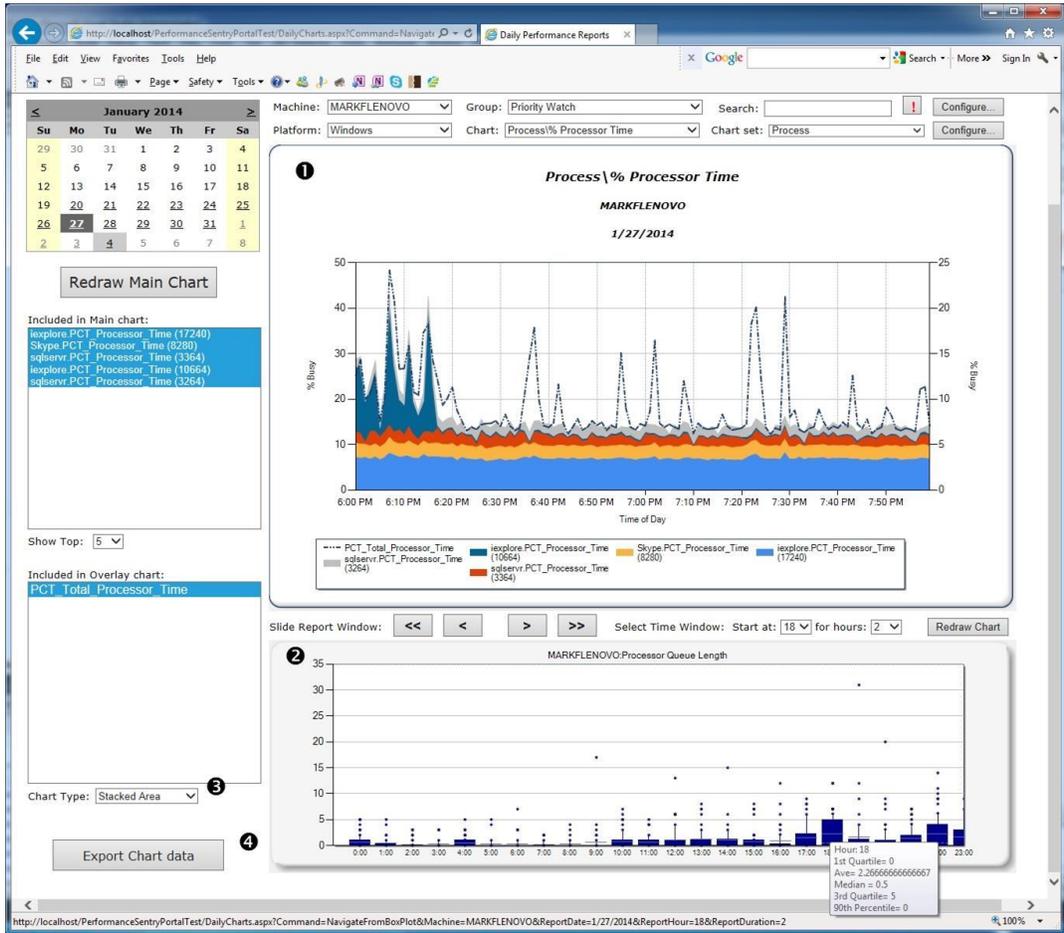
If you want to redraw the chart with some of the overlay data series excluded, use the mouse to highlight one or more of the original data series, then click the large “Redraw Main Chart” button to redraw the chart, using only the data series that you selected.

- ❖ To suppress plotting all overlay data series, use the [Chart Template Editor](#) to remove the chart series data definition from the template, and then click the “Apply” button.

Box Plot and more controls

If you scroll the web browser window down, you will be able to view the remaining elements on the Daily Performance Charts web page. These include the by hour Box Plot, plus some additional controls that let you temporarily change the chart type and export the chart data to a comma separated file that can be imported and then manipulated in Microsoft Excel.

The Box Plot and the additional controls are illustrated in the screen shot below:



2 BOX PLOT.

Each chart template optionally defines a counter value to be graphed using a visualization technique known as a box and whiskers chart, or a Box Plot for short. A box and whiskers chart is a compact visualization of the distribution of the values in a data series. The Portal generates one box plot for each one hour period for the selected date.

If the chart template does not have a Box Plot series defined, a default counter value, specific to the platform, is chosen.

The “box” portion of a box plot is constructed to enclose one half of the observations in the data series. The lower bound of the box is drawn at the 25th percentile of the distribution, while the upper bound of the box is drawn at the 75th percentile. Whiskers are lines that extend up and down from the box portion to mark the observations at the 10th and 90th percentiles. Outliers beyond the 10th and 90th percentiles are plotted as single points.

Tool tips. Using the mouse, hover over the box portion of any of the box plots displayed on the Box plot chart, to display a tool tip that shows the hour value and the data series distribution values that are plotted, including the hourly mean, median, the 10th, 25th, 75th and 90th percentile values that are plotted.

Click to navigate. Clicking on any of the hourly box plots will automatically zoom the Main Chart time window to a 4-hour time window, centered on the hour selected.

③ CHART TYPE SELECTION.

Use this dropdown menu to change the chart type temporarily. You may select among any of the following chart types: Area, Stacked area, Column, Stacked column, Line and Point. Changing the chart type will trigger regeneration of the chart at the web server immediately.

To change the chart type that is associated permanently with the chart reporting template, you must re-set the chart type using the [Chart Template Editor](#), which is described in the next section of this document.

④ EXPORT CHART DATA.

The chart Export facility generates a comma delimited file in .csv format file containing the time series data that was extracted from the PDB to create the current chart. Click the “Export Chart data” button to navigate to the Export Chart Data panel.

On the Export Chart Data panel, click the “Export to CSV” button. Then, use the web browser to save this file to your local machine. Then you can import the data into applications like Microsoft Excel, for example.

A file name for the export file is generated automatically of the form:

`NTSMFPDB.Extract.MachineName.PerformanceObjectName.dyyyyddd.thhmm.vnnnnnn.csv`

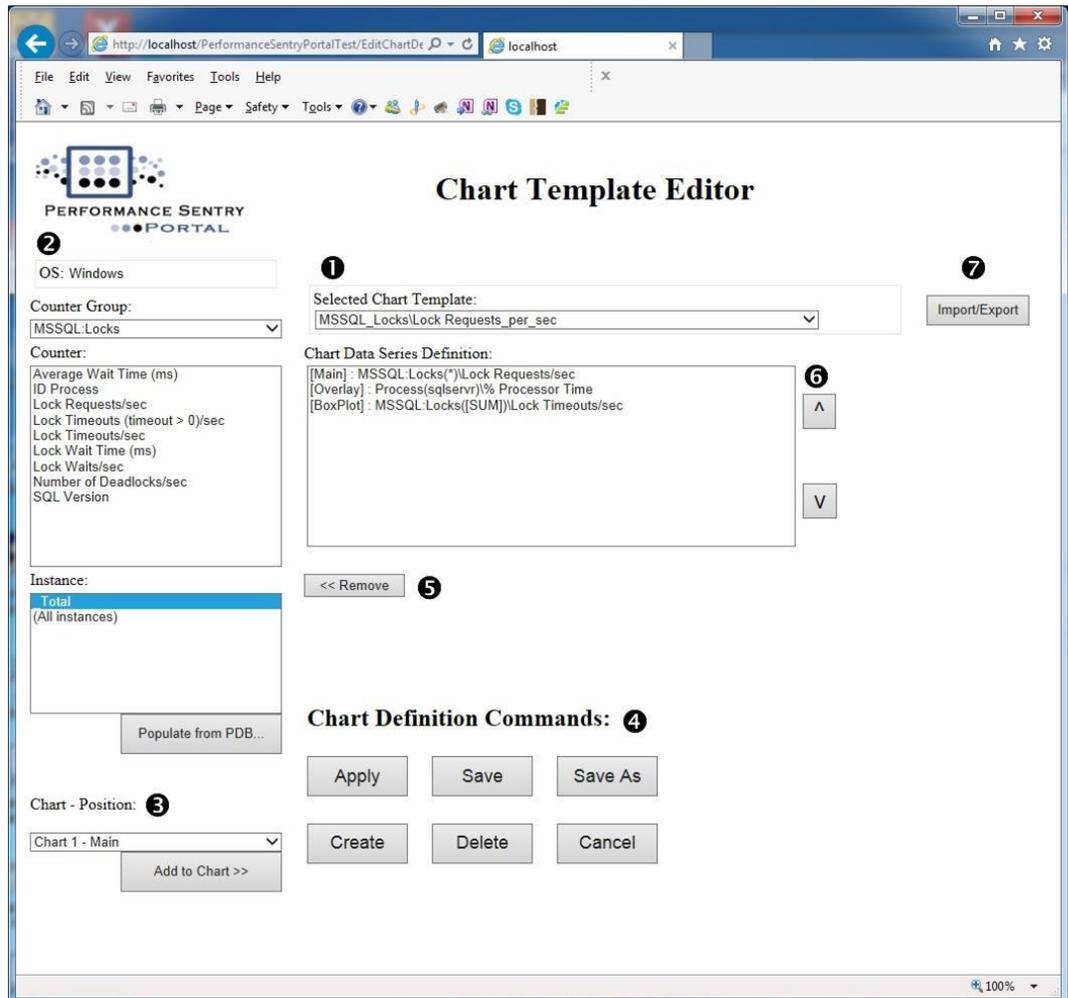
The first row of data in the export file contains the column headings.

The first column of the export file always contains the Timestamp data, which has been generated in the proper Microsoft Office-compatible format. Excel requires that you explicitly format the Timestamp column of data using one of the pre-defined Date or Time formats, using the Format Cells function.

Chart Template Editor

The Daily Performance Charts are built from templates. All of the templates that you installed were built using the Chart Template Editor, which is pictured below. You can modify any of the chart templates that are installed, or define new ones.

You navigate to the Chart Template Editor by clicking the “Configure” button immediately to the right of the chart selection controls on the Daily Performance Charts page, DailyCharts.aspx.



The Chart Definition Editor is implemented using two linked web pages: the initial Chart Definition Editor panel and a second Save Chart Definition panel where you set the chart type and specify the y-axis format and labels. Use the “Save” and “Save As” buttons 4 in the Chart Definition Editor to transfer control to the Save Chart Definition panel.

The initial Chart Definition Editor panel is used to define the data series that will be extracted from the PDB to create the chart. You have an opportunity to define additional chart formatting parameters when you go to “Save” a chart definition.

Chart templates are stored in the PDB across two SQL Server tables, **ChartTemplates** and **ChartSeries**, which you defined [during the installation process](#).

The first panel in the Chart Template Editor is used to define the chart data series that will comprise the Main chart and its associated Box Plot chart. It contains controls for identifying a performance counter and then defining that counter's position on the Main chart or as the value that will be used in the Box Plot. For any counter you want to use as overlays on the Main chart, be sure to set the chart position correctly in the Template Editor when you add the counter to the current list of counter data to extract from the PDB.

① CHART DATA SERIES DEFINITIONS.

When you first navigate to the Chart Definition Editor, the chart series definitions for the active chart template are displayed. Each line in the Chart Data Series definitions list box corresponds to a data series that will be extracted from the PDB when the template is active and added to either the Main Chart or used for the hourly Box Chart.

The chart series definition shows the series type, which can be one of the following:

- a *Main series* that is added to the graph based on the designated *chart type* (stacked area, line, etc.) and plotted against the primary (left hand) y-axis
- an *Overlay series* that is added to the Main chart as a dotted line graph, plotted against the secondary (right hand) y-axis.
- a Box Plot series that is displayed in the Box Plot, a pre-defined hourly report showing distribution values for the performance counter selected.

You can use this panel to define a chart template, or modify an existing one by adding a new chart data series or removing one of the chart data series that is already defined. You can define multiple Main and Overlay data series, but only one Box Plot series can be defined.

- ❖ In general, avoid requesting too many Main and Overlay series definitions for your chart template to ensure that the resulting Daily Performance chart isn't so busy that it is difficult to understand. However, when you want to create a chart template that is primarily to extract counter data in order to export it to a .csv file, then chart appearance may not matter.

The chart data series definition specifies the performance counter that you want to chart. Performance counters are defined using a path name that corresponds to the object name followed by the counter name in the form of a *performance object* \ *counter name*. If the performance object is instanced, you then choose to extract the counter data using one of the following options:

- the sum or total of all data instance counter values in each time interval (denoted by [SUM])
- each individual occurrence of the instanced counter values observed in each time interval (denoted by the "*" wild card value). Note that if you specify that you want to extract each individual instance, the database query that is generated automatically uses the [Top N](#) operator in SQL to return only the top *n* instances of the data series.
- only counter values that correspond to a specific instance name (in which case the specific instance name you select is coded in the series definition as follows:

performance object(instance name) \ counter name

To add a new chart data series to a chart template, you will use the Counter selection controls on the left hand side of the screen to select the counter name for the series you want to add. As the example above illustrates, you can add counters from different performance objects to a chart template and they can all be displayed on the Main Chart or Box Plot.

② COUNTER SELECTION CONTROLS.

In order to define a new data series to add to a chart, you need to access a Counter from one of the Counter sets (also known as performance Objects in Windows). If you don't see the specific counter you are looking for, try changing the Counter set. Only counter sets and counters that are in the current *data collection set* (DCS) are displayed in the Chart Template Editor, based on the machine you selected in the Daily Performance Charts. Not all machines have access to all the Windows, VMware and Linux performance counters that are available in the PDB. If you still don't find the counter you are looking, you can click the "Cancel" and return to the Daily Performance Charts page where you can select a different machine that has a different data collection set associated with it.

Once you find the Counter you would like to add to the chart template, click on the counter name to highlight it in the Counter list box.

If the counter you selected is from a Counter set that potentially can report multiple instances of the counter each measurement interval, the Instance list box is populated with two selections:

- a computed `_Total` instance which is calculated as the sum each interval of all the counter value instances
- (All instances), a designation which produces a [Top N query](#).

Alternatively, you can specify a named instance by clicking the "Populate from PDB..." button and then selecting an instance name that appears in the list.

③ CHART POSITION SELECTION.

Use the Chart Position selection menu to indicate that the series should be added to the Main Chart, based on the designated chart type, added as an overlay on the Main Chart, or is to be displayed in the hourly Box Plot.

- ❖ Only one Box Plot series can be defined. If a Box Plot series is already defined in the template, to change it, you must first remove the previous Box Plot series definition.

Once you have selected a counter, selected an instance designator, and selected its Chart position, press the "Add to Chart >>" button to add the counter data series to the chart template. After pressing the "Add to Chart >>" button, the [Chart Data Series Definition](#) control is updated. The series you just added is shown at the end of the Chart Data Series Definition list box.

After adding a new series to the chart template, you can use the Move Up and Move Down buttons  to rearrange the [Chart Data Series Definition](#) list box to better effect. The order in which data series are arrayed on the actual Main Chart is affected by the order in which the series definitions appear in the Chart Definition Editor's Chart Data Series Definition list box.

Once you have updated the chart template series definitions to your liking, issue one of the available chart definition commands to either

- Apply the changes without updating the Chart template series definitions stored in the PDB or

- Save your changes to the PDB permanently.

④ CHART DEFINITION COMMANDS.

Any changes you have made to the Chart Series definitions are temporary until you save the changed template to the database.

Press one of the Chart Definition Command buttons to issue a command:

Apply. The “Apply” command allows you to view the modifications you just made to the Daily Performance Chart without first saving those changes to the database. The Apply command returns you to the Daily Performance Chart panel where the modified template is used to build a report using the current selected machine and selected date.

After you have had a chance to view the report built using the modified chart template, press the Chart definition “Configure” button on the Daily Performance Chart panel to return to the Chart Definition Editor where the modified template will be loaded. From the Chart Definition Editor, you can then “Save” the modified template to the database or continue to make changes to it.

Save. Use the “Save” command to replace an existing chart template and save the changes you have made to the chart template permanently. The Save command transfers control to the Save Chart Definition panel where you can finish editing the chart template and then commit your changes to the database.

You also use the Save command to gain access to the Save Chart Definition panel, which is where you can set or modify the chart template formatting parameters. These include the options for setting the y-axis labels or setting y-axis maximums.

If you have not made any changes to the chart data series using the Chart Definition Editor panel, you will be prompted to confirm the “Save” operation by pressing the “Save” button again.

Save As. Use the “Save As” the command to save the chart template you have been editing to the database under a new name for the template. This command also transfers control to the Save Chart Definition panel where you will have an opportunity to supply the new chart template name.

Create. The “Create” command creates an empty chart template with no chart data series defined.

The “Create” command clears the contents of the Chart Data Series definitions list box and initializes a new chart template for you to begin modifying in the Editor. Use the Chart Definition Editor to add chart series definitions to the new template and then press the “Apply” button to verify how the new charts look.

After using the Chart Definition Editor to add chart series definitions to the new template, press either the “Save” or “Save As” button to specify the name of the new chart template and set the chart template formatting parameters.

Delete. The Delete command deletes the chart template permanently from the database.

Cancel. Use the Cancel command to terminate the chart template editing session and return to the Daily Performance Charts screen, using the chart template you previously selected.

5 REMOVE SERIES BUTTON.

To remove a chart data series from the current template, highlight the series in the Chart Data Series Definitions list box and then press the “<<< Remove” button. Like any other modification, removing a data series is temporary until you save the change permanently to the database. After removing a series, you can press the “Apply” button to request rebuilding the current chart using the modified template.

6 MOVE UP AND MOVE DOWN BUTTONS.

The order in which data series are processed and added to a chart can be important. In the case of a stacked column or area chart, the first chart series defined is the first one processed and added to the Main Chart. The first data series is positioned at the bottom of a stacked chart, with any other chart series that are defined layered on top of it in the order in which they are defined. It is easier to identify visually the y-axis values for data series positioned at the bottom of a stacked chart.

To change the current series order, highlight the data series you want to re-position in the Chart Data Series Definitions list box and press the “Move Up” or “Move Down” buttons to move the highlighted data series up or down one position at a time.

7 IMPORT/EXPORT CHART DEFINITIONS.

Press the “Import/Export” button to access the Chart Template Utility screen where you can import chart templates from an xml file or export chart templates to xml for transferring between PDBs.

Save Chart Definition

The Save Chart Definition panel is used to specify the name of the chart template, set the chart type, and specify the y-axis format and labels for the chart template. When you are satisfied with all the chart template settings, click the large “Save” button at the upper left. After the chart template definition is saved permanently in the database, the program will return to the Daily Performance Charts page. The chart template you just saved will be the current chart template used by the charting programs.

The screenshot shows a web browser window displaying the 'Save Chart Definition' form. The form has a 'Save' button on the left and a 'Cancel' button next to it. The form fields are as follows:

- Chart Name:** MSSQL_Locks \ Lock Requests_per_sec (marked with a red circled 1)
- Description:** for each named instances of SQL Server: lock requests by entity (Key, Page, Object, Database)
- Chart Type:** Area
- Primary Y Axis:** [Main] : MSSQL.Locks(*)Lock Requests/sec (marked with a red circled 2)
- Axis Label:** per Second
- Max Axis Value:** <Default>
- Axis Scale Type:** Decimal
- Secondary Y Axis:** [Overlay] : Process(sqlservr)% Processor Time (marked with a red circled 3)
- Axis Label:** Utilization
- Max Axis Value:** <Default>
- Axis Scale Type:** Decimal
- BoxPlot definition:** [BoxPlot] : MSSQL.Locks([SUM])Lock Timeouts/sec

1 CHART NAME, DESCRIPTION AND TYPE.

The only requirement for the name of the chart template is that it uniquely identifies the template. The “Save” operation will fail if you specify a duplicate chart name.

Chart Name. This version of the Portal uses a chart template naming convention to group charts into chart sets. By default a default chart name is generated in the form of *objectname/countername* based on the first Main Chart series in the chart definition. The Portal code that loads the Daily Performance Charts regards any text that precedes the “/” separator character as the name of the Chart set and then groups the charts into chart sets.

Chart Type. The chart template defines the initial chart type for the main data series. It allows you to choose among Area, Stacked Area, Column, Stacked column, Line and Point charts. Overlay data series are always plotted as line charts, and the Box Plot data series is visualized as a series of hourly Box and whisker plots.

② PRIMARY Y AXIS SETTINGS.

Time series counter data selected for the Main chart is plotted against the primary y-axis, which is the left hand y-axis. The Primary Y Axis setting section provides controls for setting the y-axis label, the y-axis max, and the type of y-axis scale. It also displays the Main Chart data series specification for reference in a read-only list box.

Primary y-axis label. You can select one of the predefined y-axis label fields or define a custom y-axis. The following y-axis labels are pre-defined:

- per Second
- % Busy
- Bytes/Second
- Megabytes/Second
- Rate (KBps)
- Utilization
- Usage (MHz)
- Seconds
- Milliseconds
- 100ns timer units
- Bytes
- Total

Custom y-axis labels. If you find that none of the pre-defined y-axis labels are appropriate, you can define a custom label. To specify a custom y-axis label, select the “Define Other...” value from the Axis Label dropdown and then type the label text you want to use in the text box immediately to the right of the dropdown menu.

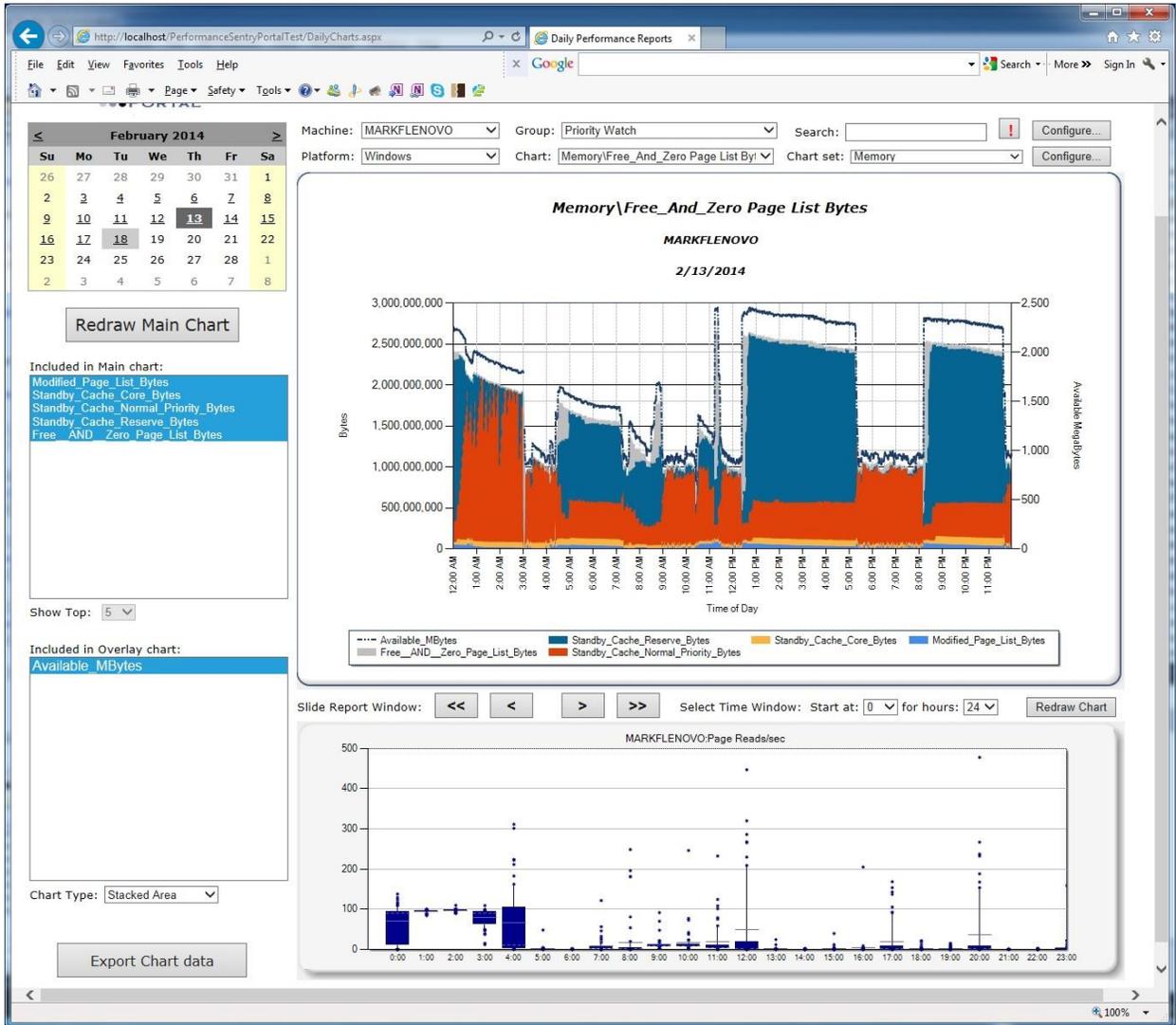
③ SECONDARY Y AXIS SETTINGS.

Time series counter data selected for the Main Chart overlay is plotted against the secondary y-axis, which is the right hand y-axis. The Secondary Y Axis settings section provides controls for setting the y-axis label, the y-axis max, and the type of y-axis scale. The secondary y-axis options are the same as the Primary y-axis controls. The Main Chart overlay data series specification is also displayed for reference in a read-only list box.

Walkthrough: Modifying an existing chart template and saving it as a new chart template

The following example is a walkthrough of the use of the Chart Template Editor for a common scenario where you modify an existing chart template, verify the results, and then save the modified template to the database as a new template.

1. Select the Memory\Free and Zero List Bytes chart.



This is a chart that displays three performance counters that were added to the Memory statistics in recent versions of Windows that breaks out the pool of Available Bytes into a Free and Zero page list and a “standby” list. Free and zeroed pages have been zeroed out by the Memory Manager and are available immediately for new allocations, typically Demand Zero allocations.

The Memory Manager treats the remainder of the Available Bytes pool as a “standby cache” that contains trimmed pages that have been removed from process working sets, but remained resident in memory, and they are unmodified since they were last referenced by a process address space. While these trimmed pages are unmodified, they are eligible to be returned to a process working set quickly in the form of a *transition fault*. Trimmed pages in the standby cache are in various stages of aging.

The three categories of Standby Cache pages are

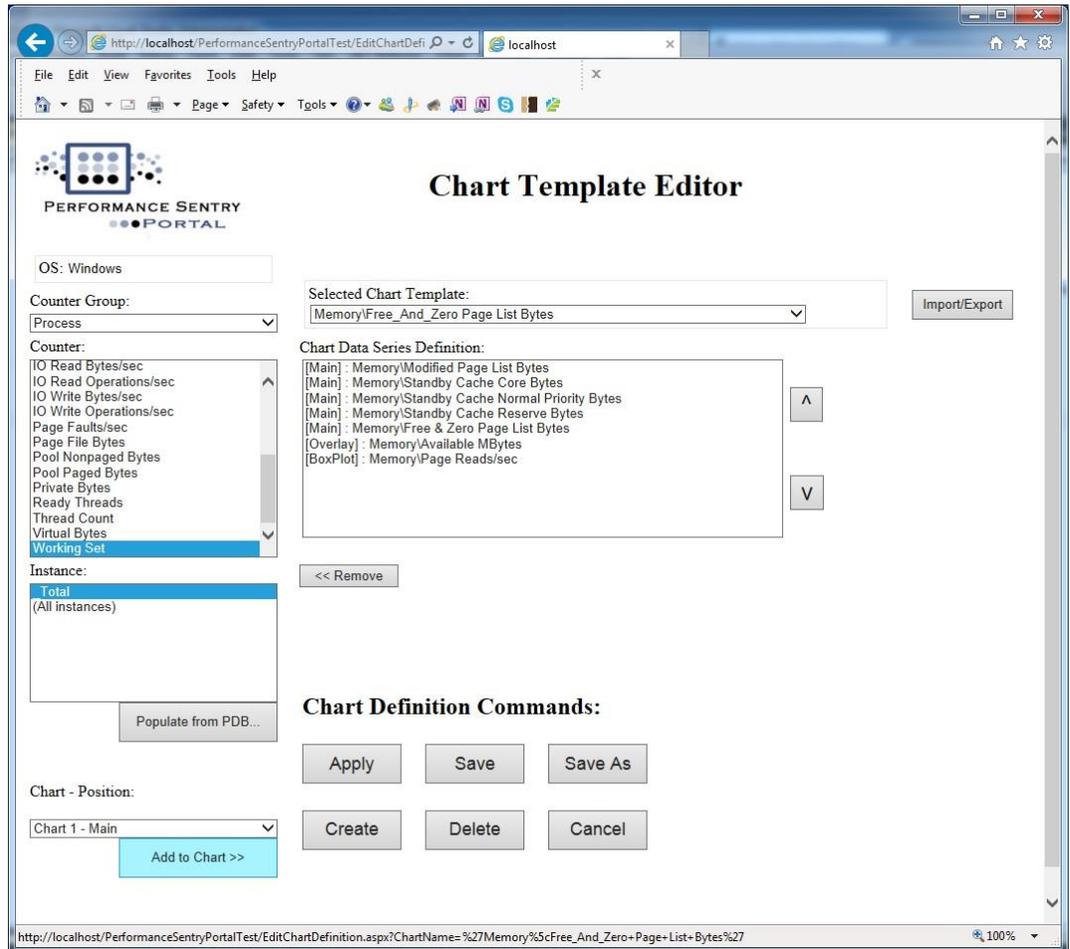
- Standby Cache Core Bytes
- Standby Cache Normal Bytes
- Standby Cache Reserve Bytes

The Standby Cache Reserve counter represents the oldest pages in the cache. They are candidates to be “re-purposed” – shifted to the Free List and their contents zeroed out in anticipation of being assigned to backing a new virtual memory allocation. When the number of Standby Cache Reserve Bytes is relatively large, as a percentage of the total amount of installed physical memory, the system has an ample cushion of physical memory available. The Standby Cache Core Bytes counter reports the number of pages that the Memory Manager is attempting to keep resident in memory – they are flagged as low priority pages for re-purposing to the Free List.

The Modified Page List counts trimmed pages whose contents have changed. They are waiting in a queue for the Modified Page Writer thread in Windows to write a current copy of the page to disk, at which point they transition to the Standby List.

The five Memory counters shown paint a rich picture of the page aging mechanism used by the Windows operating system. They do not say much, however, about how physical memory is actually being used. So, we are going to modify this chart template to add counter data for process working sets, which will show how much physical memory is allocated across active processes.

2. Click the Chart Template “Configure” button and navigate to the Chart Template Editor.



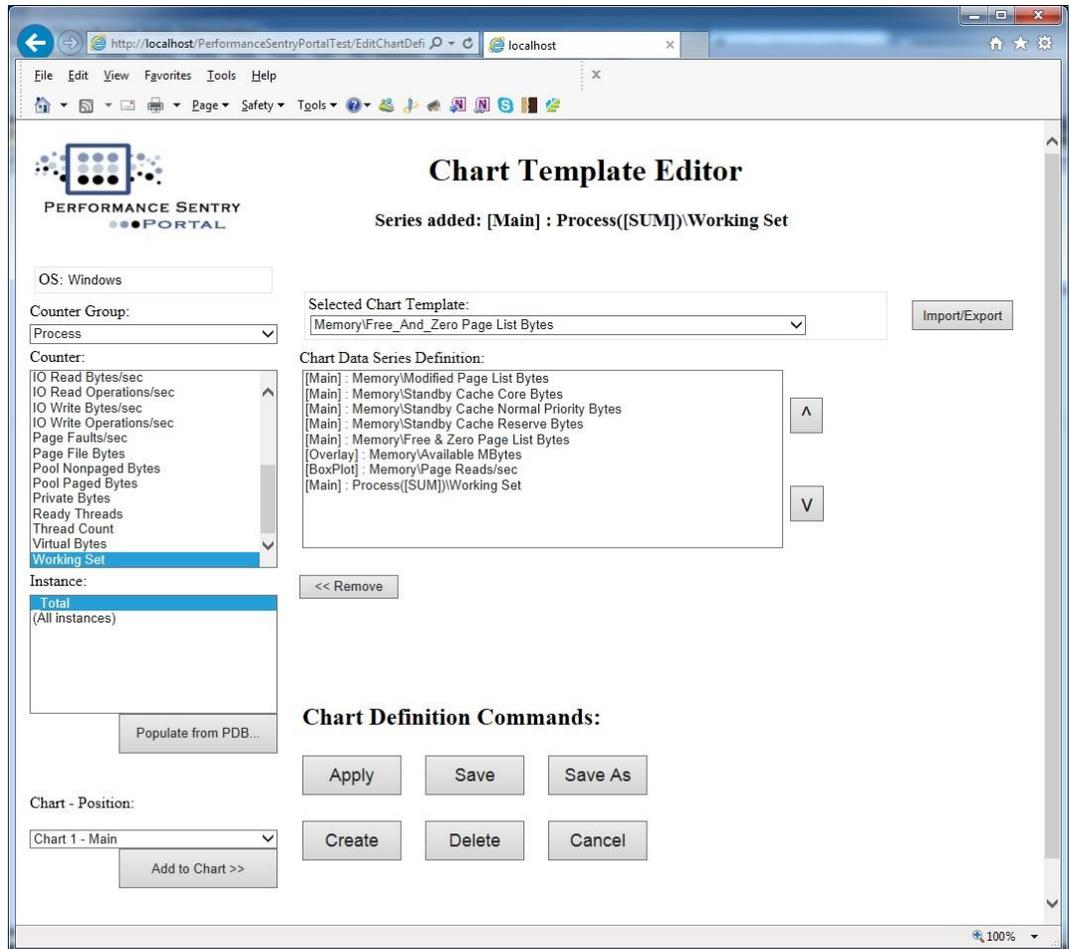
The Chart Template Editor shows the current counter data that is extracted from the PDB to create the memory management chart shown in the previous step.

To add process working set counter data to the chart, access the Counter Group dropdown menu and select the Process object. Then, in the list of process-level counters that are available, scroll down the list and click the Working Set counter to select. Since there are many process instances, in the Instance list box, highlight the `_Total` line, which will create a single data series to be added to the chart, representing the sum of all process instances that were observed each measurement interval for the selected counter.

We want to add this counter to the Main Chart, allowing the working set counter to be stacked on top of the other physical memory allocation counters that are currently specified in the template. Make sure the Chart Position indicator is set to “Chart 1 – Main”, and then click the “Add to Chart >>” button immediately below the Chart Position dropdown.

Quick Start Guide: Web Portal and PDB for Performance Sentry

- Find the Process performance object and click the Working Set counter to highlight it on the page. Select the _Total instance, which will sum all Process instances of this counter for each measurement interval.

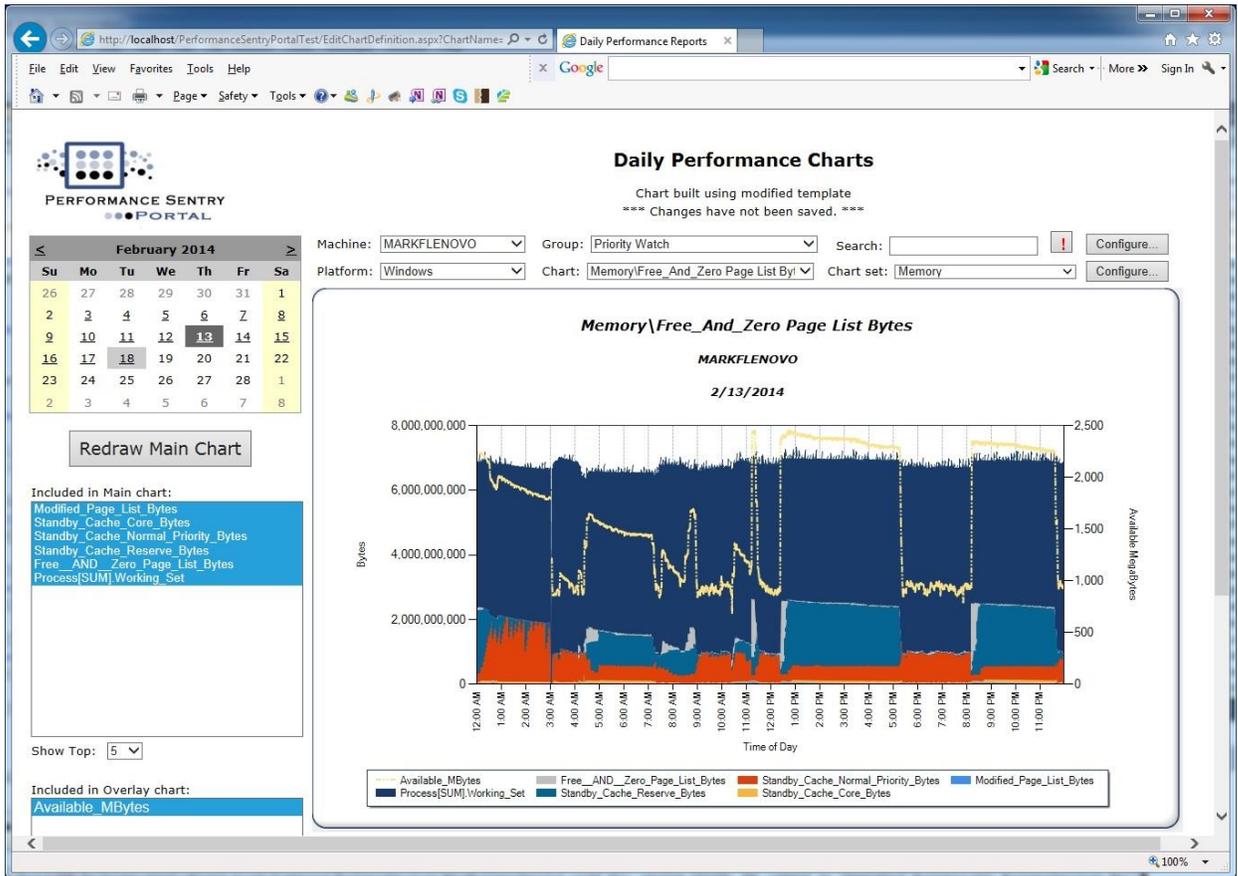


In the list of Chart Data Series Definitions in the middle of the page, verify that a data series described as [Main] : Process([SUM])\Working Set has been added to the template.

If your screen looks different from the one above, click the new entry at the bottom of the Chart Data Series Definitions list, press the “<< Remove” button, and try the Add Counter operation again.

Click the “Apply” command button to be able to preview a chart based on the modified template prior to permanently updating the chart template definition stored in the PDB.

4. Add the Process[SUM]\Working Set counter to the Main Chart and press the “Apply” command button.



The modified template stacks the Process([SUM])\Working Set counter data on top of the previous Memory counters that break out the amount of physical memory available for allocation based on the three lists of available memory pages the OS maintains: the Free list, the Modified list, and the Standby Cache.

As indicated on the screen, the Main Chart was produced from a modified chart template, and those changes have not been applied to the chart template definitions stored permanently in the PDB. To save this modified template, press the “Configure” button immediately to the right of the Chart template and Chart set dropdown menus to return to the Chart Template Editor.

Quick Start Guide: Web Portal and PDB for Performance Sentry

5. Click the Chart Template “Configure” button again, press the “Save As” button on the Chart Template screen to save the modified template as a new template with a new name. When you arrive at the Save Chart Definition page, type in a new name for the Chart and press the “Save” button to add a new chart template to the database.

The screenshot shows a web browser window with the URL `http://localhost/PerformanceSentryPortalTest/`. The page title is "Save Chart Definition" and features the Performance Sentry Portal logo. The form contains the following fields and controls:

- Chart Name:** A text input field containing "Memory" and a dropdown menu showing "Physical Memory Usage summary report".
- Description:** A large text area for entering a description.
- Chart Type:** A dropdown menu set to "Stacked Area".
- Primary Y Axis:** A list of data sources including "[Main] : Memory\Modified Page List Bytes", "[Main] : Memory\Standby Cache Core Bytes", "[Main] : Memory\Standby Cache Normal Priority Bytes", "[Main] : Memory\Standby Cache Reserve Bytes", "[Main] : Memory\Free & Zero Page List Bytes", and "[Main] : Process([SUM])\Working Set".
- Axis Label:** A dropdown menu set to "Bytes" and an "Other Label" text input field.
- Max Axis Value:** A dropdown menu set to "<Default>".
- Axis Scale Type:** A dropdown menu set to "Decimal".
- Secondary Y Axis:** A text input field containing "[Overlay] : Memory\Available MBytes".
- Axis Label:** A dropdown menu set to "Available Megaf" and an "Other Label" text input field containing "Available MegaBytes".
- Max Axis Value:** A dropdown menu set to "<Default>".
- Axis Scale Type:** A dropdown menu set to "Decimal".
- BoxPlot definition:** A text input field containing "[BoxPlot] : Memory\Page Reads/sec".

On the left side of the form, there are two buttons: "Save" and "Cancel".

Server Health Reports

Click on ‘Server Health’ in the Performance Sentry Portal’s home screen to navigate to the Server Health Report page. Server Health Report displays key metrics in a report format showing statistics such as Min, Max, Ave., Median, 95th, 98th, and 99th percentiles. For instanced objects, such as LogicalDisk, Process, and Processor, you can display per-instance metrics, or summarized [SUM] metrics. Reports are generated by specifications in report templates.

At installation, default report templates were imported for basic Server, SQL Server, Web Server, and VMware Server. You can modify these report templates or create custom report templates to meet your specific needs.

Reports are limited to reporting on one machine at a time, and the report time window is limited to no more than one 24-hour day. You can dynamically narrow the report window to a period as little as one hour through the controls on the page.

Reports are built by querying the performance database to gather a specific set of counter data fields. The PDB queries are generated dynamically, based on specifications contained in a *report template*, which indicates the PDB data fields to extract.

A sample Server Health Report is pictured below:

Server Health Report
Machine: SQLSERVER
Report template: ServerHealth
Report Date 8/28/2014; Start Hour: 0; Duration: 24

Select Report: ServerHealth (5) Rebuild Configure... (6)
 Server-side processing complete! (1)

Object	Counter	Instance	Min	Ave.	Median	Max	95%	98%	99%	n
System	% Total Processor Time		0.0	26.475	21.910	82.840	55.990	65.980	71.210	1316
System	Processor Queue Length		0.0	0.897	0.0	8.000	5.000	5.000	6.000	1316
Process	Ready Threads	sqlservr (2668)	0.0	0.061	0.0	4.000	0.0	1.000	2.000	1316
Process	Ready Threads	kntrnsm (1260)	0.0	0.035	0.0	1.000	0.0	1.000	1.000	255
Process	Ready Threads	twgescli (3168)	0.0	0.006631	0.0	1.000	0.0	0.0	0.0	754
Memory	Available Bytes		326451200	308201204	309587968	229904384	276283392	273162240	268693504	1316
LogicalDisk	% Free Space	D:	0.470	0.470	0.470	0.470	0.470	0.470	0.470	1316
Memory	Pages Input/sec		0.130	4.890	0.150	1316.440	0.680	15.370	61.100	1316
Processor	% Processor Time	[SUM]	5.350	211.812	175.320	662.700	447.890	527.870	569.680	1316
PhysicalDisk	% Disk Busy	18 F:	0.027	24.434	20.860	94.930	60.250	77.580	83.450	670
PhysicalDisk	% Disk Busy	10 F:	0.048	21.691	19.130	78.320	53.560	65.250	70.410	699

Select Report Date: (2)
 August 2014 calendar view showing 28th selected.

Select Time Window: (3)
 Start at: 0 for hours: 24

The descriptions of the elements that comprise the Server Health Report page are keyed to the numbers that overlay the example screen shot above. Descriptions start on the next page.

❶ REPORT.

The Report is displayed in the center portion of the web page. It is built on demand based on

- the counter data fields specified in the report template
- for the machine selected, and
- the date and time range selected.

The report title contains the machine name, the report template name, the date and the time range. Daily Server Health Reports that span more than one calendar day are not supported in this release.

❷ DATE SELECTION CALENDAR CONTROL.

Clicking on a Day field in the calendar control allows you to change the report date.

Date selections in the calendar control are enabled for selection based on start timestamp and last timestamp fields that are maintained in the PDB's Machines table. The Machines table contains one row for every machine for which performance counter data is available.

After selecting a new report date you will have to click the **Rebuild** button to generate the report for the selected date.

❸ TIME WINDOW SELECTION CONTROLS.

By default, the health report covers 24 hours of the selected date. The Time Window controls allow you to narrow the report window to intervals as small as one hour. First, change the report start hour and then change the duration hours. After setting a new start hour or changing the duration, click the **Rebuild** button to generate the new report.

❹ MACHINE SELECTION CONTROLS.

The machine selection controls allow you to select a new machine to report using the current report template. The machine selection controls consist of

- a Machines dropdown list populated from the current Machine Group,
- a Machine Group selection menu, and
- a Search input field that supports auto-completion, based on matching against the first three characters that you type into the Search box.

There is also a Server icon button, under the search text input field. To switch to the machine that is named in the Search input text box, click the server icon button, then select a date from the calendar control and click the **Rebuild** button.

❺ REPORT TEMPLATE SELECTION CONTROLS.

Report templates are used to specify the counter data to be queried in the PDB to build the Server Health Report.

Report template definitions are stored within the PDB in two tables, **ReportSeries** and **ReportTemplates**.

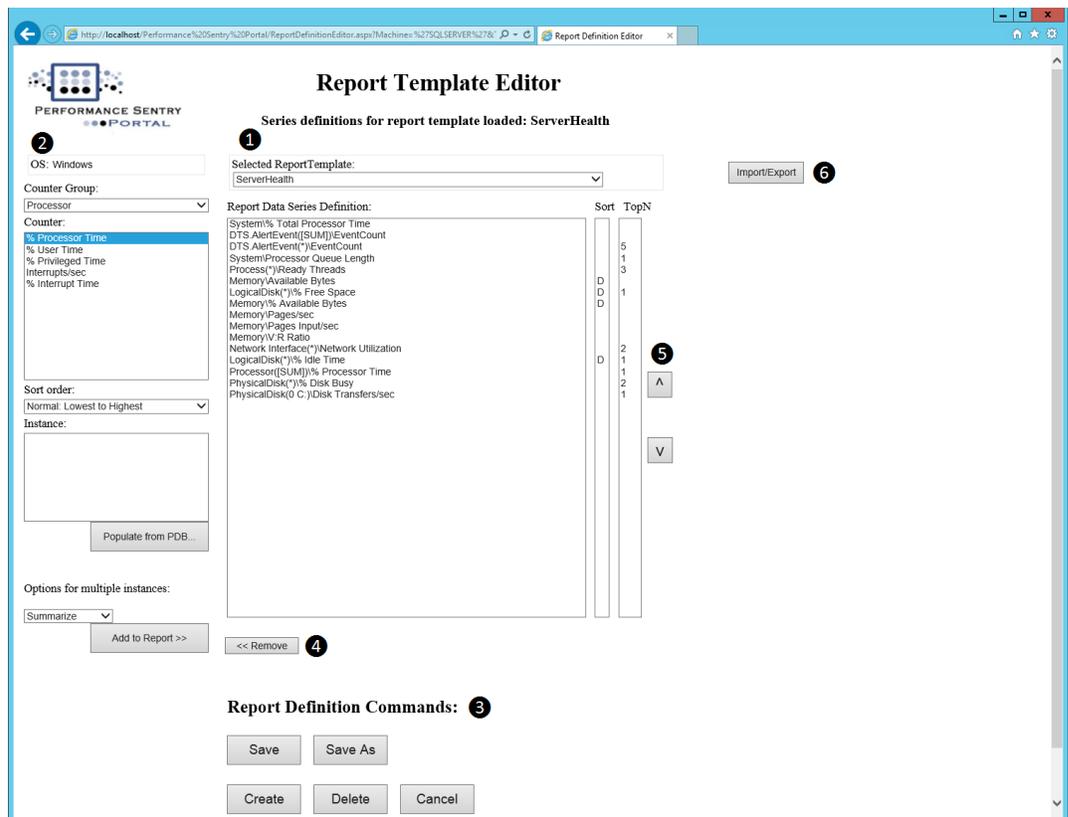
To modify the report template or create a new report template from scratch, click the **Configure** button ❹ to the right of the **Select Report** dropdown box. For more information on report templates, see the section in this document that discusses the "[Server Health Report Template Editor](#)."

6 CONFIGURE SERVER HEALTH REPORT

To modify the report template or create a new report template from scratch, click the **Configure** button to the right of the **Select Report** dropdown box **5**. For more information on editing report templates, see the section below.

Server Health Report Template Editor

The Server Health Reports are built from templates. Each of the templates that are installed initially was built using the Report Template Editor, which is pictured below. You can modify any of the report templates that are installed, or define new ones.



The Report Template Editor is used to define the report data that will comprise the report. It contains controls for specifying a performance counter and then specifying that counter's position in the report.

Report templates are stored in the PDB across two tables, **ReportSeries** and **ReportTemplates**, which you defined [during the installation process](#).

The descriptions of the elements that comprise the Report Template Editor page are keyed to the numbers that overlay the example screen shot above. Descriptions start on the next page.

❶ REPORT DATA SERIES DEFINITIONS.

When you first navigate to the Report Template Editor, the report definition for the active report template is displayed. Each line in the **Report Data Series Definitions** list box corresponds to data that will be extracted from the PDB and dynamically added to the Server Health Report.

You can use this panel to define a new report template, or modify an existing one by adding a new report data series or removing one of the existing report data series definitions.

The report data series definition specifies the performance counter that you want to report. Performance counters are defined using a name that corresponds to the object name followed by the counter name in the form of a *performance object\counter name*. If the performance object is instanced, you then choose to extract the counter data using one of the following options:

- the sum or total of all data instance counter values (denoted by [SUM])
- report top N individual occurrences of the instanced counter values (denoted by the "*" wild card value in the report data series definition).
- only counter values that correspond to a specific instance name (in which case the specific instance name you select is coded in the series definition as follows:

performance object(instance name)\counter name

To add a new report data series to a report template, you will use the counter selection controls on the left hand side of the screen to select the counter name for the series you want to add. As the **Report Data Series Definition** list box in the screen capture above illustrates, you can add counters from different performance objects to a report template.

❷ COUNTER SELECTION CONTROLS.

In order to define a new data series to add to a report, you need to access a counter from within the **Counter Group** (also known as performance Object in Windows). If you don't see the specific counter you are looking for in the **Counter** list, try changing the selection in the dropdown box under **Counter Group**. Only counter groups and counters that are in the current *Data Collection Set* (DCS) are displayed in the Report Template Editor, based on the machine you selected in the Server Health Report page. Not all machines have access to all the Windows, VMware and Linux performance counters that are available in the PDB. If you still don't find the counter you are looking for, you can click the **Cancel** button and return to the Server Health Report page where you can then try to select a machine that has a different DCS associated with it.

Once you find the counter you would like to add to the chart template, click on the counter name to highlight it in the **Counter** list box.

Sort order is relevant for counters with multiple instances (like LogicalDisk, Physical Disk, and Process). When the Performance Database (PDB) is queried for the specified counter, counter data values are returned in ascending order, lowest to highest. When 'Report Top N' is chosen under **Options for multiple instances** then the "Top N" values will be displayed in the report. This is a good option for reporting, for example, the top n processes with the highest amount of '% Processor Time' usage or Working set bytes. However, this is not good for reporting on counters like '% Free Space' for logical disk. In this case you want to report those disks with the *least* amount of % Free Space, so 'Reversed: Highest to Lowest' needs to be selected under **Sort order**, and 'Report bottom N' selected under **Options for multiple instances**.

Choosing 'Summarize' (the default) under **Options for multiple instances** will summarize all the values for multiple instances of the selected counter.

Once you have selected your counter and chosen the appropriate reporting options, click on the **Add to Report** button to add it to the **Report Data Series Definition** list box.

If the counter you selected can report multiple instances then by default, the sum [SUM] will be displayed in the Report Data Series Definition list and subsequently in the Server Health Report when the report is built.

Alternatively, you can report a named instance by choosing 'Select By Name' under **Options for multiple instances** and clicking the **Populate from PDB...** button and then selecting an instance name that appears in the list.

Click the **Add to Report** button to add the counter with the selected instance to the **Report Data Series Definition** list.

③ REPORT DEFINITION COMMANDS.

Any changes you have made to the Report Data Series Definitions are temporary until you save the changed template to the database.

Press one of the Chart Definition Command buttons to invoke an action:

Save. Use the "Save" command to permanently save the changes you have made in an existing report template in the database. The "Save" command replaces any existing Report Data Series Definitions with the current set of Report Data Series Definitions that are displayed in the Report template Editor.

Save As. Use the "Save As" the command to save the report template you have been editing to the database under a different name. A text box will appear immediately to the right of the "Save As" button where you can type the new report template name. After entering the new name in the text box, click **Save As**.

Create. The "Create" command creates an empty chart template with no report data series defined.

The "Create" command clears the contents of the Report Data Series definitions list box and initializes a new chart template for you to begin modifying in the Editor. Continue using the Report Definition Editor to add report series definitions to the new template.

After using the Report Definition Editor to add report series definitions to the new template, press either the "Save" or "Save As" button to specify the name of the new report template.

Delete. The Delete command deletes the report template permanently from the database.

Cancel. Use the Cancel command to terminate the report template editing session and return to the Server Health Report screen, which uses the report template you previously selected.

④ REMOVE SERIES BUTTON.

To remove a report data series from the current template, highlight the series in the **Report Data Series Definitions** list box then press the <<< **Remove** button. Like any other modification, removing a data series is temporary until you save the change permanently to the database.

⑤ MOVE UP AND MOVE DOWN BUTTONS.

Quick Start Guide: Web Portal and PDB for Performance Sentry

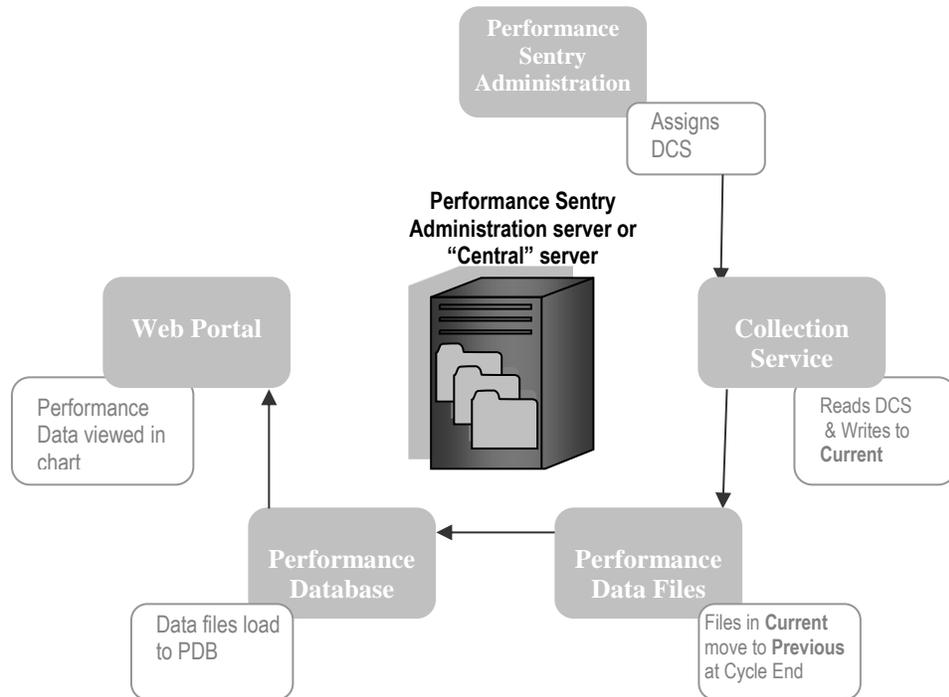
To change the current series order, highlight the data series you want to re-position in the **Report Data Series Definitions** list box and press the **Move Up** or **Move Down** buttons to move the highlighted data series up or down one position at a time.

⑥ **IMPORT/EXPORT CHART DEFINITIONS.**

Press the **Import/Export** button to access the **Report Template Utility** screen where you can import report templates from an xml file or export report templates to xml for transferring between PDBs. You can also use the Report Template Utility screen to select the default **Server Health Report**.

How All of This Works

Following is a description of how the core components of Performance Sentry interact.



The performance data viewed in the Web Portal charts is collected by the Collection Service running on each server. The Collection Service collects Windows performance counters as defined by the Data Collection Set (DCS). A Data Collection Set is a set of performance counters along with the parameters of how to collect the data for them and the interval on which to collect the data (by default every minute). Data is collected to a file with a name in the form *computername.yyyymmddhhmm.smf*. The Collection Service collects the performance data for those objects and counters defined in the DCS and applies any filters as the data is collected. The Collection Service cycles once per day, at which time the data collection file is closed and another one opened. Default DCS parameters are applied during installation, but this DCS may be modified or replaced at a later time using the Sentry Administration component.

Performance Sentry Administration is used to define the DCS by listing the objects and counters, the collection filters that limit the amount of data collected, and the parameters of collection such as frequency of collection and what to do with the data files containing the collected data at the end of the Collection Cycle. Performance Sentry Administration controls the data Collection Service by assigning a DCS to those machines you want data collected on, and is used to issue the command to Suspend, Resume, or Modify data collection.

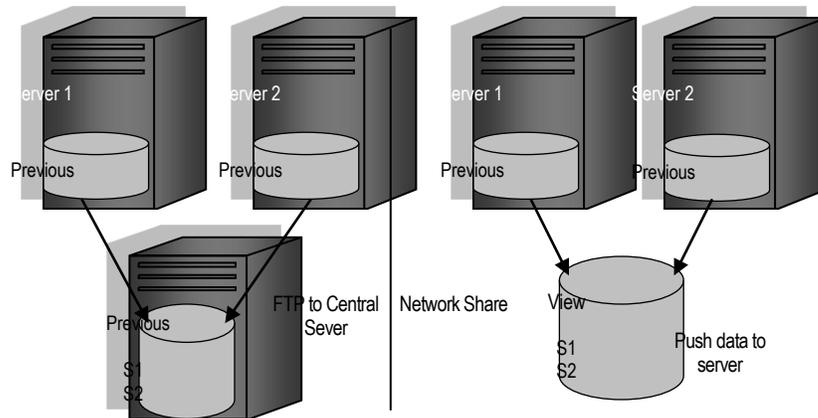
During the collection cycle of 24 hours, data is collected once every minute and stored locally on the server in a hierarchical folder structure consisting of a Current, Previous and Archive folder. At the set time of Cycle End, the current performance data collection file is closed and moved from the Current folder to the Previous folder where files will reside for one day and then move, during the next Cycle End, to the Archive folder. Data files in the Archive folder are saved for 7 days by default.

Quick Start Guide: Web Portal and PDB for Performance Sentry

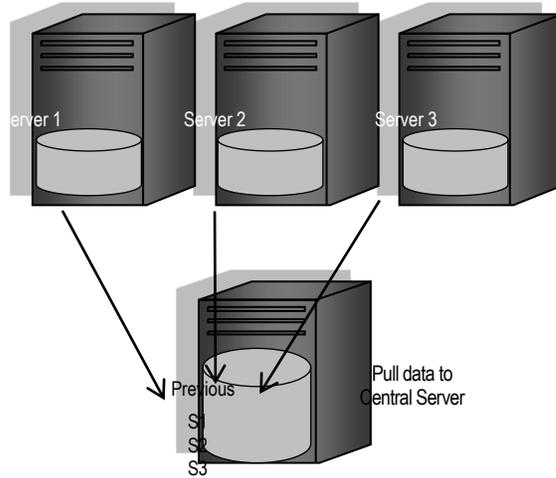
Once performance data file migration has completed, an optional command, batch script, or program can be launched to load the data to the performance database. The Cycle End Command facility, as it is called, is fully documented in the Performance Sentry User Manual.

The Cycle End command can be used to launch data loading in one of three methods:

- Launch the data loading program NTDACMD.exe to connect to the database and load the data directly. This requires that the program is launched under an account that has sufficient user rights to write to the SQL Server database. Refer to the Performance Sentry User Manual for instructions on how to configure the Collection Service to launch a program using impersonation.
- Push – at Cycle End, scripts execute to transfer data to a single central server folder, either using the FTP method, or the network share method. Once the files from all servers have been transferred, the NTDACMD.exe program is scheduled on the server to load the data in all the files to the database.



- Pull – at a designated point in time, data file(s) are transferred from remote servers using routines executed on the central server. This can either be scheduled where the central server routinely transfers data files to a central location or ad-hoc (manually), where data transfers only when problems arise. Once the scheduled 'pull' file transfer has completed, the NTDACMD.exe program is launched to load the files from the single folder to the SQL Server database.



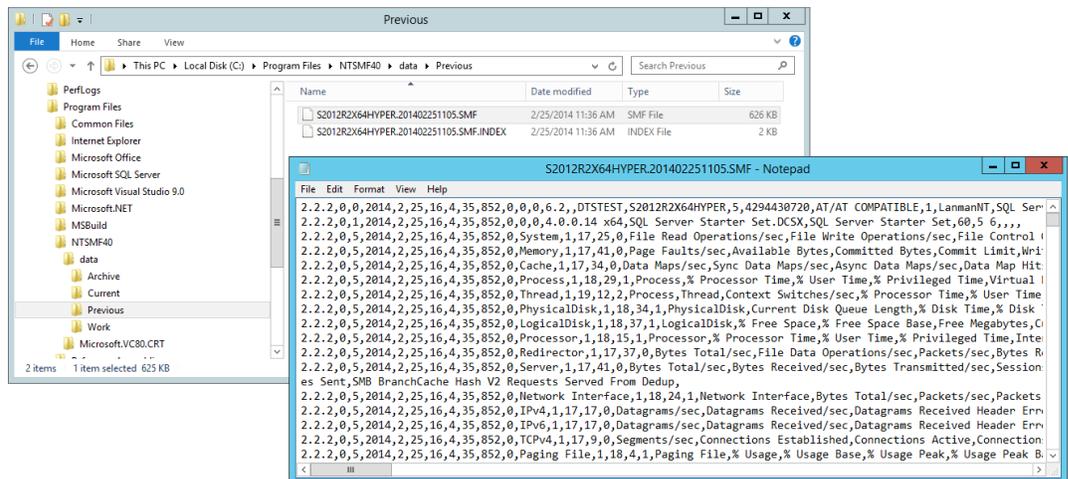
Performance Sentry Performance Database (PDB)

The Performance Sentry Performance Database (PDB) is a Microsoft SQLServer-based database that contains tables in the form of *machinename@objectname* with rows indexed by datetime and columns that index performance counters. Performance data from each monitored machine is loaded into the database using a program which reads data from the Collection Service files and loads the respective tables. Data in the database is saved by default for 15 days. Data can be retained for a longer period of time by changing a parameter. In the event that only specific performance data is needed for longer periods of time, for example to do capacity planning, then data from the Performance Sentry database can be exported to another database which is maintained by the customer.

Through the Performance Sentry Web Portal, the performance data from the PDB is displayed in chart format. Charts are customizable as requirements dictate.

How the Performance Data is Stored in the Database

The ASCII, comma delimited data files, located in the Previous folder are loaded to the Performance Database (PDB). The performance data represented in the database is roughly twice the size of the data represented in the files produced by the Collection Service.



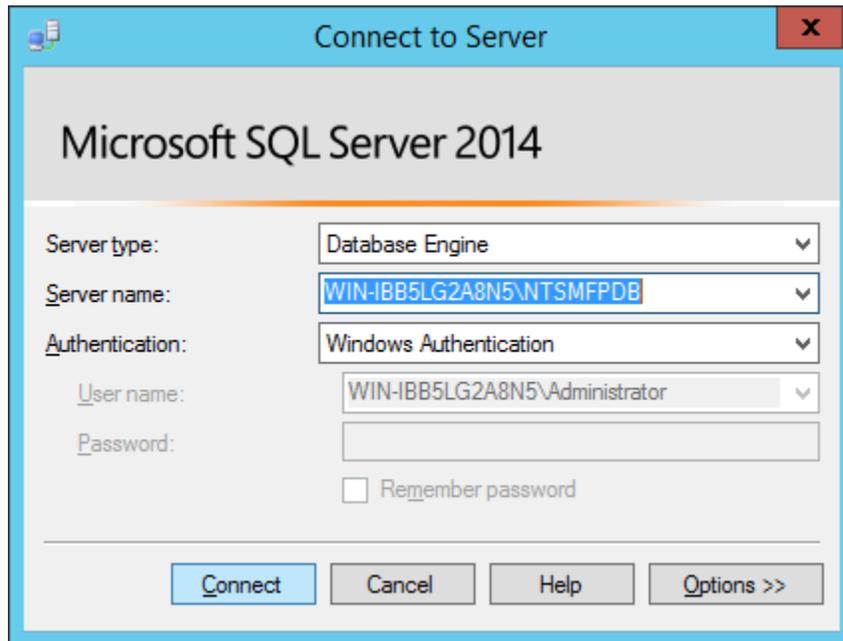
Quick Start Guide: Web Portal and PDB for Performance Sentry

Note: The Performance Sentry Portal can only access and display data for those machines loaded into the Performance Sentry database.

View Data Tables

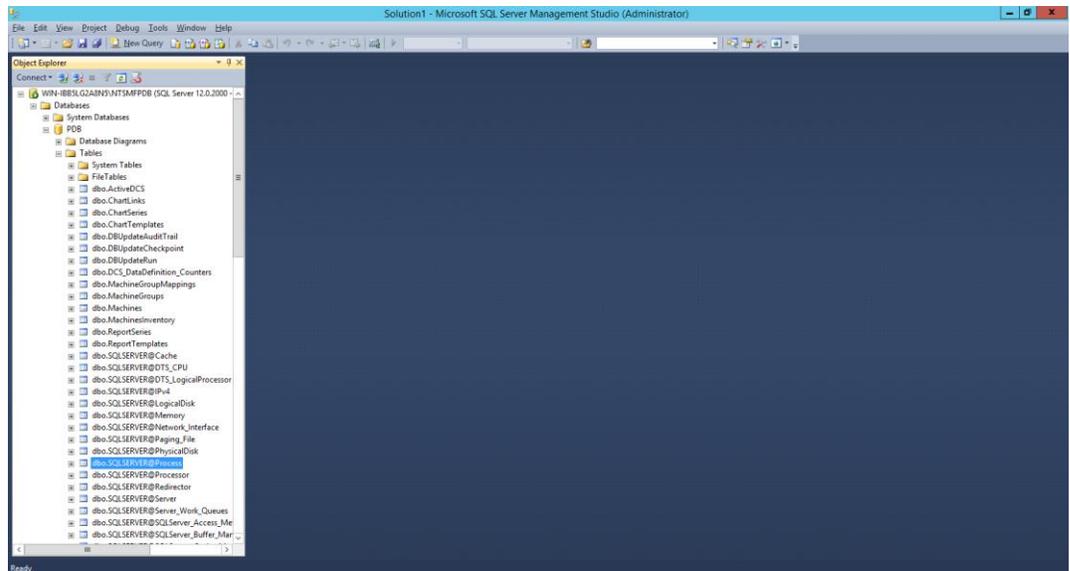
To view how the data is stored in the database, follow the steps below. You may want to know how the data is structured in order to access it from another SQL reporting tool like Microsoft Excel, Crystal Reports, or Microsoft SQL Server Reporting Services.

1. Click **Start/ All Programs/ Microsoft SQL Server 2014/ SQL Server Management Studio Express**, and verify the Server Name: is the machine you installed to during the product setup.



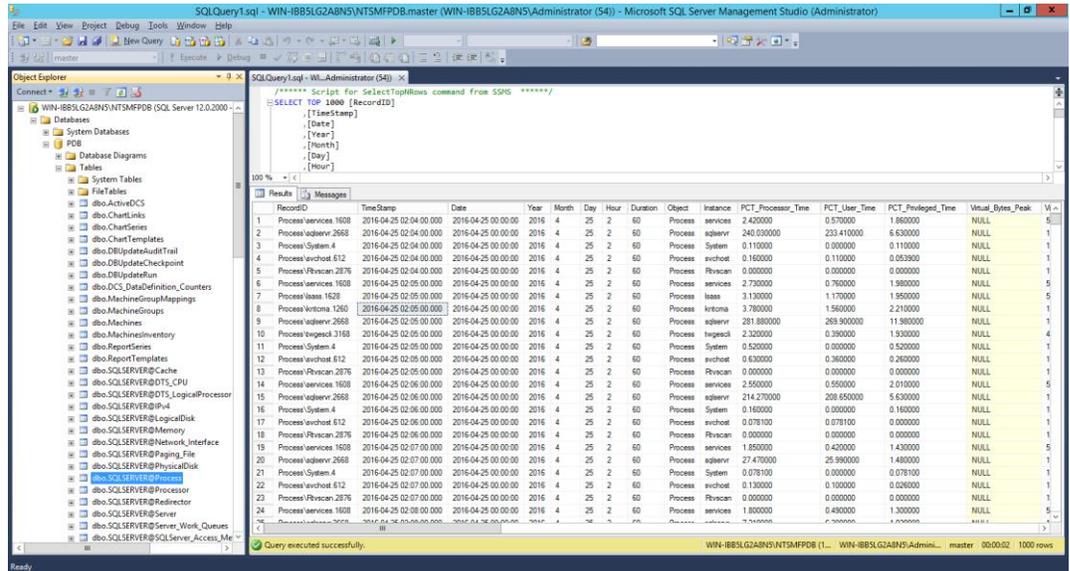
2. Click **Connect**.

3. Expand the folder tree to table level and expand the Tables folder, as displayed here.



4. Right click on the table **dbo.SQLSERVER@Process** and click on **Select Top 1000 Rows** from the popup menu.

The table displays.



The columns in the table represent the performance counters in the Chart Data Series Definition list in the Web Portal.

How to Transition from Evaluation to Production

Now that you've evaluated Performance Sentry and have seen the wealth of Windows Performance Data that can be collected, stored, and reported, you will want to configure Performance Sentry to meet your enterprise performance reporting needs and roll out data collection to other servers. For a complete overview and reference of Performance Sentry, please see the document **Performance Sentry User Manual** in the folder named:

C:\Program Files (x86)\Performance Sentry Administration v4\documents

Remember, if you need additional information or have any questions regarding Performance Sentry, consult Demand Technology Software's website at www.demandtech.com, or send email to support@demandtech.com.

Thank you for your interest in Demand Technology Software's Performance Sentry Portal and PDB!