Microsoft System Center Operations Manager 2007

Links, References, Tools and Blogs

Table of Contents

Summary

This document will provide guidance to online links, resources, tools and blogs to assist in the management of Operations Manager 2007

Documentation Links

- System Center Operations Manager 2007 Product Documentation
- Release Notes for Operations Manager Service Pack 1

  These release notes address late-breaking issues and information about this release of Operations Manager 2007 SP1.
- Operations Manager 2007 Quick Start Guide

  Provides information to help you get up and running with Operations Manager 2007 SP1 as quickly as possible.
- Operations Manager 2007 SP1 Supported Configurations

  Information about supported operating systems, hardware configurations, software requirements, installation combinations, and security configurations for Operations Manager 2007 SP1.
- Operations Manager 2007 Service Pack 1 Upgrade Guide

  Provides information to help existing customers of Operations Manager 2007 upgrade to Service Pack 1.
- Operations Manager 2007 Technical Concepts Guide

  This guide provides an in-depth description of Operations Manager 2007 services and components and shows how they work together to provide the features in Operations Manager 2007, as well as deep technical insights into the internal architecture of Operations Manager 2007 and the Operations Manager 2007 Management Packs so that you can design, deploy, and manage solutions better.
- Operations Manager 2007 Design Guide

  The Operations Manager 2007 Design Guide takes you through the steps necessary to develop a complete architectural plan for your Operations Manager 2007 implementation.
- Operations Manager 2007 Deployment Guide

  This guide steps you through the deployment process for System Center Operations Manager 2007.
- Operations Manager 2007 Key Concepts Guide

This guide describes the key concepts required to understand Operations Manager. It describes modeling, how modeling is implemented in Operations Manager, and key changes between Operations Manager 2005 and 2007.

- Operations Manager 2007 Migration Guide

  This guide helps you understand the migration process from Microsoft Operations Manager 2005 (MOM 2005) to Operations Manager 2007 and the tools that are used in the process.

- Operations Manager 2007 Performance and Scalability White Paper

  The official tool for capacity and sizing guidance for Operations Manager 2007 is System Center Capacity Planner 2007. You can get Capacity Planner 2007 at http://www.microsoft.com/sccp. You can also use this document to gain an understanding of scalability best practices and guidelines before you plan a Microsoft System Center Operations Manager 2007 deployment.

- Operations Manager 2007 Security Guide

  This guide provides you with security-related information as it pertains to Operations Manager 2007.

- Authoring Guide

  Operations Manager 2007 Management Pack Authoring Guide: This document provides detailed information about how to create a Management Pack for a product. The product can be an application, a service, or a device.

- Report Authoring Guide

  Provides scenarios for custom report creation and reference information on report parameters.

  - Report Authoring Guide Sample Report

- Best Practices for Targeting Rules and Monitors in Operations Manager 2007

  The poster illustrates some of the most common scenarios and provides best practices on how to properly target rules and monitors in System Center Operations Manager 2007.

- Operations Manager 2007 Operations Guide

  This guide is a comprehensive resource that can be used to understand and use your Operations Manager 2007 implementation to your best advantage. It includes procedures for backup and restore, and for restructuring your Operations Manager implementation.

- System Center Operations Manager 2007 SDK

  This software development kit (SDK) documentation includes descriptions and examples that show how to automate and extend Operations Manager features.

- Management Pack Guides for Windows Operating Systems and Technologies

The guides for Operations Manager management packs for Windows operating systems and technologies are installed with the management pack and can be viewed online.

- Management Pack Guides for Server Products

  The guides for Operations Manager management packs for server products are installed with the management pack and can be viewed online.

System Center Operations Manager TechCenter

TechCenter home page for Microsoft System Center Operations Manager 2007

Technet Videos: System Center Operations Manager

System Center Operations Manager 2007 Service Pack 1 Training Videos

TechNet Virtual Labs: System Center

- TechNet Virtual Lab: Building Management Packs with the Authoring Console
- TechNet Virtual Lab: Extending Operations Manager 2007 to Monitoring SAP on SQL
- TechNet Virtual Lab: System Center Operations Manager 2007- Advanced Topics
- TechNet Virtual Lab: System Center Operations Manager 2007- Introduction
- TechNet Virtual Lab: Understanding the Operations Manager 2007 SDK
- TechNet Virtual Lab: Using Engyro's Connectors to interoperate with Tivoli TEC and HP OpenView Operations

TechNet Webcasts: System Center Operations Manager

Below are the currently available Webcasts, check the link above for added webcasts

- TechNet Webcast: Assure the Availability and Performance of Your SharePoint Environments (Level 300)
- TechNet Webcast: Client Monitoring with System Center Operations Manager 2007 (Level 300)
- TechNet Webcast: Client Monitoring with System Center Operations Manager 2007 (Level 200)

- TechNet Webcast: Client Monitoring with System Center Operations Manager 2007 (Level 100)
- TechNet Webcast: Client Monitoring with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: End-to-End Service Monitoring with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: End-to-End Service Monitoring with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: Installation and Management Pack Migration of Operations Manager 2007 (Level 200)
- NEW TechNet Webcast: Introducing Operations Manager 2007 R2 (Level 300)
- TechNet Webcast: Monitoring with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: Monitoring with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: Monitoring with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: Reporting with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: Reporting with System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: Security and Enterprise Features of System Center Operations Manager 2007 (Level 200)
- TechNet Webcast: SQL Server 2008: New Performance Monitoring and Troubleshooting Using Management Studio (Level 300)
- NEW TechNet Webcast: Successfully Monitor UNIX and Linux Alongside Your Windows Infrastructure with Operations Manager 2007 R2 (Level 300)
- TechNet Webcast: System Center Operations Manager 2007 Installation and Management Pack Migration (Level 200)
- TechNet Webcast: System Center Operations Manager 2007 Management Pack Life-Cycle Management (Level 300)
- TechNet Webcast: System Center Operations Manager 2007 Technical Overview (Level 200)
- TechNet Webcast: System Center Operations Manager 2007 Technical Overview (Level 200)
- TechNet Webcast: System Center Operations Manager 2007 Technical Overview (Level 200)
- TechNet Webcast: System Center Operations Manager 2007: Install and MP Migration (Level 200)

Software Links

- System Center Pack Catalog
- Evaluate Microsoft System Center Operations Manager 2007 SP1 (180 days)
- Operations Manager 2007 Service Pack 1 Upgrade Package
- System Center Operations Manager Authoring Console

System Center Operations Manager 2007 Newsgroups

Microsoft public newsgroups are great places to exchange ideas with others and discuss common issues. You can read and write messages using an NNTP-based newsreader such as Microsoft Outlook Express. You can also use our Web-based newsreader to access all of the newsgroups.

- Agentless Exception Monitoring
- Audit Collection Discussions
- Operations Manager Active Directory Management Pack
- Operations Manager Authoring
- Operations Manager Documentation
- Operations Manager Exchange Management Pack
- Operations Manager General Discussions
- Operations Manager Globalization Localization
- Operations Manager IIS Management Pack
- Operations Manager Management Packs
- Operations Manager PowerShell
- Operations Manager Reporting
- Operations Manager SDK
- Operations Manager Setup
- Operations Manager SP1
- Operations Manager SQL Management Pack
- Operations Manager User Interface

Knowledgebase Articles

All Knowledgebase articles for System Center Operations Manager 2007

Subscribe to Knowledgebase articles through email

Operations Manager Tools

- Active Directory Integration Script

Enables you to extract a list of computer names from your custom SQL Server database, based on a specified query parameter, and add them to an Active Directory security group. You can assign the members of the security group to a specific Management Server. – OpsMgr Resource Kit

- System Center Capacity Planner

The official tool to determine server requirements for Operations Manager 2007

- Operations Manager Cleanup Tool

A command-line utility that enables you to remove all of the components of Operations Manager from a local computer in cases where the typical method through the Add/Remove Programs application in the Control Panel has failed. – OpsMgr Resource Kit

- Action Account Tool

A Windows PowerShell script that allows you to set the action account on multiple computers. – OpsMgr Resource Kit

- Effective Configuration Viewer

A tool that displays the set of rules and monitors that are running on a computer, distributed application, or any other managed entity after any configured overrides have been applied. – OpsMgr Resource Kit

- Operations Manager Inventory

A command-line utility that captures the configuration of your Operations Manager 2007 Management Servers and stores it in a .cab file that can be sent to Microsoft support to assist in problem analysis. – OpsMgr Resource Kit

- AEM Validation

A command line utility that will allow you to perform end-to-end validation of Agent-less Exception Monitoring to verify that AEM is properly configured and operational. – OpsMgr Resource Kit

- AEM Management Pack

A Management Pack that enables you to identify generic errors sent by Windows Error Reporting (WER) clients to Management Servers that are AEM-enabled. Without this mapping function these errors appear in Operations Manager as "unknown application" and "unknown version". Check the Management Pack Catalog as well. – OpsMgr Resource Kit

- MPViewer – **Boris Yanushpolsky, Microsoft**

Small utility to display the following contents of a management pack: Rules, Monitors, Views, Tasks, Console Tasks, and Reports. At the bottom of the window it will also show you the knowledge associated with the particular management pack item. The only requirement is that you have the OpsMgr console installed on the same computer.

- Override Explorer v3.3 – **Boris Yanushpolsky, Microsoft**

The purpose of override explorer is to simplify understanding what overrides exist in a management group. It provides two views:

Type Based - This view shows types for that have rules/monitors/discoveries for which overrides were created.

Computer Based - This is in essence a resultant set of overrides that apply to a computer. You can also drill in and see what overrides are applied to various computer components such as OS, Databases, Websites.

Except of viewing the overrides, you can also do the following by right clicking on an override:

1 - Move the override to a different MP. This is useful if in the past you saved overrides to the "Default Management pack"

2 - Change the target of the override. If you created an override to lets say disable a monitor for all SQL 2005 Databases and want to change it to only SQL 2005 Databases in a particular group, you can do it.

3 - Delete an override.

- Override Creator – **Boris Yanushpolsky, Microsoft**

"One of the things I have heard on several occasions is that people want to disable or enable a number of rules/monitors/discoveries at once. Currently it's only possible to enable or disable one at a time. This can be an issue when you are trying to disable 100 rules. To help with this problem I created a utility that allows you to multi-select a number of rules/monitors/discoveries and create an override to either disable them or enable them".

- Module Explorer – **Boris Yanushpolsky, Microsoft**

In v1 of module explorer I enumerate code based modules and then populated the tree view with composite module that rely on the code modules. In this version I added another view which instead groups modules by the MP in which the module is defined. This should help in some scenarios where you just want to find a particular module and see its settings and configuration.

- MPtoXML – **Boris Yanushpolsky, Microsoft**

Ever wanted to dig into a management pack but could not because it is sealed? One workaround is to import the management pack into a management group and then export it. There is a much quicker way to do this using PowerShell and the SDK.

Attached is a small script that will do the trick. The prerequisite for this script to work is that you have the OpsMgr console or Management server and PowerShell installed.

The script is attached to this post.

Here is how you can run the script to get the XML representation of a sealed MP:

powershell d:\MpToXml.ps1 -mpFilePath:'d:\Microsoft.Exchange.Server.2003.Monitoring.mp' -outputDirectory:'d:\'

- AgentMM **– Clive Eastwood, Microsoft**

Puts OpsMgr agents into maintenance mode

- Bulk Monitor State Reset (GreenMachine) **– Tim Helton, Microsoft**

Command line tool for resetting the health of all monitors for a single agent or all agents

- Data Warehouse Data Retention Policy Tool **– Daniel Savage, Microsoft**

Allows view and configure the data warehouse data retention policies configured for the Operations Manager Data Warehouse

- Eventlog Explorer **– Zbigniew Butor, Microsoft**

Exposes sources installed on a local machine, design and fire user selected sets of events involving multiple sources with one button click.
Great for identifying event parameter numbers for OpsMgr event filters.

- HealthServiceWatcher Connector **– Marius Sutara, Microsoft**

This OpsMgr connector eases the task of auto-ticketing and notifying computer owners of heartbeat failure associated with their systems.

- Maintenance Mode Web Application **- Steve Rachui, Microsoft**

Operates on both individual agents and groups; Allows 'right now' configuration of maintenance mode; Allows future scheduling of maintenance mode; Displays systems currently in maintenance mode.

- Management Pack Export Utility **– Neale Brown, External**

This utility provides several command line options for exporting sealed and unsealed MPs from both Operations Manager 2007 and System Center Essentials 2007.

- MGInfo **– Clive Eastwood, Microsoft**

Displays licensing and summary information about the Management Group

- OpsMgr Linear Explorer Build 316 **- Vin Dipippo, External**

This tool allows you to explore the lineage of OpsMgr MP Elements.

- ProxyCFG **– Clive Eastwood, Microsoft**

Configure and view agent proxy settings from the command line.

- ProxySettings 1.1 - **Boris Yanushpolsky, Microsoft**

GUI tool for configuring and viewing agent proxy settings.

- Restart Monitoring Tool **- Marius Sutara, Microsoft**

GUI tool that facilitates bulk reset health monitors and restart monitoring on demand for a group of monitored objects. Perfect for situations where you need to reset many health monitors at once

- Run As profile configuration helper v1 - **Boris Yanushpolsky, Microsoft**

Will allow you to configure a RunAs profile for agents in bulk

- SCOM Remote Maintenance Mode Scheduler **– Tim McFadden, External**

GUI based tool that lets administrators easily schedule maintenance mode for a server or group of servers inside System Center Operations Manager 2007.

- Subscription Tool **– Tim McFadden, External**

Subscription Tool is a simple GUI based tool that lets you enable and disable all notification subscriptions. The tool will automatically re-enable all subscriptions using "Send notifications for alerts Generated after the subscription is enabled"

- SVT Install Tool **– Ryan Brennan, External**

Remove override references from the MP you are trying to remove

- System Center Content Search Gadget **– Chris Scoville, External**

This gadget makes it easier to find help for Microsoft System Center products because it uses Live Search macros to search specific sites instead of the entire web

Microsoft Employee Blogs and Sites

System Center Operations Manager

Communications from the Operations Manager Product team

This site is the main Product Team Blog

The Operations Manager Support Team Blog

Communications from the Operations Manager Support team

This Site is the main Support Team Blog

BWren's Management Space

Brian Wren is the Lead Architect for MCS and MP development

Operations Manager Management packs and other automation technologies

OpsManJam

"Hello, and welcome to the OpsManJam Web site! On this site you will find unofficial management packs, management pack authoring tutorials and guidance, featured articles on everything OpsMgr 2007, command shell scripts, and more.

The content on this site comes from Microsoft Consulting Services, Microsoft IT and the Operations Manager Product team. While the content that you will find here *is not supported*

by Microsoft, it has been used in the field and is used everyday inside of Microsoft. Consider it to be useful tidbits from the experts. "

Author MP's

Steve Wilson is a Senior Program Manager Lead from the OpsMgr Product Group

The goal of this site is to provide the MP author with all that he or she needs to write great management packs. This includes things such as:

- Concepts
- MP Schema
- Module Types / Monitor Type library
- Tutorials
- Samples
- Authoring principles
- Application instrumentation guidance

OpsMgr ++

Boris Yanushpolsky is a Senior Program Manager from the OpsMgr Product Group

His site is dedicated to providing tools and resources to better manage Operations Manager

Walter Chomak's System Center Operations Manager 2007 Landing Zone

Walter Chomak is a Senior Consultant with Microsoft Consulting Services

His blog writes about all things OpsMgr in the Field. Especially noteworthy for his section on IO requirements

Matt Goedtel on Operations Management

Matt Goedtel is a Senior Consultant with Microsoft Consulting Services

His blog writes about all things OpsMgr in the Field. He has posted several extended Management Packs for some of our core products

Notes on System Center Operations Manager

Marius Sutara is a Senior Developer working on OpsMgr

Notes, troubleshooting, development and comments related to System Center Operations Manager

[Jakub@Work]

Jakub Oleksy is a Senior Development Lead with the Product Group

His Blog is mostly about SDK and Programmability with System Center Operations Manager

[System Center Operations Manager Command Shell]

Roger Sprague is a Senior Developer working on OpsMgr

The blog has been quiet for a while but still has very valuable entries in it

[OpsMgr, SCE And MOM]

Clive Eastwood is a Senior Program Manager with the OpsMgr Product Group

While this blog has not been updated in a while, past contributions are still valuable today

[Eugene Bykov OpsMgr Reporting]

Eugene Bykov is a Senior Developer working on OpsMgr.

His Blog has been quiet for a while, but he still has important posts for OpsMgr Reporting

[Kevin Holman's OpsMgr Blog]

Kevin Holman is a Senior Premier Field Engineer for Microsoft OpsMgr

“I am a Premier Field Engineer for Microsoft, and I support and consult around the System Center Microsoft Operations Manager product suite.

This blog is a collection of my random thoughts, and how-to's for anything I find interesting and think might help someone in the field.”

[Mother]

Justin Incarnato is a Senior Program Manager with the OpsMgr Product Group

His blog is quiet at the moment

[System Center WebLog by Russ Slaten]

Russ Slaten is a Senior Premier Field Engineer for Microsoft OpsMgr

Russ blogs about useful resources and procedures in the field

Nexus SC: The System Center Team Blog

Get filled in on the Microsoft System Center story.

Steve Rachui's Manageability blog - ConfigMgr/OpsMgr

Steve Rachui is a Premier Senior Field Engineer with Microsoft Dedicated Support Engineering

Steve blogs about useful resources and procedures in the field

Sampa @ Work

Sam Patton is a Senior Developer working on OpsMgr

Sam's Blog in the past has been Module oriented. Currently it is quiet, but still has lot of information for advanced MP development

External OpsMgr resource sites and forums:

http://systemcenterforum.org/

Incredible site full of scripts, knowledge, how-to's and other information

PETE'S MANAGEMENT BLOG....

Discursive discourse on Operations Manager 2007, Essentials 2007 (SCE), the MS System Center Suite, and IT systems management news de jour

Everything System Center Operations Manager 2007

Tim McFadden's OpsMgr Blog

Contoso.se

Anders Bengtsson, A Microsoft MVP has a great blog about System Center

http://www.systemcenterforum.org/tools/

OpsMgr Tools

WMI and WMI and OpsMgr 2007

WMI Diagnostics Tool

WMI Administrative Tools

WMI Events in OpsMgr 2007

Unlocking the Mystery of WMI Events in MOM

This one was for MOM 2005 but still applies for 2007

MSDN WMI Reference

Technet Administration and Scripting Tools

Using WBEMTest user interface

MSDN Windows Management Instrumentation

http://msdn.microsoft.com/en-us/library/aa394582(VS.85).aspx

http://blogs.msdn.com/wmi/archive/2006/05/12/596266.aspx#comments

http://www.microsoft.com/technet/scriptcenter/guide/sas_wmi_overview.mspx?mfr=true

WMI Scripting Primer

How to fix WMI Issues

By David Allen

After implementing Operations Manager in your environment you may find some of your servers experiencing WMI problems. This can result in not being able to collect data or monitor what you need as the WMI queries in the rules and monitors will fail.
Here are some simple command sets to reset and re-register WMI which in my experience I have found to solve many WMI issues.

Resetting WMI

1. net stop winmgmt
2. c:
3. cd %windir%\system32\wbem\

4. rmdir /s /q Repository
5. rmdir /s /q Logs
6. mkdir Logs
7. net start winmgmt

Re-registering WMI

1. net stop winmgmt
2. c:
3. cd %windir%\system32\wbem\
4. for %i in (*.dll) do RegSvr32 /s %i
5. for %i in (*.mof, *.mfl) do Mofcomp %i
6. net start winmgmt

Query a database with a monitor

By Anders Bengtsson

I have seen a number of questions about how to run queries against a database and verify the answer. One way is to run a script inside a monitor. In this blog I wrote how to setup a script in a two state monitor. The script in this post will count number of fields, if there are more than five, status of the monitor will be changed. Note that counting starts at 0 with fields collection.

```
Const adOpenStatic = 3
Const adLockOptimistic = 3

Set oAPI = CreateObject("MOM.ScriptAPI")
Set oBag = oAPI.CreatePropertyBag()

Set objConnection = CreateObject("ADODB.Connection")
Set objRecordSet = CreateObject("ADODB.Recordset")

objConnection.Open _
"Provider=SQLOLEDB;Data Source=R2B1;" & _
```

```
"Trusted_Connection=Yes;Initial Catalog=ContosoConfiguration;" & _
"User ID=CORP\Administrator;Password=;"

objRecordSet.Open "SELECT * FROM roles", _
objConnection, adOpenStatic, adLockOptimistic

varNo = objRecordSet.Record.Count

If varNO > 5 Then
Call oBag.AddValue("SQL_Status","Error")
Else
Call oBag.AddValue("SQL_Status","Ok")
End If

Call oAPI.Return(oBag)
```

SCOM command line parameters

By Rikard Ronnkvist

Agent - (MOMAgent.msi)

msiexec.exe /i \\path\Directory\MOMAgent.msi /qn /l*v \logs\MOMAgent_install.log USE_SETTINGS_FROM_AD=0 MANAGEMENT_GROUP= MANAGEMENT_SERVER_DNS= ACTIONS_USE_COMPUTER_ACCOUNT=0 ACTIONSUSER= ACTIONSDOMAIN= ACTIONSPASSWORD=

Typical - (MOM.msi)

msiexec.exe /i \\path\Directory\MOM.msi /qn /l*v \logs\MOMTypical_install.log ADDLOCAL=ALL USE_SETTINGS_FROM_AD=0 MANAGEMENT_GROUP= MANAGEMENT_SERVER_DNS= SQLSVR_INSTANCE= ADMIN_ROLE_GROUP="DomainName/Account" ACTIONS_USE_COMPUTER_ACCOUNT=0 ACTIONSUSER= ACTIONSDOMAIN= ACTIONSPASSWORD= SDK_USE_COMPUTER_ACCOUNT=0 SDK_ACCOUNT= SDK_DOMAIN= SDK_PASSWORD=

Database –(MOM.msi)

msiexec.exe /i \\path\Directory\MOM.msi /qn /l*v \logs\MOMDB_install.log ADDLOCAL=MOMDB USE_SETTINGS_FROM_AD=0 MANAGEMENT_GROUP= SQLSVR_INSTANCE= DB_SIZE=500 ADMIN_ROLE_GROUP= DATA_DIR= LOG_DIR=

Server –(MOM.msi)

msiexec.exe /i \\path\Directory\MOM.msi /qn /l*v \logs\MOMServer_install.log ADDLOCAL=MOMServer USE_SETTINGS_FROM_AD=0 MANAGEMENT_GROUP= MOM_DB_SERVER=

ACTIONS_USE_COMPUTER_ACCOUNT=0 ACTIONSUSER= ACTIONSDOMAIN= ACTIONSPASSWORD= SDK_USE_COMPUTER_ACCOUNT=0 SDK_ACCOUNT= SDK_DOMAIN= SDK_PASSWORD=

UI –(MOM.msi)

msiexec.exe /i \\path\Directory\MOM.msi /qn /l*v \logs\MOMUI_install.log ADDLOCAL=MOMUI USE_SETTINGS_FROM_AD=0 MANAGEMENT_GROUP= ROOT_MANAGEMENT_SERVER_DNS=

WebConsole (MOM.msi)

msiexec.exe /i \\path\Directory\MOM.msi /qn /l*v \logs\MOMUI_install.log ADDLOCAL=MOMWebConsole WEB_CONSOLE_AUTH_TYPE=0 ROOT_MANAGEMENT_SERVER_DNS= ('0' is windows auth and '1' is Forms auth)

Data Warehouse- (Reporting2007.msi)
msiexec.exe /i \\path\Directory\Reporting2007.msi /qn /l*v "D:\LOGS\REPORTING_INSTALL.LOG" ADDLOCAL="MOMREPORTINGDB" SQLSVR_INSTANCE="" MOMREPORTINGDBNAME="SCOMDW" DB_SIZE="1000"

Reporting Server (Reporting2007.msi)

msiexec.exe /i \\path\Directory\Reporting2007.msi /qn /l*v "D:\LOGS\REPORTING_INSTALL.LOG" ADDLOCAL="MOMREPORTING" SQLSVR_INSTANCE="" MOMREPORTINGDBNAME="SCOMDW" MGSERVER= PREREQ_COMPLETED="1" REPORT_SERVER_FULL_HTTP_PATH="http://%COMPUTERNAME%:80/ReportServer$INSTANCE1" DATAREADER_USER= DATAREADER_PASSWORD= DATAREADER_DOMAIN= DBWRITEACTIONSUSER= DBWRITEACTIONSPASSWORD= DBWRITEACTIONSDOMAIN=

Gateway Server (MOMGateway.msi)

msiexec /i \\path\Directory\MOMGateway.msi /qn /l*v D:\GATEWAY_SERVER_INSTALL_1.LOG ADDLOCAL=MOMGateway,MOMNonRootServer SECURE_PORT=5723 MANAGEMENT_GROUP=MyConfigGroup MANAGEMENT_SERVER_DNS= IS_ROOT_HEALTH_SERVER=0 ROOT_MANAGEMENT_SERVER_AD= ROOT_MANAGEMENT_SERVER_DNS= ROOT_MANAGEMENT_SERVER_PORT=5723 ACTIONS_USE_COMPUTER_ACCOUNT=0 ACTIONSUSER= ACTIONSDOMAIN= ACTIONSPASSWORD=

Audit Collection Server (AdtSetup.exe)

The cmd line you will use would look like this:\\PATH\Directory\AdtSetup.exe /i /s /p:ACSInstallParameters.xml
You will need to specify ACSInstallParameters.xml while passing the command line you will configure all your setting in the XML file before passing it. The XML file is attached.

For '/i' is for 'install', '/s' is for 'silent', and '/p' is for parameter (file)…

Useful Operations Manager 2007 SQL queries

By Kevin Holman

Operational Database Section:

Alerts Section:

Most Common console Alerts in an Operational Database, by Alert Count

```
SELECT AlertStringName, AlertStringDescription, Name, AlertParams, SUM(1) AS AlertCount,
SUM(RepeatCount+1) AS AlertCountWithRepeatCount
FROM Alertview WITH (NOLOCK)
WHERE TimeRaised is not NULL
GROUP BY AlertStringName, AlertStringDescription, AlertParams, Name
ORDER BY AlertCount DESC
```

Most Common console Alerts in an Operational Database, by Repeat Count

```
SELECT AlertStringName, AlertStringDescription, Name, AlertParams, SUM(1) AS AlertCount,
SUM(RepeatCount+1) AS AlertCountWithRepeatCount
FROM Alertview WITH (NOLOCK)
Where TimeRaised is not NULL
GROUP BY AlertStringName, AlertStringDescription, AlertParams, Name
ORDER BY AlertCountWithRepeatCount DESC
```

Number of console Alerts per Day:

```
SELECT CONVERT(VARCHAR(20), TimeAdded, 101) AS DayAdded, COUNT(*) AS NumAlertsPerDay
FROM Alert WITH (NOLOCK)
WHERE TimeRaised is not NULL
GROUP BY CONVERT(VARCHAR(20), TimeAdded, 101)
ORDER BY DayAdded DESC
```

Number of console Alerts per Day by Resolution State:

```
SELECT
CASE WHEN(GROUPING(CONVERT(VARCHAR(20), TimeAdded, 101)) = 1) THEN 'All Days' ELSE
CONVERT(VARCHAR(20), TimeAdded, 101) END AS [Date],
CASE WHEN(GROUPING(ResolutionState) = 1) THEN 'All Resolution States' ELSE CAST(ResolutionState AS
VARCHAR(5)) END AS [ResolutionState],
COUNT(*) AS NumAlerts
FROM Alert WITH (NOLOCK)
WHERE TimeRaised is not NULL
GROUP BY CONVERT(VARCHAR(20), TimeAdded, 101), ResolutionState WITH ROLLUP
ORDER BY DATE DESC
```

(Note: There will be more alerts in the "Alert" table in the form of rows, than exist in the console. This is because there are non-console alerts where TimeRaised is NULL - these have to do with driving state change records, and are not included in the above queries by design)

Events Section:

All Events by count by day, with total for entire database:

(this tells us how many events per day we are inserting - and helps us look for too many events, event storms, and the result after tuning rules that generate too many events)

```
SELECT CASE WHEN(GROUPING(CONVERT(VARCHAR(20), TimeAdded, 101)) = 1)
THEN 'All Days'
ELSE CONVERT(VARCHAR(20), TimeAdded, 101) END AS DayAdded,
COUNT(*) AS NumEventsPerDay
FROM EventAllView
GROUP BY CONVERT(VARCHAR(20), TimeAdded, 101) WITH ROLLUP
ORDER BY DayAdded DESC
```

Most Common Events by event number:

(This helps us know which event ID's are the most common in the database)

```
SELECT top 50 Number, COUNT(*) AS TotalEvents
FROM EventView
GROUP BY Number
ORDER BY TotalEvents DESC
```

Most common events by event number and event publishername:

(This gives us the event publisher name to help see what is raising these events)

```
SELECT top 50 Number, Publishername, COUNT(*) AS TotalEvents
FROM EventAllView
GROUP BY Number, Publishername
ORDER BY TotalEvents DESC
```

Most common events, grouped by identical event number, publisher name, and event parameters:

(*This shows use completely redundant events with identical data - but might be different than the above queries... you need to see both data outputs to fully tune)*

```
SELECT top 100 Number, Publishername, EventParameters, COUNT(*) AS TotalEvents
FROM EventAllView
```

```
GROUP BY Number, Publishername, EventParameters
ORDER BY TotalEvents DESC
```

Computers generating the most events:

(This shows us which computers create the most event traffic and use the most database space)

```
SELECT top 50 LoggingComputer, COUNT(*) AS TotalEvents
FROM EventallView
GROUP BY LoggingComputer
ORDER BY TotalEvents DESC
```

Computers generating the most events, by event number: *(This shows the noisiest computers, group by unique event numbers)*

```
SELECT top 50 LoggingComputer, Number, COUNT(*) AS TotalEvents
FROM EventallView
GROUP BY LoggingComputer, Number
ORDER BY TotalEvents DESC
```

Computers generating the most events, grouped by identical event number and publisher name:

```
SELECT top 50 LoggingComputer, PublisherName, Number, COUNT(*) AS TotalEvents
FROM EventallView
GROUP BY LoggingComputer, PublisherName, Number
ORDER BY TotalEvents DESC
```

Performance Section:

Performance insertions per day:

```
SELECT CASE WHEN(GROUPING(CONVERT(VARCHAR(20), TimeSampled, 101)) = 1)
THEN 'All Days' ELSE CONVERT(VARCHAR(20), TimeSampled, 101)
END AS DaySampled, COUNT(*) AS NumPerfPerDay
FROM PerformanceDataAllView
GROUP BY CONVERT(VARCHAR(20), TimeSampled, 101) WITH ROLLUP
ORDER BY DaySampled DESC
```

Most common performance insertions by perf counter name:

```
select pcv.countername, count (pcv.countername) as total from performancedataallview as pdv,
performancecounterview as pcv
where (pdv.performancesourceinternalid = pcv.performancesourceinternalid)
group by pcv.countername
order by count (pcv.countername) desc
```

Most common performance insertions by perf object name:

```
select pcv.objectname, count (pcv.countername) as total from performancedataallview as pdv,
performancecounterview as pcv
where (pdv.performancesourceinternalid = pcv.performancesourceinternalid)
group by pcv.objectname
order by count (pcv.countername) desc
```

Most common performance insertions by perf object and counter name:

(This is the most interesting - tells us specifically which perf insertions are the noisiest)

```
select pcv.objectname, pcv.countername, count (pcv.countername) as total from
performancedataallview as pdv, performancecounterview as pcv
where (pdv.performancesourceinternalid = pcv.performancesourceinternalid)
group by pcv.objectname, pcv.countername
order by count (pcv.countername) desc
```

To retrieve all performance data for a given rule in a readable format use the following query:

```
SELECT pc.ObjectName, pc.CounterName, ps.PerfmonInstanceName, pd.SampleValue, pd.TimeSampled
FROM PerformanceDataAllView AS pd, PerformanceCounter AS pc, PerformanceSource AS ps WHERE
pd.PerformanceSourceInternalId IN (SELECT PerformanceSourceInternalId FROM PerformanceSource
WHERE RuleId = (SELECT RuleId FROM Rules WHERE RuleName ='
Microsoft.Windows.Server.2003.LogicalDisk.FreeSpace.Collection'))
```

To view all performance insertions for a given computer:

```
select path, objectname, countername, instancename, samplevalue, timesampled from
PerformanceDataAllView pdv
inner join PerformanceCounterView pcv on pdv.performancesourceinternalid =
pcv.performancesourceinternalid
```

```
inner join BaseManagedEntity bme on pcv.ManagedEntityId = bme.BaseManagedEntityId
where path = 'dc1.opsmgr.net'
order by countername
```

To refine the above query to pull all perf data for a given computer, object, counter, and instance:

```
select path, objectname, countername, instancename, samplevalue, timesampled from
PerformanceDataAllView pdv
inner join PerformanceCounterView pcv on pdv.performancesourceinternalid =
pcv.performancesourceinternalid
inner join BaseManagedEntity bme on pcv.ManagedEntityId = bme.BaseManagedEntityId
where path = 'dc1.opsmgr.net' AND
objectname = 'Process' AND
countername = '% Processor Time' AND
instancename = 'HealthService'
order by timesampled
```

State Section:

State changes per day:

```
SELECT CASE WHEN(GROUPING(CONVERT(VARCHAR(20), TimeGenerated, 101)) = 1)
THEN 'All Days' ELSE CONVERT(VARCHAR(20), TimeGenerated, 101)
END AS DayGenerated, COUNT(*) AS NumEventsPerDay
FROM StateChangeEvent WITH (NOLOCK)
GROUP BY CONVERT(VARCHAR(20), TimeGenerated, 101) WITH ROLLUP
ORDER BY DayGenerated DESC
```

Management Pack info:

Rules section:

Rules per MP:

```
SELECT mp.MPName, COUNT(*) As RulesPerMP
FROM Rules r
INNER JOIN ManagementPack mp ON mp.ManagementPackID = r.ManagementPackID
GROUP BY mp.MPName
ORDER BY RulesPerMP DESC
```

Rules per MP by category:

```
SELECT mp.MPName, r.RuleCategory, COUNT(*) As RulesPerMPPerCategory
FROM Rules r
INNER JOIN ManagementPack mp ON mp.ManagementPackID = r.ManagementPackID
GROUP BY mp.MPName, r.RuleCategory
ORDER BY RulesPerMPPerCategory DESC
```

To find all Rules per MP that generate an alert:

```
declare @mpid as varchar(50)
select @mpid= managementpackid from managementpack where
mpName='Microsoft.Exchange.2007'
select rl.rulename,rl.ruleid,md.modulename from rules rl, module md
where md.managementpackid = @mpid
and rl.ruleid=md.parentid
and moduleconfiguration like '%<AlertLevel>50</AlertLevel>%'
```

To find all rules per MP with a given alert severity:

```
declare @mpid as varchar(50)
select @mpid= managementpackid from managementpack where
mpName='Microsoft.Exchange.Server.2003.Monitoring'
select rl.rulename,rl.ruleid,md.modulename from rules rl, module md
where md.managementpackid = @mpid
and rl.ruleid=md.parentid
and moduleconfiguration like '%<Severity>2</Severity>%'
```

Rules are stored in a table named Rules. This table has columns linking rules to classes and Management Packs. To find all rules in a Management Pack use the following query and substitute in the required Management Pack name:

```
SELECT * FROM Rules WHERE ManagementPackID = (SELECT ManagementPackID from
ManagementPack WHERE MPName = 'Microsoft.Windows.Server.2003')
```

To find all rules targeted at a given class use the following query and substitute in the required class name:

```
SELECT * FROM Rules WHERE TargetManagedEntityType = (SELECT ManagedTypeId FROM
ManagedType WHERE TypeName = 'Microsoft.Windows.Computer')
```

Monitors Section:

Monitors Per MP:

```
SELECT mp.MPName, COUNT(*) As MonitorsPerMPPerCategory
FROM Monitor m
INNER JOIN ManagementPack mp ON mp.ManagementPackID = m.ManagementPackID
GROUP BY mp.MPName
ORDER BY COUNT(*) Desc
```

To find your Monitor by common name:

```
select * from Monitor m
Inner join LocalizedText LT on LT.ElementName = m.MonitorName
where LTValue = 'Monitor Common Name'
```

To find your Monitor by ID name:

```
select * from Monitor m
Inner join LocalizedText LT on LT.ElementName = m.MonitorName
where m.monitorname = 'Monitor ID name'
```

To find all monitors targeted at a specific class:

```
SELECT * FROM monitor WHERE TargetManagedEntityType = (SELECT ManagedTypeId FROM
ManagedType WHERE TypeName = 'Microsoft.Windows.Computer')
```

Management Pack general:

To find all installed Management Packs and their version:

```
SELECT MPName, MPFriendlyName, MPVersion, MPIsSealed
FROM ManagementPack WITH(NOLOCK)
ORDER BY MPName
```

Number of Views per Management Pack:

```
SELECT mp.MPName, v.ViewVisible, COUNT(*) As ViewsPerMP
FROM [Views] v
        INNER JOIN ManagementPack mp ON mp.ManagementPackID = v.ManagementPackID
GROUP BY  mp.MPName, v.ViewVisible
ORDER BY v.ViewVisible DESC, COUNT(*) Desc
```

Classes available in the DB:

```
SELECT * FROM ManagedType
```

Classes available in the DB for Microsoft Windows type:

```
SELECT * FROM ManagedType WHERE TypeName LIKE 'Microsoft.Windows.%'
```

Every property of every class:

```
SELECT * FROM MT_Computer
```

All instances of all types once discovered

```
SELECT * FROM BaseManagedEntity
```

To get the state of every instance of a particular monitor the following query can be run, (replace <MonitorName> with the name of the monitor):

```
SELECT bme.FullName, bme.DisplayName, s.HealthState FROM state AS s, BaseManagedEntity as bme
WHERE s.basemanagedentityid = bme.basemanagedentityid AND s.monitorid IN (SELECT MonitorId
FROM Monitor WHERE MonitorName = =<MonitorName>‘)
```

For example, this gets the state of the Microsoft.SQLServer.2005.DBEngine.ServiceMonitor for each instance of the SQL 2005 Database Engine class.

```
SELECT bme.FullName, bme.DisplayName, s.HealthState FROM state AS s, BaseManagedEntity as bme
WHERE s.basemanagedentityid = bme.basemanagedentityid AND s.monitorid IN (SELECT MonitorId
FROM Monitor WHERE MonitorName = 'Microsoft.SQLServer.2005.DBEngine.ServiceMonitor')
```

To find the overall state of any object in OpsMgr the following query should be used to return the state of the System.EntityState monitor:

```
SELECT bme.FullName, bme.DisplayName, s.HealthState FROM state AS s, mt_managedcomputer AS mt,
BaseManagedEntity as bme WHERE s.basemanagedentityid = bme.basemanagedentityid AND
s.monitorid IN (SELECT MonitorId FROM Monitor WHERE MonitorName = 'System.Health.EntityState')
```

The Alert table contains all alerts currently open in OpsMgr. This includes resolved alerts until they are groomed out of the database. To get all alerts across all instances of a given monitor use the following query and substitute in the required monitor name:

```
SELECT * FROM Alert WHERE ProblemID IN (SELECT MonitorId FROM Monitor WHERE MonitorName =
'Microsoft.SQLServer.2005.DBEngine.ServiceMonitor')
```

To retrieve all alerts for all instances of a specific class use the following query and substitute in the required table name, in this example MT_DBEngine is used to look for SQL alerts:

```
SELECT * FROM Alert WHERE BaseManagedEntityID IN (SELECT BaseManagedEntityID from
MT_DBEngine)
```

To determine which table is currently being written to for event and performance data use the following query:

```
SELECT * FROM PartitionTables WHERE IsCurrent = 1
```

To retrieve events generated by a specific rule use the following query and substitute in the required rule ID:

```
SELECT * FROM Event_00 WHERE RuleId = (SELECT RuleId FROM Rules WHERE RuleName =
'Microsoft.Windows.Server.2003.OperatingSystem.CleanShutdown.Collection ')
```

To retrieve all events generated by rules in a specific Management Pack the following query can be used where the Management Pack name is substituted with the required value:

```
SELECT * FROM EventAllView WHERE RuleID IN (SELECT RuleId FROM Rules WHERE ManagementPackId
= (SELECT ManagementPackId FROM ManagementPack WHERE MPName =
'Microsoft.Windows.Server.2003'))
```

Number of instances of a type: (Number of disks, computers, databases, etc that OpsMgr has discovered)

```
SELECT mt.ManagedTypeID, mt.TypeName, COUNT(*) AS NumEntitiesByType
FROM BaseManagedEntity bme WITH(NOLOCK)
        LEFT JOIN ManagedType mt WITH(NOLOCK) ON mt.ManagedTypeID = bme.BaseManagedTypeID
WHERE bme.IsDeleted = 0
GROUP BY mt.ManagedTypeID, mt.TypeName
ORDER BY COUNT(*) DESC
```

Agent Info:

To find all managed computers that are currently down and not pingable:

```
SELECT bme.DisplayName,s.LastModified as LastModifiedUTC, dateadd(hh,-5,s.LastModified) as
'LastModifiedCST (GMT-5)'
FROM state AS s, BaseManagedEntity AS bme
WHERE s.basemanagedentityid = bme.basemanagedentityid
AND s.monitorid
IN (SELECT MonitorId FROM Monitor WHERE MonitorName =
'Microsoft.SystemCenter.HealthService.ComputerDown')
AND s.Healthstate = '3' AND bme.IsDeleted = '0'
ORDER BY s.Lastmodified DESC
```

All managed computers count:

```
SELECT COUNT(*) AS NumManagedComps FROM (
SELECT bme2.BaseManagedEntityID
FROM BaseManagedEntity bme WITH (NOLOCK)
        INNER JOIN BaseManagedEntity bme2 WITH (NOLOCK) ON bme2.BaseManagedEntityID =
```

```
bme.TopLevelHostEntityID
WHERE bme2.IsDeleted = 0
        AND bme2.IsDeleted = 0
        AND bme2.BaseManagedTypeID = (SELECT TOP 1 ManagedTypeID FROM ManagedType WHERE
TypeName = 'microsoft.windows.computer')
GROUP BY bme2.BaseManagedEntityID
) AS Comps
```

To find a computer name from a HealthServiceID (guid from the Agent proxy alerts)

```
select DisplayName, Path, basemanagedentityid from basemanagedentity where basemanagedentityid
= 'guid'
```

To view the agent patch list (all hotfixes applied to all agents)

```
select bme.path AS 'Agent Name', hs.patchlist AS 'Patch List' from MT_HealthService hs
inner join BaseManagedEntity bme on hs.BaseManagedEntityId = bme.BaseManagedEntityId
order by path
```

To view all agents missing a specific hotfix (change the KB number below to the one you are looking for):

```
select bme.path AS 'Agent Name', hs.patchlist AS 'Patch List' from MT_HealthService hs
inner join BaseManagedEntity bme on hs.BaseManagedEntityId = bme.BaseManagedEntityId
where hs.patchlist not like '%951380%'
order by path
```

Misc:

To view grooming info:

```
SELECT * FROM PartitionAndGroomingSettings WITH (NOLOCK)
```

Information on existing User Roles:

```
SELECT UserRoleName, IsSystem from userrole
```

Operational DB version:

```
select DBVersion from __MOMManagementGroupInfo__
```

To view all Run-As Profiles, their associated Run-As account, and associated agent name:

```
select srv.displayname as 'RunAs Profile Name',
srv.description as 'RunAs Profile Description',
cmss.name as 'RunAs Account Name',
cmss.description as 'RunAs Account Description',
cmss.username as 'RunAs Account Username',
cmss.domain as 'RunAs Account Domain',
mp.FriendlyName as 'RunAs Profile MP',
bme.displayname as 'HealthService'
from dbo.SecureStorageSecureReference sssr
inner join SecureReferenceView srv on srv.id = sssr.securereferenceID
inner join CredentialManagerSecureStorage cmss on cmss.securestorageelementID =
sssr.securestorageelementID
inner join managementpackview mp on srv.ManagementPackId = mp.Id
inner join BaseManagedEntity bme on bme.basemanagedentityID = sssr.healthserviceid
order by srv.displayname
```

Data Warehouse:

Grooming in the DataWarehouse:

Grooming no longer uses SQL agent jobs. Grooming is handled by scheduled stored procedures, that run much more frequently, which provides less impact than in the previous version.

Default grooming for the DW for each dataset, to examine Data Warehouse grooming settings:

```
SELECT AggregationIntervalDurationMinutes, BuildAggregationStoredProcedureName,
GroomStoredProcedureName, MaxDataAgeDays, GroomingIntervalMinutes FROM
StandardDatasetAggregation
```

The first row is the interval in minutes.
NULL is raw data, 60 is hourly, and 1440 is daily.
The second and third row shows what data it is
MaxDataAgeDays has the retention period in days - this is the field to update if the administrator wants to lower the days of retention.
RAW alert – 400 days
RAW event – 100 days
RAW perf – 10 days (hourly and daily perf = 400 days)
RAW state – 180 days (hourly and daily state = 400 days)

Here is a better view of the current data retention in your data warehouse:

```
select ds.datasetDefaultName AS 'Dataset Name', sda.AggregationTypeId AS 'Agg Type 0=raw,
20=Hourly, 30=Daily', sda.MaxDataAgeDays AS 'Retention Time in Days'
from dataset ds, StandardDatasetAggregation sda
WHERE ds.datasetid = sda.datasetid
ORDER by ds.datasetDefaultName
```

To view the number of days of total data of each type in the DW:

```
SELECT DATEDIFF(d, MIN(DWCreatedDateTime), GETDATE()) AS [Current] FROM Alert.vAlert
SELECT DATEDIFF(d, MIN(DateTime), GETDATE()) AS [Current] FROM Event.vEvent
SELECT DATEDIFF(d, MIN(DateTime), GETDATE()) AS [Current] FROM Perf.vPerfRaw
SELECT DATEDIFF(d, MIN(DateTime), GETDATE()) AS [Current] FROM Perf.vPerfHourly
SELECT DATEDIFF(d, MIN(DateTime), GETDATE()) AS [Current] FROM Perf.vPerfDaily
SELECT DATEDIFF(d, MIN(DateTime), GETDATE()) AS [Current] FROM State.vStateRaw
SELECT DATEDIFF(d, MIN(DateTime), GETDATE()) AS [Current] FROM State.vStateHourly
SELECT DATEDIFF(d, MIN(DateTime), GETDATE()) AS [Current] FROM State.vStateDaily
```

To view the oldest and newest recorded timestamps of each data type in the DW:

```
select min(DateTime) from Event.vEvent
select max(DateTime) from Event.vEvent
select min(DateTime) from Perf.vPerfRaw
select max(DateTime) from Perf.vPerfRaw
select min(DWCreatedDateTime) from Alert.vAlert
select max(DWCreatedDateTime) from Alert.vAlert
```

To inspect total events in DW, and then break it down per day: (this helps us know what we will be grooming out, and look for particular day event storms)

```
SELECT CASE WHEN(GROUPING(CONVERT(VARCHAR(20), DateTime, 101)) = 1)
THEN 'All Days'
ELSE CONVERT(VARCHAR(20), DateTime, 101) END AS DayAdded,
COUNT(*) AS NumEventsPerDay
FROM Event.vEvent
GROUP BY CONVERT(VARCHAR(20), DateTime, 101) WITH ROLLUP
ORDER BY DayAdded DESC
```

Most Common Events by event number: (This helps us know which event ID's are the most common in the database)

```
SELECT top 50 EventDisplayNumber, COUNT(*) AS TotalEvents
FROM Event.vEvent
GROUP BY EventDisplayNumber
ORDER BY TotalEvents DESC
```

Most common events by event number and raw event description (this will take a very long time to run but it shows us not only event ID - but a description of the event to help understand which MP is the generating the noise)

```
SELECT top 50 EventDisplayNumber, Rawdescription, COUNT(*) AS TotalEvents
FROM Event.vEvent evt
inner join Event.vEventDetail evtd on evt.eventoriginid = evtd.eventoriginid
GROUP BY EventDisplayNumber, Rawdescription
ORDER BY TotalEvents DESC
```

To view all event data in the DW for a given Event ID:

```
select * from Event.vEvent ev
inner join Event.vEventDetail evd on ev.eventoriginid = evd.eventoriginid
inner join Event.vEventParameter evp on ev.eventoriginid = evp.eventoriginid
where eventdisplaynumber = '528'
```

To search for all computers who have NOT logged a specific event in the DW:

```
select distinct elc.computername from Event.vEvent ev
inner join vEventLoggingComputer elc on elc.eventloggingcomputerrowid = ev.loggingcomputerrowid
where NOT eventdisplaynumber = '223'
```

To get all raw alert data from the data warehouse to build reports from:

```
select * from Alert.vAlertResolutionState ars
inner join Alert.vAlertDetail adt on ars.alertguid = adt.alertguid
inner join Alert.vAlert alt on ars.alertguid = alt.alertguid
```

To view data on all alerts modified by a specific user:

```
select ars.alertguid, alertname, alertdescription, statesetbyuserid, resolutionstate, statesetdatetime,
severity, priority, managedentityrowID, repeatcount
from Alert.vAlertResolutionState ars
inner join Alert.vAlert alt on ars.alertguid = alt.alertguid
where statesetbyuserid like '%username%'
order by statesetdatetime
```

To view a count of all alerts closed by all users:

```
select statesetbyuserid, count(*) as 'Number of Alerts'
from Alert.vAlertResolutionState ars
where resolutionstate = '255'
group by statesetbyuserid
order by 'Number of Alerts' DESC
```

AEM Queries (Data Warehouse):

Default query to return all RAW AEM data:

```
select * from [CM].[vCMAemRaw] Rw
inner join dbo.AemComputer Computer on Computer.AemComputerRowID = Rw.AemComputerRowID
inner join dbo.AemUser Usr on Usr.AemUserRowId = Rw.AemUserRowId
inner join dbo.AemErrorGroup EGrp on Egrp.ErrorGroupRowId = Rw.ErrorGroupRowId
Inner join dbo.AemApplication App on App.ApplicationRowId = Egrp.ApplicationRowId
```

Count the raw crashes per day:

```
SELECT CONVERT(char(10), DateTime, 101) AS "Crash Date (by Day)", COUNT(*) AS "Number of Crashes"
FROM [CM].[vCMAemRaw]
GROUP BY CONVERT(char(10), DateTime, 101)
ORDER BY "Crash Date (by Day)" DESC
```

Count the total number of raw crashes in the DW database:

```
select count(*) from CM.vCMAemRaw
```

Default grooming for the DW for the AEM dataset: (Aggregated data kept for 400 days, RAW 30 days by default)

```
SELECT AggregationTypeID, BuildAggregationStoredProcedureName, GroomStoredProcedureName,
MaxDataAgeDays, GroomingIntervalMinutes
FROM StandardDatasetAggregation WHERE BuildAggregationStoredProcedureName = 'AemAggregate'
```

Misc Section:

Simple query to display large tables, to determine what is taking up space in the database:

```
SELECT so.name,
8 * Sum(CASE WHEN si.indid IN (0, 1) THEN si.reserved END) AS data_kb,
Coalesce(8 * Sum(CASE WHEN si.indid NOT IN (0, 1, 255) THEN si.reserved END), 0) AS index_kb,
Coalesce(8 * Sum(CASE WHEN si.indid IN (255) THEN si.reserved END), 0) AS blob_kb
FROM dbo.sysobjects AS so JOIN dbo.sysindexes AS si ON (si.id = so.id)
WHERE 'U' = so.type GROUP BY so.name  ORDER BY data_kb DESC
```

Is SQL broker enabled?

```
SELECT is_broker_enabled FROM sys.databases WHERE name = 'OperationsManager'
```

How to identify your version of SQL server:

```
SELECT  SERVERPROPERTY('productversion'), SERVERPROPERTY ('productlevel'), SERVERPROPERTY
('edition')
```

SQL 2005:
SQL Server 2005 RTM 2005.90.1399
SQL Server 2005 SP1 2005.90.2047
SQL Server 2005 SP1 plus 918222 2005.90.2153
SQL Server 2005 SP2 2005.90.3042

How to identify your version of OpsMgr 2007:

RTM: 6.0.5000.0

SP1-RC: 6.0.6246.0

SP1: 6.0.6278.0

To get better performance manually:

Update Statistics (will help speed up reports and takes less time than a full reindex):

```
EXEC sp_updatestats
```

Show index fragmentation (to determine how badly you need a reindex – logical scan frag > 10% = bad. Scan density below 80 = bad):

```
DBCC SHOWCONTIG
```

DBCC SHOWCONTIG WITH FAST (less data than above – in case you don't have time)

Reindex the database:

```
USE OperationsManager
go
SET ANSI_NULLS ON
SET ANSI_PADDING ON
SET ANSI_WARNINGS ON
SET ARITHABORT ON
SET CONCAT_NULL_YIELDS_NULL ON
SET QUOTED_IDENTIFIER ON
SET NUMERIC_ROUNDABORT OFF
EXEC SP_MSForEachTable "Print 'Reindexing '+'?' DBCC DBREINDEX ('?')"
```

Table by table:

```
DBCC DBREINDEX ('TableName')
```

Query to view the index job history on domain tables in the databases:

```
select *
from DomainTable dt
inner join DomainTableIndexOptimizationHistory dti
on dt.domaintablerowID = dti.domaintableindexrowID
ORDER BY optimizationdurationseconds DESC
```

Query to view the update statistics job history on domain tables in the databases:

```
select *
from DomainTable dt
inner join DomainTableStatisticsUpdateHistory dti
on dt.domaintablerowID = dti.domaintablerowID
ORDER BY UpdateDurationSeconds DESC
```

Data Warehouse query to examine the index and statistics history - run the following query for the Alert, Event, Perf, and State tables (these are non-domain tables):

```
select basetablename, optimizationstartdatetime, optimizationdurationseconds,
    beforeavgfragmentationinpercent, afteravgfragmentationinpercent,
    optimizationmethod, onlinerebuildlastperformeddatetime
from StandardDatasetOptimizationHistory sdoh
inner join StandardDatasetAggregationStorageIndex sdasi
on sdoh.StandardDatasetAggregationStorageIndexRowId =
sdasi.StandardDatasetAggregationStorageIndexRowId
inner join StandardDatasetAggregationStorage sdas
on sdasi.StandardDatasetAggregationStorageRowId = sdas.StandardDatasetAggregationStorageRowId
ORDER BY optimizationdurationseconds DESC
```

Regular expression support in SCOM 2007

Many teams that are authoring management packs may need to include regular expression matching in their discoveries and groups, as well as for pattern matching in expression criteria in monitors and rules.

There are two different types of regular expression support in the SCOM product, and you have to know which element you are working in to choose the correct one. Specifically, Group membership calculation and expression filters use distinctly different syntaxes for pattern matching.

Group Calculation matching criteria

Group calculation uses PERL regular expression syntax. By default, the matching is case insensitive, but in the XML you can specify that an expression needs to be case sensitive by way of a special attribute dedicated to specifying that the expression content should be evaluated in a case sensitive way.

Group Calculation is found in your MP whenever you are using the Group Calc module.

The GroupCalc expression has an operator called MatchesRegularExpression that is used to create dynamic group membership based on pattern matching expressions. The implementation of this operator passes the expression found in the MP XML to the SQL call name dbo.fn_MatchesRegularExpression. If this call returns 0, the match is false. If the expression returns 1, the match is true.

GroupCalc also supports two special sub-elements that abstract away a couple of common regex style queries.

GroupCalc sub element	Regex Equivalent	
ContainsSubstring	^*{O}.*$ ({O} is replaced by the substring)	
MatchesWildcard	MP expression	Regex Equivalent
	?	.
	*	.*
	#	[0-9]

Table 1: GroupCalc special functions

> ***Note: If either of these two special operators are used, the evaluation will always be case sensitive.***

Expression Filter matching criteria

Expression filters used in management packs use .NET Regex expression syntax. A summary of the .NET regular expression syntax elements appears below. Expression filters are present in your management pack whenever you are using the Expression Eval module.

Construct	SCOM Regex
Any Character	.
Character in Range	[ ]
Character not in range	[^]
Beginning of Line	^

End of Line	$
Or	|
Group	( )
0 or 1 matches	?
0 or more matches	*
1 or more matches	+
Exactly N matches	{n}
Atleast N matches	{n, }
Atmost N matches	{ , n}
N to M Matches	{n, m}
New line character	\n
Tab character	\t

Regular expressions via SDK

The SCOM SDK has a ***Matches*** criteria operator for filtering objects. This operator use the same functionality as MatchesCriteria in the GroupCalc case explained above.

When using the SDK to construct a criteria expression to find objects in the Ops Manager database, the following syntax elements are valid (see below). This syntax is useful when creating a criteria expression that includes any of the following elements:

- Comparison operators
- Wildcard characters
- DateTime values
- Integer to XML Enumeration comparisons

Comparison operators

You can use comparison operators when constructing a criteria expression. The valid operators are described in the following table:

Operator	Description	Example(s)
=, **==**	Evaluates to **true** if the left and right operand are equal.	`Name = 'mymachine.mydomain.com'`
!=, **<>**	Evaluates to **true** if the left and right operand are unequal.	`Name != 'mymachine.mydomain.com'`
>	Evaluates to **true** if the left operand is greater than the right operand.	`Severity > 0`
<	Evaluates to **true** if the left operand is	`Severity < 2`

	less than the right operand.	
>=	Evaluates to **true** if the left operand is greater than or equal to the right operand.	`Severity >= 1`
<=	Evaluates to **true** if the left operand is less than or equal to the right operand.	`Severity <= 3`

LIKE	Evaluates to **true** if the left operand matches the pattern that is defined by the right operand. Use the characters in the wildcard table later in this topic to define the pattern.	`Name 'LIKE SQL%'`
		Evaluates to **true** if the **Name** value is "SQLEngine."
		`Name LIKE '%SQL%'`
		Evaluates to **true** if the **Name** value is "MySQLEngine."

MATCHES	Evaluates to **true** if the left operand matches the regular expression defined by the right operand.	`Name MATCHES 'SQL*05'` Evaluates to **true** if the **Name** value is "SQL2005."
IS NULL	Evaluates to **true** if the value of the left operand is null.	`ConnectorId IS NULL` Evaluates to **true** if the **ConnectorId** property does not contain a value.
IS NOT NULL	Evaluates to **true** if the value of the left operand is not null.	`ConnectorId IS NOT NULL` Evaluates to **true** if the **ConnectorId** property contains a value.
IN	Evaluates to **true** if the value of the left operand is in the list of values defined by the right operand. **Note** The **IN** operator is valid for use only with properties of type Guid.	`Id IN ('080F192C-52D2-423D-8953-B3EC8C3CD001', '080F192C-53B2-403D-8753-B3EC8C3CD002')` Evaluates to **true** if the value of the **Id** property is one of the two globally unique identifiers provided in the expression.
AND	Evaluates to **true** if the left and right operands are both true.	`Name = 'SQL%' AND Description LIKE 'MyData%'`
OR	Evaluates to **true** if either the left or right operand is true.	`Name = 'SQL%' OR Description LIKE 'MyData%'`
NOT	Evaluates to **true** if the right operand is not true.	`NOT (Name = 'IIS' OR Name = 'SQL')`

Table 3: SDK comparison operators

Wildcards

The following table defines the wildcard characters you can use to construct a pattern when using the **LIKE** operator:

Wildcard	Description	Example

%	A wildcard that matches any number of characters.	`Name LIKE 'SQL%'`
		Evaluates to **true** if the **Name** value is "SQLEngine."
		`Name LIKE '%SQL%'`
		Evaluates to true if the **Name** value is "MySQLEngine."

_	A wildcard that matches a single character.	`Name LIKE 'SQL200_'` Evaluates to **true** for the following **Name** values: "SQL2000" "SQL2005" **Note** The expression evaluates to **false** for "SQL200" because the symbol _ must match exactly one character in the **Name** value.
[]	A wildcard that matches any one character that is enclosed in the character set. **Note** Brackets are also used when qualifying references to MonitoringObject properties. For more information, see Defining Queries for Monitoring Objects.	`Name LIKE 'SQL200[05]'` Evaluates to **true** for the following **Name** values: "SQL2000" "SQL2005" The expression evaluates to **false** for "SQL2003."
[^]	A wildcard that matches any one character that is not enclosed in the character set.	`Name LIKE 'SQL200[^05]'` Evaluates to **true** for "SQL2003." The expression evaluates to **false** for "SQL2000" and "SQL2005."

Table 4: Wildcard operators used with LIKE operator

DateTime comparisons

When you use a DateTime value in a query expression, use the general DateTime format ("G") to convert the **DateTime** value to a string value. For example,

C#

```
string qStr = "TimeCreated <= '" + myInstant.ToString("G") + "'";
ManagementPackCriteria mpCriteria = new ManagementPackCriteria(qStr);
```

All date values need to be converted to the G format (GMT) so that valid string comparisons can be made.

Integer value comparison to enumerations

When you use an integer enumeration value in a query expression, cast the enumeration value to an integer. For example,

C#

```
string qStr = "Severity > " + (int)ManagementPackAlertSeverity.Warning;
MonitoringAlertCriteria alertCriteria = new MonitoringAlertCriteria(qStr);
```

Operations Manager 2007 Command Shell

By Jonathan Almquist

Have you started using the Operations Manager Command Shell? Using PowerShell for Operations Manager tasks can save a lot of time, and helps automate some of those mundane chores we sometimes find ourselves faced with. Like resolving alerts generated from an alert storm perhaps? Or pushing agents during a deployment? Using Command Shell is also a great tool while performing tuning efforts!

I've started a list of commands that I've found useful at one time or another. Most of these are one-liners and can be pasted directly into Command Shell. Replace any text in *red* with your input.

Check back now and then, as I'll be adding more examples periodically.

Agent

Approve Manual Installation for single agent

Get-AgentPendingAction | where {$_.AgentName –match "*netbios_name*"} | Approve-AgentPendingAction

Approve Manual Installation for all pending agents

Get-AgentPendingAction | where {$_.AgentPendingActionType –match "ManualApproval"} | Approve-AgentPendingAction

Approve Manual Installation for *n* number of agents

```
$i = 1; foreach ($agent in Get-AgentPendingAction | where {$_.AgentPendingActionType –eq "ManualApproval"}) {if ($i -le n) {$agent | Approve-AgentPendingAction;$i++}}
```

***To approve updates to agents, you can use the same commands here. Just replace "ManualApproval" with "UpdateFailed". This is the Pending Action Type string to match for updates.*

Discover and Install agent (not a one-liner)

***This example will configure the agent to report to the Management Server you are currently connected to while performing this action.*

```
$query = New-LdapQueryDiscoveryCriteria -domain domain -ldapquery "(cn=target_netbios_name)"

$discoverycfg = New-WindowsDiscoveryConfiguration -ldapquery $query

$discoveryResults = Start-Discovery -managementServer (get-managementServer) -
windowsDiscoveryConfiguration $discoverycfg

Install-Agent -managementServer (get-ManagementServer) -agentManagedComputer
$discoveryResults.CustomMonitoringObjects
```

Get agent state (Windows Computer Instance)

```
get-agent | where {$_.computername -eq "netbios_name"} | ft name,HealthState
```

Get group members and contained instance state, by group name

```
foreach ($group in get-monitoringobjectGroup) {if($group.DisplayName -eq "group_name")
{$group.GetRelatedMonitoringObjects() | ft DisplayName,HealthState}}
```

Rule

Which Management Pack contains this Rule?

```
(get-rule | where {$_.displayname -eq "rule"}).getManagementPack() | ft DisplayName,Name –auto
```

Find collection rule writing event to OperationsManager and/or OperationsManagerDW database

```
Foreach ($rule in (get-rule | where {$_.category -eq "EventCollection" -and $_.enabled -eq "True"} |
foreach-object {$_.DataSourceCollection}))
  {
  if ($rule.get_Configuration() -match ">event_number<")
    {$rule.get_ParentElement().DisplayName}
  }
```

Monitor

Which Management Pack contains this Monitor?

```
(get-monitor | where {$_.displayname -eq "monitor"}).getManagementPack() | ft DisplayName,Name –auto
```

Alert

New alert count

```
get-alert | where {$_.resolutionState -eq 0} | measure-object
```

Open alert count (all resolution states, except new and closed)

```
$states = 2..254;get-alert | where {$states -contains $_.resolutionState} | measure-object
```

Closed alert count

```
get-alert | where {$_.resolutionState -eq 255} | measure-object
```

Alerts raised on specific date

```
get-alert | where {$_.TimeRaised.date -eq "mm/dd/yyyy"}
```

Alerts raised in date range

```
get-alert | where {$_.TimeRaised.date -ge "mm/dd/yyyy" -and $_.TimeRaised.date -le "mm/dd/yyyy"}
```

Alert count on specific date

```
get-alert | where {$_.TimeRaised.date -eq "mm/dd/yyyy"} | Measure-Object
```

Alert count by date

```
$array = @();foreach ($date in Get-Alert | foreach-object {$_.get_TimeRaised().toShortDateString()}) {$array += $date};$array | Group-Object | select-object count,name | sort-object name –desc
```

Top 10 alerts

```
get-alert | Group-Object Name | Sort-object Count -desc | select-Object -first 10 Count, Name | ft –auto
```

Last 10 critical alerts (not closed)

```
get-alert | where {$_.severity -eq "error" -and $_.resolutionstate -ne 255} | sort-object TimeRaised -desc | select-object -first 10 name,timeraised
```

Top 10 Repeat Count alerts (not closed)

```
get-alert | where {$_.RepeatCount -gt 0 -AND $_.resolutionState -ne 255} | sort-object RepeatCount -desc | select-object -first 10 repeatcount,name | ft –auto
```

Resolve all open alerts in date range

```
get-alert | where {$_.TimeRaised.date -ge "mm/dd/yyyy" -and $_.TimeRaised.date -le "mm/dd/yyyy" -and $_.resolutionState -ne 255} | resolve-alert
```

Resolve all open alerts, by Alert Name

```
get-alert | where {$_.Name -eq "alert_name" -AND $_.resolutionState -ne 255} | resolve-alert
```

Resolve all open alerts for specific Agent

```
get-alert | where {$_.netbiosComputerName -eq "netbios_name" -AND $_.resolutionState -ne 255} | Resolve-Alert
```

Resolve all alerts, by Resolution State

```
get-alert | where {$_.resolutionState -ne 255} | Resolve-Alert
```

Class

Get class properties, by class name

```
get-monitoringclass | where {$_.name -eq "class_name"} | foreach-object {$_.getMonitoringProperties()} | select-object name
```

Get HOST class, by class name

```
get-monitoringclass | where {$_.name -eq "class_name"} | foreach-object {$_.findHostClass()} | select-object DisplayName
```

Get HOST class properties, by class name (if any)

```
get-monitoringclass | where {$_.name -eq "class_name"} | foreach-object {$_.findHostClass().PropertyCollection} | ft name
```

Get BASE class, by class name

```
foreach ($base in Get-MonitoringClass | where {$_.name -eq "class_name"}) {get-monitoringclass | where {$_.id -eq $base.base.id} | select-object name}
```

Get BASE class properties, by class name (if any)

```
foreach ($base in Get-MonitoringClass | where {$_.name -eq "class_name"}) {get-monitoringclass | where {$_.id -eq $base.base.id} | foreach-object {$_.getMonitoringProperties()} | ft -auto parentElement,name}
```

***Check out my GetClassPath script to view the entire BASE and HOST class path (to System.Entity), and their properties, for a particular class.*

Event

Find collection rule writing event to OperationsManager and/or OperationsManagerDW database

```
Foreach ($rule in (get-rule | where {$_.category -eq "EventCollection" -and $_.enabled -eq "True"} | foreach-object {$_.DataSourceCollection}))
  {
  if ($rule.get_Configuration() -match ">event_number<")
    {$rule.get_ParentElement().DisplayName}
  }
```

***Regarding the next two command: Querying events tends to be quite resource intensive, given the sheer number of events OpsMgr collects. Even more so if performing foreach loops, sorting and grouping (like my first example).*

Event count, by date

```
$array = @();foreach ($date in Get-Event | foreach-object {$_.get_TimeGenerated().toShortDateString()}) {$array += $date};$array | Group-Object | select-object count,name | sort-object name –desc
```

Top 10 Events, by Event Number

```
get-event | Group-Object number | Sort-object Count -desc | select-Object -first 10 Count, Name | ft –auto
```

Override

All monitors overridden from MP

```
foreach ($monitor in Get-ManagementPack | where {$_.DisplayName -match "mp_display_name"} | get-override | where {$_.monitor}) {get-monitor | where {$_.Id -eq $monitor.monitor.id} | select-object DisplayName}
```

All rules overridden from MP

```
foreach ($rule in Get-ManagementPack | where {$_.DisplayName -match "mp_display_name"} | get-override | where {$_.rule}) {get-rule | where {$_.Id -eq $rule.rule.id} | select-object DisplayName}
```

Overrides created in date range

```
Get-ManagementPack | where {$_.sealed -eq $false} | get-override | where {$_.TimeAdded.date -ge "mm/dd/yyyy" -and $_.TimeAdded.date -le "mm/dd/yyyy"} | fl name,TimeAdded
```

Overrides that have been modified

```
Get-ManagementPack | where {$_.sealed -eq $false} | get-override | where {$_.LastModified -gt $_.TimeAdded} | fl name,LastModified
```

Overrides modified in date range

```
Get-ManagementPack | where {$_.sealed -eq $false} | get-override | where {$_.LastModified -gt $_.TimeAdded -and $_.LastModified.date -ge "mm/dd/yyyy" -and $_.LastModified.date -le "mm/dd/yyyy"} | select-object name,LastModified
```

All overrides

```
Get-ManagementPack | where {$_.sealed -eq $false} | get-override | select-object name,parameter,module,rule,value
```

Misc

***There is no way to differentiate a Console Session and a Command Shell session (AFAIK). These next two examples will show connections to SDK.*

List users connected to SDK

```
Get-ManagementGroupConnection | foreach-object {$_.ManagementGroup.getConnectedUserNames()}
```

Number of users connected to SDK

```
Get-ManagementGroupConnection | foreach-object
{$_.ManagementGroup.getConnectedUserNames()} | Measure-Object
```

Get agent state (Windows Computer Instance)

```
get-agent | where {$_.computername -eq "netbios_name"} | ft name,HealthState
```

Get group members and contained instance state, by group name

```
foreach ($group in get-monitoringobjectGroup) {if($group.DisplayName -eq "group_name")
{$group.GetRelatedMonitoringObjects() | ft DisplayName,HealthState}}
```

Find Agent, by Health Service Id

```
get-agent | where {$_.hostedHealthService.id -eq "guid"} | select-object name
```

Enable Agent Proxying, by Health Service Id

```
$a=get-agent | where {$_.hostedHealthService.id -eq
"guid"};$a.set_proxyingEnabled($true);$a.applyChanges()
```

Alert grooming (evaluate whether alert grooming is working. No results is good.)

```
$Threshold = (Get-Date).AddDays(-(get-defaultsetting)[42].Value).Date;Get-Alert | Where
{$_.TimeResolved -and $_.TimeResolved.Date -lt $Threshold} | Measure-Object
```

Get BaseManagedEntityId for Agent

```
get-agent | where {$_.ComputerName -eq "netbios_name"} | select {$_.HostedHealthService.id},name
```

Which hotfixes should I apply?

By Kevin Holman

In general - you should evaluate all hotfixes available, and only apply those applicable to your environment. However, some of these below I have seen impact almost **every** environment, and should be heavily considered.

This list is nothing official.... this is just a general list of the recommended hotfixes I end up proactively applying to most environments.... it is not a complete list of ALL hotfixes, and you may be affected by other issues.

This list ABSOLUTELY assumes you are at OpsMgr SP1-RTM level as a base (6.0.6278.0).

This is updated as of 2-5-2009

Hotfix	Update Files	Supersedes Hotfix(s)	Resolves	Targets	Comments
951979	Microsoft.SystemCenter.2007.mp 6.0.6278.19 Microsoft.SystemCenter.ACS.Internal.mp 6.0.6278.19 Microsoft.SystemCenter.Internal.mp 6.0.6278.19 Microsoft.Mom.BackwardCompatibility.mp 6.0.6278.19	none	Problem 1: CSDVersion property from Windows Vista-based computers Problem 2: operating system properties from Windows 2000 based computers. Problem 3: The HandleCountThreshold monitor doesn't restart the HealthService Problem 4: ACS events cannot be collected correctly. Problem 5: Alerts are raised for Performance Data Source Module Problem 6: Cluster discovery does not work correctly	RMS Console	This hotfix is unique in that it only contains for updated core management packs. **I recommend this hotfix for ALL environments.**
954903	Mommodules.dll 6.0.6278.36	KB951380 Mommodules.dll 6.0.6278.20 KB950853 Mommodules.dll 6.0.6278.11	The Monitoringhost.exe process may consume all the CPU resources when a large amount of performance data is created by using a managed data source module in System Center Operations Manager 2007	RMS MS GW Agents	**I recommend this hotfix for ALL environments.** (at a minimum 951380 should be deployed if not this hotfix, as 951380 covers a cluster discovery, and includes a memory leak issue.)
956240	Microsoft.mom.dataaccesslayer.dll 6.0.6278.37	none	The SQL Server process may consume lots of CPU resources on the server that hosts the Operations Manager 2007 database after you make Operations Manager 2007 configuration changes	RMS MS OpsDB	**I recommend this hotfix for ALL environments.** This is a ***CRITICAL*** patch, especially for larger environments, as the database performance is significantly impacted if you don't apply it.
957511	Momnetworkmodules.dll 6.0.6278.45	KB951526 Momnetworkmodules.dll 6.0.6278.24 KB956689	Warning alerts and warning monitor state changes do not occur as expected for a Web application that is created by using the Web application template in System Center Operations Manager 2007 Service	RMS MS GW Agents	**I recommend this hotfix for ALL environments.** **951526** is ***required*** if you use the current

		Momnetworkmodules.dll 6.0.6278.41	Pack 1		Dell Hardware MP, or other MP's which utilize SNMP heavily. However, you might as well implement the link above for 957511 since it includes 951526 and 956689.
958253	Microsoft.systemcenter.internal.mp 6.0.6278.55	none	When you try to view the **Patch List** property in Microsoft System Center 2007 Operations Manager, the list of Operations Manager agent hotfixes may be truncated. Therefore, you cannot determine which hotfixes are already installed on agents and which hotfixes must be installed	RMS Console	**I recommend this hotfix for ALL environments.**
954049	Multiple - See article	none	Server 2008 support	RMS MS GW Agents	**I recommend this hotfix for ALL environments which will have Server 2008 as a server role. Also, for environments that will have Server 2003 agents upgraded to 2008.**
957135	Microsoft.mom.ui.common.dll 6.0.6278.46 Microsoft.mom.ui.components.dll 6.0.6278.46	none	If you set an alert resolution to ***closed***, the resolution state of closed is displayed if the Resolution State column is visible in the view. ***However***, the alert resolution state remains in its original state if one of the following is true: The view is refreshed. A new view or a new node is selected, and then the original alert view is revisited. This is true if one or more alerts that are raised by agents contain a created date or a created time that is in the future.	All Consoles	**I recommend this hotfix only if you are impacted with this issue.** I have seen several large environments, with large agent counts, and large alerts counts, that required it.
956423	Microsoft.enterprisemanagement.operationsmanager.dll 6.0.6278.40	none	Reconnecting a management group with Tiering scenario sets incorrect MG	All Consoles	**Only required if using connected management groups.**
954643	Managementpackinstall.s	none	Event ID 31569 is logged after you install a management pack that	Data Warehouse	This hotfix includes a SQL script, which you

	p.sql		includes reports on a System Center Operations Manager 2007 SP1 server	Database	execute on the database in a query window. **I recommend this hotfix only if you are impacted with these events.**
956446	Microsoft .enterprisemanagement .healthservice .modules.notification.dll 6.0.6278.39	none	The email notifications may contain garbled subject lines if you enable the "Generate subject line with no encoding" option in System Center Operations Manager 2007 Service Pack 1	RMS	Apply this hotfix if sending to a phone or pager and the subject is garbled **I recommend this hotfix only if you are impacted with this issue.**

Make sure you see these additional posts on the subject of hotfixes:

http://blogs.technet.com/kevinholman/archive/2008/06/25/a-little-tidbit-on-hot-fixes-for-opsmgr.aspx

http://blogs.technet.com/kevinholman/archive/2008/06/24/how-do-i-know-which-hotfixes-have-been-applied-to-which-agents.aspx

http://blogs.technet.com/kevinholman/archive/2008/06/27/a-report-to-show-all-agents-missing-a-specific-hotfix.aspx

Several of these updates require agent updates as well, so be prepared to deal with those.

ALWAYS make sure you read the instructions to understand if the hotfix is a SQL update, installed to the RMS, MS, and/or Gateway, AND/OR applies to agents as well.

ALWAYS make sure you double-check the DLL version of the updated files to make sure the hotfix successfully applied after installing.

ALWAYS make sure you double-check the \AgentManagement directory of the management servers and gateways, to make sure if there is an agent update, the MSP was copied over correctly.

ALWAYS, on Server 2008 OS, run the hotfix from an elevated command prompt window.

Console based Agent Deployment Troubleshooting table

By Kevin Holman

This post is a list of common agent push deployment errors… and some possible remediation options.

Most common errors while pushing an agent:

Error	Error Code(s)	Remediation Steps
The MOM Server could not execute WMI Query "Select * from Win32_Environment where NAME='PROCESSOR_ARCHITECTURE'" on computer server.domain.com Operation: Agent Install Install account: domain\account Error Code: 80004005 Error Description: Unspecified error	80004005	1. Check the PATH environment variable. If the PATH statement is very long, due to lots of installed third party software - this can fail. Reduce the path by converting any long filename destinations to 8.3, and remove any path statements that are not necessary. 2. The cause could be corrupted Performance Counters on the target Agent. To rebuild all Performance counters including extensible and third party counters in Windows Server 2003, type the following commands at a command prompt. Press ENTER after each command. cd \windows\system32 lodctr /R Note /R is uppercase. Windows Server 2003 rebuilds all the counters because it reads all the .ini files in the C:\Windows\inf\009 folder for the English operating system. How to manually rebuild Performance Counter Library values http://support.microsoft.com/kb/300956 3. Manual agent install.
The MOM Server could not execute WMI Query "Select * from Win32_OperatingSystem" on computer “servername.domain.com” Operation: Agent Install Install account: DOMAIN\account Error Code: 800706BA Error Description: The RPC server is unavailable.	8004100A 800706BA	1. Ensure agent push account has local admin rights 2. Firewall is blocking NetBIOS access 3. Inspect WMI health and rebuild repository if necessary 4. Firewall is blocking ICMP (Live OneCare) 5. DNS incorrect
The MOM Server failed to open service control manager on computer "servername.domain.com". Access is Denied	80070005	1. Verify SCOM agent push account is in Local

Operation: Agent Install Install account: DomainName\User Account Error Code: 80070005 Error Description: Access is denied.	80041002	Administrators group on target computer. 2. On Domain controllers will have to work with AD team to install agent manually if agent push account is not a domain admin. 3. Disable McAfee antivirus during push
The MOM Server failed to open service control manager on computer "servername.domain.com". Therefore, the MOM Server cannot complete configuration of agent on the computer. Operation: Agent Install Install account: DOMAIN\account Error Code: 800706BA Error Description: The RPC server is unavailable.	800706BA	1. Firewall blocking NetBIOS ports 2. DNS resolution issue. Make sure the agent can ping the MS by NetBIOS and FQDN. Make sure the MS can ping the agent by NetBIOS and FQDN 3. Firewall blocking ICMP 4. RPC services stopped.
The MOM Server failed to acquire lock to remote computer servername.domain.com. This means there is already an agent management operation proceeding on this computer, please retry the Push Agent operation after some time. Operation: Agent Install Install account: DOMAIN\account Error Code: 80072971 Error description: Unknown error 0x80072971	80072971	This problem occurs if the LockFileTime.txt file is located in the following folder on the remote computer: %windir%\422C3AB1-32E0-4411-BF66-A84FEEFCC8E2 When you install or remove a management agent, the Operations Manager 2007 management server copies temporary files to the remote computer. One of these files is named LockFileTime.txt. This lock file is intended to prevent another management server from performing a management agent installation at the same time as the current installation. If the management agent installation is unsuccessful and if the management server loses connectivity with the remote computer, the temporary files may not be removed. Therefore, the LockFileTime.txt may remain in the folder on the remote computer. When the management server next tries to perform an agent installation, the management server detects the lock file. Therefore, the management agent installation is unsuccessful. http://support.microsoft.com/kb/934760/en-us
The MOM Server detected that the following services on computer "(null);NetLogon" are not running. These services are required for push agent installation. To complete this operation, either start the required services on the computer or install the MOM agent manually by using MOMAgent.msi located on the product CD. Operation: Agent Install Remote Computer Name: servername.domain.com Install account: DOMAIN\account Error Code: C000296E Error Description: Unknown error 0xC000296E	C000296E	1. Netlogon service is not running. It must be set to auto/started

The MOM Server detected that the following services on computer "winmgmt;(null)" are not running	C000296E	1. WMI services not running or WMI corrupt
The MOM Server detected that the Windows Installer service (MSIServer) is disabled on computer "servername.domain.com". This service is required for push agent installation. To complete this operation on the computer, either set the MSIServer startup type to "Manual" or "Automatic", or install the MOM agent manually by using MOMAgent.msi located on the product CD. Operation: Agent Install Install account: DOMAIN\account Error Code: C0002976 Error Description: Unknown error 0xC0002976	C0002976	1. Windows Installer service is not running or set to disabled – set this to manual or auto and start it.
The Agent Management Operation Agent Install failed for remote computer servername.domain.com. Install account: DOMAIN\account Error Code: 80070643 Error Description: Fatal error during installation. Microsoft Installer Error Description: For more information, see Windows Installer log file "C:\Program Files\System Center Operations Manager 2007\AgentManagement\AgentLogs\servernameAgentInstall.LOG C:\Program Files\System Center Operations Manager 2007\AgentManagement\AgentLogs\servernameMOMAgentMgmt.log" on the Management Server.	80070643	1. Enable the automatic Updates service…. Install the agent – then disable the auto-updates service if desired.
Call was canceled by the message filter	80010002	Install latest SP and retry. One server that failed did not have Service pack installed
The MOM Server could not find directory \\I.P.\C$\WINDOWS\. Agent will not be installed on computer "name". Please verify the required share exists.	80070006	1. Manual agent install Possible locking on registry? http://www.sysadmintales.com/category/operations-manager/ Try manual install. Verified share does not exist.
The network path was not found.	80070035	1. Manual agent install
The Agent Management Operation Agent Install failed for remote computer "name". There is not enough space on the disk.	80070070	1. Free space on install disk
The MOM Server failed to perform specified operation on computer "name". The semaphore timeout period has expired.	80070079	NSlookup failed on server. Possible DNS resolution issue. Try adding dnsname to dnssuffix search list.
The MOM Server could not start the MOMAgentInstaller service on computer "name" in the time.	8007041D 80070102	NSlookup failed on server. Possible DNS resolution issue. Verify domain is in suffix search list on management servers.

	2	
The Agent Management Operation Agent Install failed for remote computer "name"	80070643	1. Ensure automatic updates service is started 2. Rebuild WMI repository 3. DNS resolution issue
The Agent Management Operation Agent Install failed for remote computer "name". Another installation is already in progress.	80070652	Verify not in pending management. If yes, remove and then attempt installation again.
The MOM Server detected that computer "name" has an unsupported operating system or service pack version	80072977	Install latest SP and verify you are installing to Windows system.
Not discovered		Agent machine is not a member of domain
Ping fails		1. Server is down 2. Server is blocked by firewall 3. DNS resolving to wrong IP.
Fail to resolve machine		1. DNS issue
The MOM Server failed to perform specified operation on computer "name". Not enough server storage…	8007046A	1. This is typically a memory error caused by the remote OS that the agent is being installed on.
There are currently no logon servers available to service the logon request.	8007051F	1. Possible DNS issue
This installation package cannot be installed by the Windows Installer service. You must install a Windows service pack that contains a newer version of the Windows Installer service.	8007064D	1. Install Windows Installer 3.1
The network address is invalid	800706AB	Possible DNS name resolution issue. Tried nslookup on server name and did not get response. Verify domain is in suffix search list on management servers.
The MOM Server failed to perform specified operation on computer servername.domain.com	80070040	1. Ensure agent push account has local admin rights
The MOM Server detected that the actual NetBIOS name SERVERNAME is not same as the given NetBIOS name provide for remote computer SERVERNAME.domain.com.	80072979	1. Correct DNS/WINS issue. 2. Try pushing to NetBIOS name