

SCOM2K7 Windows Disk Monitoring

Administrator Guide · Management pack version 1.0.0.0 · scom2k7.com

Management pack ID	SCOM2K7.Windows.Disk.Monitoring
Version	1.0.0.0 (sealed, key token ccc0c7331efac0c1)
Companion pack	SCOM2K7.Windows.Disk.Monitoring.Overrides 1.0.0.0 (unsealed, optional)
Publisher	scom2k7.com, the people behind SCOM Pulse
Applies to	System Center Operations Manager 2016, 2019, 2022 and later; Windows Server logical disks
Audience	SCOM administrators and operators
Document date	7 October 2026

Everything in this guide that is stated as a fact was verified either with the Operations Manager SDK or in a lab management group (SCOM 2022, Windows Server 2022 agents). Where something was not tested, the guide says so.

Contents

- 1 What this pack does
- 2 What is in the pack
- 3 How the monitors work
- 4 Alerts and notifications
- 5 Health Explorer: free space details
- 6 The "largest folders and files" task
- 7 Tuning with overrides
- 8 Installation and upgrade
- 9 Known limitations
- 10 Troubleshooting
- 11 Comparison with the built-in Microsoft monitor
- 12 Version history and test summary
- A Appendix: element reference

1. What this pack does

The pack watches free space on every Windows Server logical disk that SCOM has discovered and raises two kinds of alert:

- **Warning** when a disk has been below **10 % free** for two consecutive samples.
- **Critical** when a disk has been below **5 % free** for two consecutive samples.

That is the whole policy. There is one number per monitor, it is a percentage, and it can be changed per disk, per group of disks, or for everything. The explanation an operator can give an end user is one sentence:

"The warning alert fires at 10 % free, the critical at 5 %, and each stays open until the disk is back above its own line."

The pack replaces Microsoft's built-in *Logical Disk Free Space* monitor, which uses eight numbers (percent and megabytes, for system and non-system drives, at two severities), requires both percent and megabytes to be breached, and by default only alerts on Critical. See section 11 for the comparison.

Design principles

- **Health is decided by the performance counter alone.** No script sits in the path between the counter and the alert, so a slow or failing script can never suppress an alert. Scripts are used only to add detail after the fact.
- **Two separate monitors, not one three-state monitor.** SCOM does not raise a second alert while a monitor's first alert is open, so a single monitor that alerts on Warning would never produce a new

Critical alert for a notification subscription to pick up. Two monitors give two alerts.

- **Recovery needs a real reading.** A monitor returns to healthy only when an actual sample is at or above its threshold. An agent restart cannot clear a still-low disk (this was tested; see section 12).
- **Every knob has a name and a description** in the override dialog, and every alert says which threshold it crossed and where to change it.

2. What is in the pack

Element	Purpose
SCOM2K7: Logical Disk Free Space - Warning	Unit monitor. Warning state and Warning alert (priority Normal) below 10 % free. Rolls up to Availability.
SCOM2K7: Logical Disk Free Space - Critical	Unit monitor. Critical state and Critical alert (priority High) below 5 % free. Rolls up to Availability.
Free space details (two diagnostics)	Run automatically when a monitor goes unhealthy. One WMI query; writes free GB, total GB, volume label and current percent into Health Explorer.
SCOM2K7: List largest folders and files on this disk	Console task, on demand only. Bounded scan of the volume that lists the biggest folders and files.
Knowledge articles	On both monitors: summary, what the alert shows, causes, resolutions, how to tune.
Companion pack (optional, unsealed)	A dynamic group "SCOM2K7 - Large volumes (1 TB and over)" with lower thresholds: 3 % warning, 1.5 % critical. Edit or delete freely.

Target class for everything: `Microsoft.Windows.Server.LogicalDisk` (drive letters and mount points on Windows Server). Cluster shared volumes and cluster disks are a different class and are not covered.

3. How the monitors work

Sampling and thresholds

Each monitor samples the Windows performance counter `LogicalDisk \ % Free Space` for its disk every **300 seconds**. When **2 consecutive** samples are below the threshold, the monitor changes state and raises its alert. The first sample at or above the threshold returns it to healthy and auto-resolves the alert.

Parameter (as shown in the override dialog)	Warning	Critical
Free space threshold (% free)	10	5
Sample interval (seconds)	300	300
Consecutive samples before alert	2	2

How long it takes

Two samples five minutes apart means 5 to 10 minutes of sampling. On top of that, **Windows refreshes the disk free-space performance counters only every few minutes**. In the lab, every state change arrived 2 to 4.5 minutes after the disk actually changed. Plan on:

- **7 to 15 minutes** from "disk ran out of space" to the alert, at default settings.
- **About 2 to 10 minutes** from "space was freed" to the alert auto-resolving.

Shortening the sample interval below 300 seconds buys little, because the counter refresh, not the interval, dominates. Microsoft's built-in monitor reads the file system directly through a script and does not have this lag; that is the one place where it reacts faster. This pack traded that for an alert that no script can block.

What happens on agent restart

Monitors keep their state across a Health Service restart. Because recovery requires an actual reading at or above the threshold, a disk that is still low stays unhealthy after the restart and its alert stays open. This differs from Microsoft's stock "consecutive samples" pattern, where the first sample after a restart can clear a still-low disk.

Recalculate Health

"Recalculate Health" in Health Explorer does nothing for these monitors. There is no safe single-reading answer for a consecutive-sample monitor, so none is defined (the same is true of Microsoft's own consecutive-sample monitor type). The next scheduled sample re-evaluates.

4. Alerts and notifications

Each monitor raises its own alert and resolves it itself. A disk at 4 % free therefore has two open alerts: a Warning (it is below 10 %) and a Critical (it is below 5 %). When space is freed to 7 %, the Critical resolves and the Warning stays open until the disk passes 10 % again.

Logical disk free space is critically low Severity: Critical Priority: High

Disk C: on DB01.contoso.com dropped below this monitor's free-space critical threshold.

Sampled free space: 4.38708877563477 %.

Free and total GB, measured after the state change when the diagnostic succeeds, are in Health Explorer under State Change Events ("Free space details").

To see or change the threshold in effect for this disk, open Overrides Summary on this monitor.

The percentage is the exact counter sample that crossed the threshold (SCOM alert parameters cannot round it). The alert deliberately does not contain the threshold value or the disk size: the threshold may be overridden, and the disk size that SCOM discovers is a raw byte count. Both are one click away: the threshold in Overrides Summary, the GB values in Health Explorer.

Subscriptions

Because Warning and Critical are different alerts from different monitors, a subscription on Critical severity always sees a new alert when a disk crosses 5 %, even if a Warning alert for the same disk is already open. Test your subscriptions once after import; delivery is still decided by subscription configuration.

5. Health Explorer: free space details

When either monitor goes unhealthy, its "Free space details" diagnostic runs on the agent: one filtered query of `Win32_Volume` for that volume (which handles drive letters and mount points alike). The result appears in Health Explorer under the monitor's **State Change Events**, on the event that changed state:

Reading taken	2026-10-06T20:12:38
Volume label	(blank on an unlabelled disk)
Free GB	3.6
Total GB	79.3
Percent free now	4.5
Note	Measured after the state change. The alert shows the sample that tripped the threshold.

If the query fails (for example the volume cannot be found), the diagnostic reports an **Error** line instead. If it times out (30 seconds) it reports nothing. In either case the alert has already been raised; the diagnostic only adds detail.

6. The "largest folders and files" task

Select a logical disk in any state view and run **SCOM2K7: List largest folders and files on this disk**. It runs on the agent, walks the whole volume, and returns a text report:

Largest items on C:\
Scanned 297592 entries in 4.1 s (engine: C#, 8 threads). Report depth 3, time budget 120 s.

Top-level folders by size (share of 58.7 GB used on C:\):

29.2 GB	50%	C:\Windows
22.7 GB	39%	C:\Backups
3.2 GB	5%	C:\Program Files (x86)
1.7 GB	3%	C:\ProgramData
1.6 GB	3%	C:\Program Files
0.3 GB	1%	(files directly in C:\, plus smaller folders)

Inside the biggest top-level folders (indented = inside the folder above; each total includes everything beneath it; a subfolder is listed when it holds at least 5% of its parent and 1% of the volume):

29.2 GB	50%	C:\Windows
12.4 GB	21%	WinSxS
5.3 GB	9%	SoftwareDistribution
5.3 GB	9%	Download
4.7 GB	8%	System32
6.8 GB	12%	(files directly here, plus smaller folders)
22.7 GB	39%	C:\Backups
22.7 GB	39%	AppServer
3.2 GB	5%	C:\Program Files (x86)
3.1 GB	5%	Microsoft
1.4 GB	2%	EdgeCore

Top 15 files by size:

11.3 GB	C:\Backups\AppDataServer\AppDataServer-2026-10-05.bak
---------	---

...

Reparsing points (junctions / mount points) skipped, NOT scanned: 152 (first 10 shown)

Three sections. The first ranks the folders directly under the root by size with their share of used space; every line is a different folder, so there is nothing to misread. The second drills into the big ones only (at least 5 % of used space): indentation shows which folder a line sits in, each total includes everything beneath it, a subfolder appears when it holds at least 5 % of its parent and 1 % of the volume, and a remainder line accounts for what is not listed. Here Backups holds 22.7 GB, all of it inside AppServer. The third lists the largest files with full paths.

Totals are exact: every file on the volume is counted and attributed to its ancestor at the report depth. The walk runs in compiled .NET code inside the PowerShell script on several threads; a 300,000-entry server system drive takes roughly 3 to 10 seconds, a 770,000-entry workstation drive 4 seconds. The task is still bounded, because an unbounded scan is the wrong thing to start on a server whose disk is already struggling:

Parameter	Default	What it does
Folder depth to report	3	How many levels of folders appear in the report. Files below that depth are still counted, in their ancestor's total.
Time budget (seconds)	120	Checked every 256 entries. When reached, the scan stops and returns what it has, marked PARTIAL with the last path reached. Totals are then incomplete.
Number of folders per level, and files, to list	15	Maximum child folders shown under each folder in the tree, and the number of largest files listed. Bounds the output only, not the scanning work.
Script timeout (seconds)	180	Hard limit. If reached, the script is terminated and nothing is returned. Keep it above the time budget.

Junctions and mount points are listed but never followed, so the scan stays on the selected volume. `System Volume Information`, the page file, hibernation file and swap file are skipped. Folders that cannot be read are counted and reported. If the .NET compile step is unavailable on a host, the script falls back to a slower PowerShell walk with the same semantics and says so in the first line of the report.

7. Tuning with overrides

The three parameters

Both monitors expose the same three parameters, with these names in the override dialog:

Display name	Internal ID	Description shown in the dialog
Free space threshold (% free)	ThresholdPercent	Alert when % free space is below this value. This is the only threshold.
Sample interval (seconds)	IntervalSeconds	How often the counter is sampled. Default 300 (5 minutes).
Consecutive samples before alert	SamplesToTrigger	The breach must hold for this many samples in a row. Time to alert is roughly interval x samples, plus the few minutes Windows takes to refresh the disk counters. Default 2.

Where to override

- **One disk:** Overrides > Override the Monitor > For a specific object of class: Windows Server Logical Disk.
- **A set of disks:** create a group of logical disks and override "For a group". Groups can be built on any discovered property, for example Name equals `c:`, or on the computer that hosts the disk.
- **Everything:** "For all objects of class: Windows Server Logical Disk".

Precedence: an override on a single disk beats a group override, which beats the default. The effective value for any disk is shown in **Overrides Summary** on the monitor.

Very large disks: the companion pack

A percent-only rule is a policy choice, and on very large volumes 5 % can be hundreds of gigabytes. The optional companion pack `SCOM2K7.Windows.Disk.Monitoring.Overrides` handles this without adding a second threshold type:

- A dynamic group **SCOM2K7 - Large volumes (1 TB and over)**. Membership is property based: the discovered `Size` of the disk, which SCOM stores as a byte count, has 13 or more digits.
- Two overrides on that group: Warning at **3 %**, Critical at **1.5 %**.

The pack is unsealed on purpose. Change the percentages, change the size rule, or delete the pack; nothing else depends on it. A per-disk override still beats the group override.

Why not a megabyte threshold or a "system drive" rule? Both reintroduce the "why didn't I get an alert?" conversation this pack exists to end. And on the Windows Server 2016+ management pack the discovered `SizeNumeric` property is -1, so numeric size comparisons are not available anyway. If you need a tighter rule for OS drives, create a group of disks named `c:` and override that group.

8. Installation and upgrade

Requirements

Referenced pack	Minimum version
Microsoft.Windows.Library	7.5.8501.0
Microsoft.Windows.Server.Library	6.0.6957.0
System.Health.Library	7.0.8433.0
System.Library	7.5.8501.0
System.Performance.Library	7.0.8433.0
Companion pack only: Microsoft.SystemCenter.Library, Microsoft.SystemCenter.InstanceGroup.Library	7.0.8433.0 / 7.5.8501.0

Agents need Windows PowerShell (any supported Windows Server version) for the diagnostics and the task. The monitors themselves need nothing beyond the performance counter.

New installation

1. **Disable Microsoft's monitor first** or you will get two sets of alerts per disk. In the Windows Server 2016 and above pack it is *Logical Disk Free Space Monitor* (`Microsoft.Windows.Server.10.0.LogicalDisk.FreeSpace`); older OS packs have an equivalent. Override "Enabled" to False for all objects of the relevant logical disk class, in your own override pack.
2. Import `SCOM2K7.Windows.Disk.Monitoring.mp` (sealed, 1.0.0.0).
3. Optionally import `SCOM2K7.Windows.Disk.Monitoring.Overrides.xml`.
4. Within a few minutes the two monitors appear on every Windows Server logical disk and initialise to healthy. Confirm in a state view, and test one notification subscription.

If a pre-release build (**Custom.Windows.Disk.Monitoring 3.0.0.x**) is installed

The public pack has a new ID, so it installs alongside rather than over a 3.0.0.x build. Delete the 3.0.0.x pack first (it has no dependents unless you created override packs for it), then import 1.0.0.0. Re-create any overrides against the new monitor names; the parameter names are unchanged.

Upgrading from 2.0.0.0

The 2.0.0.0 pack was sealed with a key that is no longer available (token `8dffabdd0a079bc8`), so SCOM will not accept 3.x as an upgrade of it. The procedure is delete-and-reimport:

1. Find packs that reference the old one. SCOM refuses to delete a pack that is referenced:

```
Get-SCOMManagementPack | Where-Object { $_.References.Values.Name -contains  
'SCOM2K7.Windows.Disk.Monitoring' } |  
Select-Object Name, Version, Sealed
```

For each unsealed hit, export it, remove the <Reference> to `SCOM2K7.Windows.Disk.Monitoring` (and anything that uses it), and re-import it.

2. Export any override packs that hold overrides for the old monitors, so you can re-create the values afterwards. The old monitor IDs were `Custom.Warning.DiskSpace.PercentFreeSpace.Perf.ThenScript.Monitor` and `Custom.Critical.DiskSpace.PercentFreeSpace.Perf.ThenScript.Monitor`; the old parameter names were `Threshold`, `Frequency`, `NumSamples`.
3. Delete `SCOM2K7.Windows.Disk.Monitoring 2.0.0.0`.
4. Import the sealed 1.0.0.0, then the optional companion pack.
5. Re-create overrides against the new monitors and parameter names (section 7).

Alert and state history of the old monitors is not carried over.

9. Known limitations

- **Counter refresh lag.** Windows updates the logical-disk free-space counters only every few minutes. Expect 7 to 15 minutes to alert at default settings (section 3). Measured in the lab; it is a property of the Windows performance provider, not of this pack.
- **Percent only.** No megabyte floor, by design. Use the companion pack or your own group overrides for very large or very small disks.
- **No Recalculate Health.** See section 3.
- **Alert percentage is unrounded** (for example 4.38708877563477 %). SCOM alert parameters cannot format numbers; the diagnostic shows a rounded value.
- **No disk size in the alert.** The discovered size is a raw byte count; it is shown in GB by the diagnostic instead.
- **Windows Server only.** Client operating systems, cluster shared volumes and cluster disks are not targeted.
- **Mount points were not exercised in the lab** (the lab had none). The diagnostic was unit-tested for the mount-point path format; the monitors use the performance counter instance name that SCOM discovered.

10. Troubleshooting

Symptom	What to check
No alert although the disk is low	Allow 15 minutes at default settings. Then check Overrides Summary on the monitor for that disk (threshold, interval, samples, enabled). Confirm the disk is a discovered Windows Server Logical Disk instance (state view). Confirm the agent is healthy and LogicalDisk(X:)\% Free Space exists in Performance Monitor on the server.
Two sets of disk alerts per disk	Microsoft's Logical Disk Free Space monitor is still enabled. Disable it by override (section 8).
Alert resolved and re-raised repeatedly	Free space is oscillating around the threshold. Raise "Consecutive samples before alert" for that disk, or move the threshold.
Alert raised but Health Explorer shows no "Free space details"	The diagnostic failed or timed out on the agent. Look at the state change event for an Error line, and at the agent's Operations Manager event log for PowerShell errors. The alert itself is unaffected.
Task result says PARTIAL	The time budget was reached before the whole volume was walked, so totals are incomplete. Raise the time budget and keep the script timeout above it. The top files list is correct for what was scanned.
Task report says "engine: PowerShell fallback"	The in-script .NET compile was not possible on that agent. The result is still correct, only slower. Check the agent's Operations Manager log for the reason given on the first line.
Task fails with no output	The script timeout was reached before the time budget. Set the budget at least 30 seconds below the timeout.
Large volumes group is empty	Only disks whose discovered Size has 13 or more digits (1,000,000,000,000 bytes and up) are members. Check the Size property of the disk in a state view with that column added.
Import of 3.x refused as an upgrade of 2.0.0.0	Expected: different signing key. Follow the delete-and-reimport procedure in section 8.

11. Comparison with the built-in Microsoft monitor

	Microsoft: Logical Disk Free Space Monitor	This pack
Thresholds	Percent AND megabytes, both must be breached; separate defaults for system and non-system drives (8 numbers)	One percent per monitor
Alerts	One three-state monitor; alerts on Critical only by default; severity follows state	Separate Warning and Critical alerts
Health decision	Script	Performance counter; no script in the path
Reaction time	One script interval; reads the file system directly, no counter lag	7 to 15 minutes at defaults, including counter refresh lag
Large and small disks	Handled by the megabyte floor	Optional "Large volumes" group; per-disk or group overrides otherwise
Detail after the alert	Percent and megabytes in the alert	GB details in Health Explorer; largest folders and files on demand
Override experience	Many parameters	Three named, described parameters

12. Version history and test summary

Version	Notes
2.0.0.0 (2017)	Two monitors on the "performance counter, then script" pattern, 20-minute samples, 3 consecutive samples. The alert depended on the enrichment script succeeding; the Warning script did not handle mount points; no parameter descriptions or knowledge.
3.0.0.x (2026, pre-release, ID Custom.Windows.Disk.Monitoring)	Rebuilt and lab-tested. Script removed from the health decision; 5-minute samples, 2 consecutive; restart-safe recovery; GB details by diagnostic; named parameters; knowledge articles; companion pack. The "largest folders and files" task went through four iterations: compiled .NET walk of the whole volume with exact totals, then a tree report, then a multi-threaded walk with a ranked top-level list followed by a drill-down.
1.0.0.0 (2026, first public release, ID SCOM2K7.Windows.Disk.Monitoring)	Same content as the last pre-release build, under the SCOM2K7 name, with an "About this pack" section in the knowledge articles.

What was tested (lab: SCOM 2022, three Windows Server 2022 machines, 6 October 2026)

Test	Result
SDK verification of the sealed pack and the companion pack	Pass
Warning and Critical trip after two consecutive low samples	Pass
Separate alerts, correct severities and priorities, correct alert text	Pass
Diagnostics run on both transitions with correct values	Pass
Recovery on the first good sample; alerts auto-resolve	Pass
Health Service restart with both monitors unhealthy: states and alerts held	Pass
Workflow reload (pack re-import) with a monitor unhealthy: state held	Pass
Task on a real server (107,860 entries, 13 seconds, junctions skipped)	Pass
Task guard rails (10-second budget, junction to another volume)	Pass
Large volumes group rule evaluates correctly	Pass
Counter refresh lag	Measured: 2 to 4.5 minutes
Mount points	Not tested

Appendix A. Element reference

ID	Type	Notes
SCOM2K7.Windows.Disk.Monitoring	Management pack	Sealed, 1.0.0.0
SCOM2K7.Windows.Disk.FreeSpace.MonitorType	Unit monitor type (Internal)	Config: InstanceName, IntervalSeconds, ThresholdPercent, SamplesToTrigger
SCOM2K7.Windows.Disk.FreeSpace.Warning.Monitor	Unit monitor	10 %, Warning, priority Normal, parent Availability
SCOM2K7.Windows.Disk.FreeSpace.Critical.Monitor	Unit monitor	5 %, Error, priority High, parent Availability
SCOM2K7.Windows.Disk.FreeSpace.Details.Warning.Diagnostic	Diagnostic	Runs on Warning; 30 s timeout
SCOM2K7.Windows.Disk.FreeSpace.Details.Critical.Diagnostic	Diagnostic	Runs on Error; 30 s timeout
SCOM2K7.Windows.Disk.LargestItems.ProbeAction	Probe action module type (Internal)	Overridable: Depth, TimeBudgetSeconds, TopN, TimeoutSeconds
SCOM2K7.Windows.Disk.LargestItems.Task	Console task	Target: Windows Server Logical Disk
SCOM2K7.Windows.Disk.Monitoring.Overrides	Management pack (unsealed, optional)	References the sealed pack by token ccc0c7331efac0c1
SCOM2K7.Windows.Disk.LargeVolumes.Group	Instance group (companion pack)	Size matches ^[0-9]{13,}\$

Monitor states

State ID	Display name	Health
FreeSpaceOK	Free space above threshold	Healthy
FreeSpaceLow	Free space below threshold	Warning or Critical, by monitor

Detection logic (for authors)

```
FreeSpaceLow: System.Performance.DataProvider (% Free Space, every IntervalSeconds)
               -> System.Performance.ConsecutiveSamplesCondition (Threshold = ThresholdPercent,
Direction = Less)
               -> ExpressionFilter: consecutive count >= SamplesToTrigger

FreeSpaceOK:   System.Performance.DataProvider (same, cooked down)
               -> ExpressionFilter: Value >= ThresholdPercent      (one actual reading; independent
of the count)
```

Alert parameters: {0} = `$Data/Context/InstanceName$`, {1} = host computer principal name, {2} = `$Data/Context/SampleValue$`. Only values that can never be empty are used, because SCOM drops an empty alert parameter and shifts the numbering of the rest.

About SCOM2K7 and SCOM Pulse

SCOM2K7 has been building SCOM tools since 2007. SCOM2K7 Windows Disk Monitoring is free, and so is the help on it: documentation, updates and the blog are at **scom2k7.com**.

Most of our time goes into **SCOM Pulse: your everyday SCOM console, in your browser**. Investigate alerts, explore health, deploy and manage Windows agents, and schedule maintenance from one web console. It works with your existing SCOM environment, without changing how your agents report, and puts no additional software on monitored computers.

- See why an alert fired, read the management pack knowledge, and explore the affected object's health on one page.
- Acknowledge, assign and comment; put an object into maintenance mode, create an override, or run a task, from the alert page.
- Deploy, repair and reassign Windows agents from your browser, with live progress per computer.
- Read-only shareable dashboard links for the people who only need to look something up, scoped by the SCOM user roles they already have.

One signed MSI on one Windows server. Supported with Operations Manager 2016 UR7 and later, 2019, 2022 and 2025. **The 30-day trial needs no key**: download it at **scom2k7.com**, install, and it starts on its own.

SCOM2K7 also makes the SCOM Maintenance Mode Scheduler, Alert Update Connector Pro and ServiceNow Connector Pro.