How do I use the CMDB health dashboard?

Questions addressed:

Page 1:

- What is the CMDB health dashboard?

- Why use the CMDB health dashboard?

Page 2:

- What metrics does the dashboard track to assess CMDB health?

- How do I use the dashboard to improve CMDB health?

Page 3:

- What are the best practices for setting up the health dashboard?

The Configuration Management Database (CMDB) health dashboard aggregates key performance metrics and offers actionable insights that enable you to monitor and improve the performance and health of your CMDB.

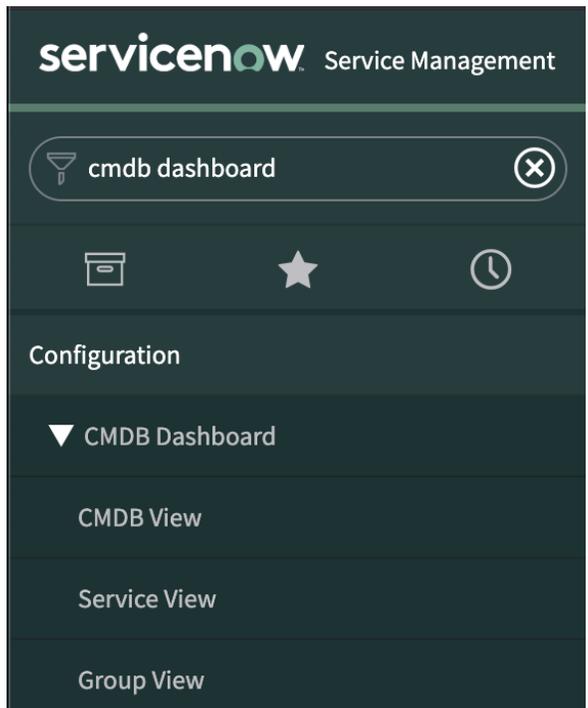**Why use the CMDB health dashboard?**

- **CMDB health is critical to success –** A well-configured CMDB can save you as much as 40% in IT costs and is critical to avoiding unplanned outages. Data quality in the CMDB is an essential goal of any IT organization.

- **Proactive issue management –** The dashboard enables you to proactively manage your CMDB/configuration items (CI) and identify issues, improve the CMDB's accuracy and usability, and act upon any issues before they become more severe risks to your operational environment.

- **Stakeholder transparency –** The dashboard makes the overall health of your CMDB visible to key stakeholders, enabling you to share status updates, track SLAs, and keep everyone aligned on key issues or success.

Related resources
- CMDB and Discovery deployment
- Populate and maintain your CMDB with Discovery
- Success Checklist – Plan your architecture, instances, integrations, and data flows
- Success Checklist – Discover and map your service assets
- Community Blog – Determine CMDB Health with the CMDB Dashboard
- CMDB Overview
- Ask the Expert – CMDB Basics Webinar

You can find the dashboard by searching for "CMDB Dashboard" in the application menu.

servicenow. Service Management

cmdb dashboard

Configuration

▼ CMDB Dashboard

CMDB View

Service View

Group View

If you have any questions on this topic or you would like to be a contributor to future ServiceNow best practice content, please contact us.

How do I use the CMDB health dashboard? (Continued)

What metrics does the dashboard track to assess CMDB health?

KPI category	Description	Question answered	Examples
Completeness	Test for required and recommended fields that are not populated.	Do you have all the information you need?	• CI owner (assigned to) • Cost center
Compliance	Audit the CMDB for its adherence to predefined regulatory requirements, internal governance, and certificates.	Is the data doing what you need the CIs to be doing?	• Workstations have anti-virus installed • Production servers have a QA server
Correctness	Test against predefined data integrity rules such as identification rules, orphan CI rules, and stale CI rules.	Is the data up to date and accurate?	• Duplicates • Server CIs not related to a service • CIs that exists in the CMDB but not on the network

How do I use the dashboard to improve CMDB health?

Drive resolution for compliance failures

☰	Follow On Tasks	Go to	Number ▾	Search

🔻 **Audit = Server Minimum Statistics**

⚙ 🔍 | ☰ **Number** ▲ | ☰ Short description

		Number	Short description
☐	ⓘ	TASK0004197	Update this server to match the conditions listed in the audit results
☐	ⓘ	TASK0004198	Update this server to match the conditions listed in the audit results
☐	ⓘ	TASK0004199	Update this server to match the conditions listed in the audit results
☐	ⓘ	TASK0004200	Update this server to match the conditions listed in the audit results

Proactively address top offenders

Top 10 Incident Generators

CI Name	Count
ⓘ MailServerUS	2
ⓘ Sales Force Automation	2
ⓘ SAP AppSRV01	2
ⓘ *JEMPLOYEE-IBM	1
ⓘ Adobe Acrobat 7.0.3 and Reader 7.0.3 Update	1

◀◀ ◀ | 1 | to 5 of 10 ▶ ▶▶

Track health over time

Missing Recommended Attribute CI Trend

Total Failed (0k–60k) vs dates 2017-01-11 through 2017-01-18

How do I use the CMDB health dashboard?(Continued)

What are the best practices for setting up the health dashboard?

Configure inclusion rules

Inclusion rules filter the CIs included in the dashboard. To avoid noise in the data, *limit the CIs included to those classes that are necessary to achieving your outcomes.*

Recommended CI classes for inclusion:

- Applications
- Servers (optionally change to computers)
- Virtual machines (VMWare, AWS, etc.)
- Network gear (routers, switches, etc.)

Determine required and recommended fields

Limit the number of required fields, and use required fields that are automatically populated (e.g., using Discovery or SCCM). Limiting the number of required fields reduces the risk that CIs identified through Discovery will be blocked.

As required, fields are mandatory across all areas of the CMDB while recommended fields only impact health metrics, *use recommended fields to inform health metrics without impacting other processes and to test potential future required fields.*

Create staleness rules

Health staleness rules help keep your CMDB up to date and accurate, since most infrastructure and application environments change rapidly. Out of the box, everything has a 60-day limit before becoming stale, but you can set limits by asset class. *Adjust the staleness rules by considering how often you're discovering these classes and if classes depend on external sources.*

For example:

- Servers are discovered weekly and have a 21-day limit
- PCs are imported monthly and have a 60-day limit

Determine orphan rules

When creating orphan rules (CIs missing a defined relationship within the CMDB), *consider your application stack and what relationships are most important to troubleshooting an issue and keeping your CMDB complete.*

Recommended orphan rules:

- Applications should be related to a server
- Virtual servers should be related to virtual machines
- VMs should be related to a hypervisor (ESX, AWS)
- WebSphere servers should contain an enterprise application archive (EAR) file