Network Discovery Data Quality

More and more companies adapt their IT processes to the ITIL best practices to increase the efficiency of their IT service management. According to Wikipedia, ITIL

> "is a set of practices for IT service management (ITSM) that focuses on aligning IT services with the needs of business. In its current form (known as ITIL 2011 edition), ITIL is published in a series of five core volumes, each of which covers an ITSM lifecycle stage. ITIL underpins ISO/IEC 20000 (previously BS15000), the International Service Management Standard for IT service management, although differences between the two frameworks do exist.
>
> ITIL describes processes, procedures, tasks and checklists that are not organization-specific, used by an organization for establishing integration with the organization's strategy, delivering value and maintaining a minimum level of competency. It allows the organization to establish a baseline from which it can plan, implement and measure. It is used to demonstrate compliance and to measure improvement."

The ITIL best practices also define the concept of a CMDB (configuration management database) that is a kind of data warehouse for IT organizations. A CMDB acts as a central repository for all IT related information including computers, software, etc. But how do you get computers, software and the like into a CMDB? Virtually all CMDB solutions offer import facilities from Microsoft Excel, CSV or XML files but also selected network discovery tools.

Network discovery tools are popular and widely used because they can collect hardware-, software- and configuration information from devices on the network quickly. After a network discovery tool has scanned the network , device information can be imported into a CMDB. IT processes such as incident-, problem management,etc, benefit from accurate and complete device information. However, inaccurate or incomplete device information might cause poor IT operations performance and might even lead wrong decisions.

What really confuses me is that many IT organizations blindly import device information from discovery tools into their CMDB without knowing the data quality. Most network discovery tools cannot measure the data quality of the device information that has been collected. Therefore, it is difficult for IT administrators to judge the data quality, especially if these data centers host thousands of servers.

What's Data Quality?

The data quality of network discovery tools has several aspects:

Accuracy	• Is the collected hardware-, software and configuration information accurate?
Completeness	• Have all networks, Windows domains, Active Directory objects and devices been discovered? • Did the discovery tool collect all device information that is technically accessible?
Normalization	• Is the hardware-, software and configuration information normalized so that it is compliant to industry standard representations?

Accuracy – Do devices really tell you the truth?

This might be a bit surprising. Shouldn't be device information correct, when a network discovery tool has collected it directly from a device? The short answer is: Not necessarily, devices do not always tell the whole truth!

Let's take a closer look on a typical example. The Windows Management Protocol (WMI) is the most common protocol for collecting hardware-, software and configuration information from Windows computers. Virtually all network discovery tools use WMI to collect hardware-, software and configuration information. WMI has been introduced with Windows NT 4.0 and is available since then. At the time when Windows 2000 hit the market, multicore processors have been in their infant stages and therefore the specifics of multicore processors were not reflected in the WMI data model. WMI represented a two core processor as two physical processors. When hyperthreading was turned on, a two core processor often times appeared as four processor machine. You might think, does that make a difference? Yes, it can make a big difference, especially for software license management. Many enterprise software companies license their software on the number of physical processors and the number of cores. Database server licenses are typically more expensive when running on more than one physical processor but are less expensive when running on a single physical multicore processor. Starting with Windows XP and Windows Server 2003, Microsoft added new attributes to the processor class in the WMI data model to represent multicore processors. This example shows the impact of devices reporting inaccurate hardware information that might negatively affect software license cost and software license compliance.

Note: Devices might report inaccurate information through their management protocols!
This can negatively impact software licensing cost and software licensing compliance.

The accuracy of device information is also dependent on how it is collected. The example above illustrates some aspects of the WMI protocol which most network discovery tools use to collect hardware-, software- and configuration information. If you had an agent installed on the target computer, the agent could read the computer's system management BIOS. The system management BIOS also stores the number of cores and threads and is accessible independently of the operating system. An agent could access the accurate hardware information even if the WMI protocol on Windows 2000 returns inaccurate hardware information. I have discussed the differences between agent-based and agent-less network discovery tools in my previously published whitepaper (http://www.jdisc.com/en/support/whitepapers). Agent based network discovery tools typically deliver more accurate data because the agent does have full access to all local resources such as the operating system or the system management BIOS.

The data accuracy of devices, on which agents cannot be installed, greatly depends on the manufacturer's implementation of the network management protocols. For example, most network switches provide the serial number as a SNMP MIB variable, however some switches do not have the serial number populated.

Data accuracy depends on multiple factors:

- The protocol that provided the information
- Agent based tools often collect more accurate information
- The device manufacturer's management protocol implementation

Completeness – Did we get it all?

You are enthusiastic about IT asset management (ITAM) and you have introduced a new network discovery tool. After running your first scan, the database is populated with hundreds or thousands of devices. Of course, you're happy! Without a network discovery tool, you could never gather that much information manually in such a short timeframe.

Then, your manager appears next to your desk and asks questions like:

- Did you get all my consultant laptops? They are only occasionally online!
- Did you get all hardware-, software- and configuration information of all devices?
- Did you discover all corporate sites?
- What can I do to improve the discovery results?

For most of the discovery tools, the simple answer is: *Hmm, well, I don't know!* And even worse, most of the tools offer none or only poor support to answer those questions.
Just imagine, you are preparing for an upcoming software compliance audit and you would like to know how many of your colleagues have Microsoft Office installed on their computers. Of course, your network discovery tool has found many of them. To prepare for the upcoming software compliance audit, you must be sure to have software information from *all* computers on your network. It is important for you to know, whether you have captured only 70% ,80% or even 95% of your devices!

Although many discovery tools offer a good and rich feature-set, only few ones assist their users to make sure all devices and hardware-, software- and configuration information has been captured.

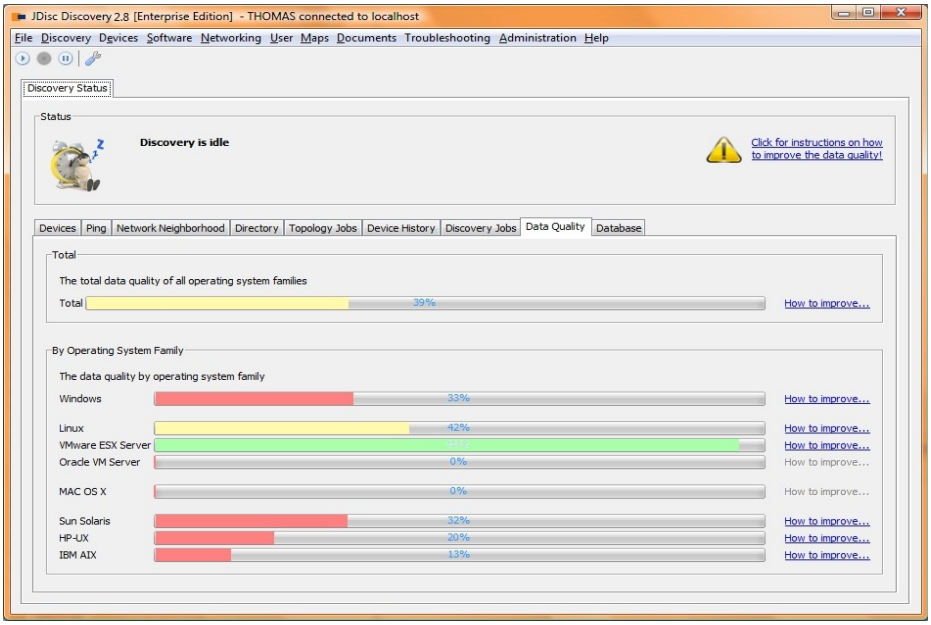
For example, a discovery tools that discovers member computers of an Microsoft Active Directory objects, could track the number of member computers of a directory object and compare it with the discovered computers that are assigned to the directory object. If these two numbers differ, you know your network discovery has missed computers that probably were offline at the time of the scan.

Even if could scan all of your computers, how do you know what the quality of the collected device information is like? Did you collect all technically accessible information from all devices? Are you missing device information just because access credentials are incorrect or network protocols have been blocked by firewalls? Ideally, a network discovery tool should *know* how to collect the best device information and notify you when access credentials are incorrect or important protocols fail.

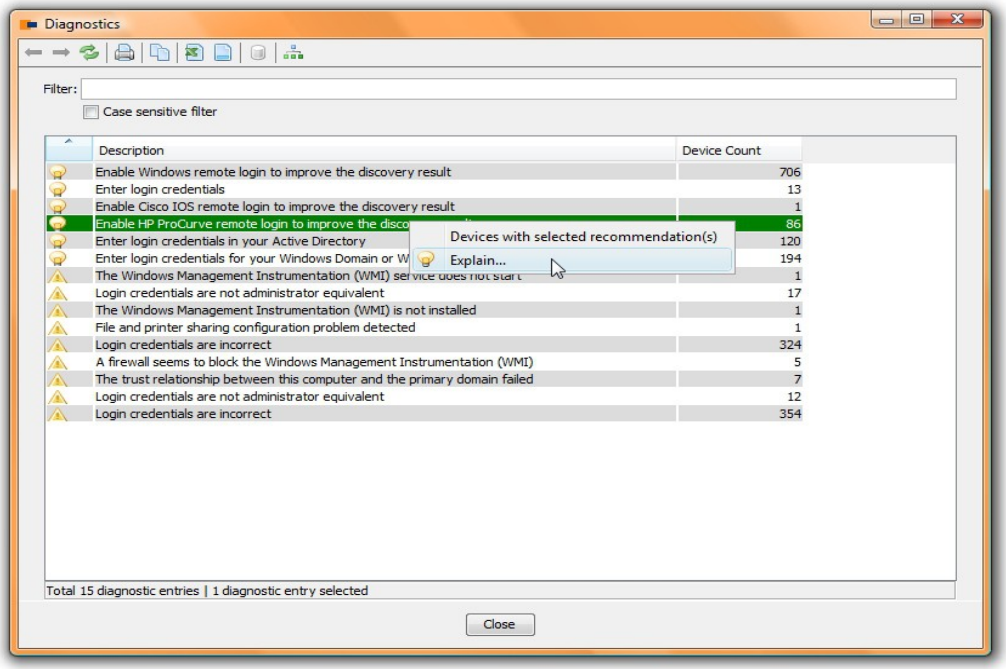
Network discovery tools should know how to collect the best device information possible and notify users when access credentials are incorrect or important protocols fail.

JDisc Discovery offers diagnostic rules that help users to improve the data quality. Diagnostic rules describe common protocol failures. For example, if you're getting a WMI error message x, then the most common reason is a firewall blocking protocol traffic. Based on the results of the diagnostic rules , JDisc Discovery calculates a *quality metric* that compares the accomplished data quality to the best possible data quality.

The screen shot below shows JDisc Discovery's quality meter dashboard. The quality meter is composed of a total data quality bar that is calculated for all devices and subsections that show the data quality for all devices by operating system family.



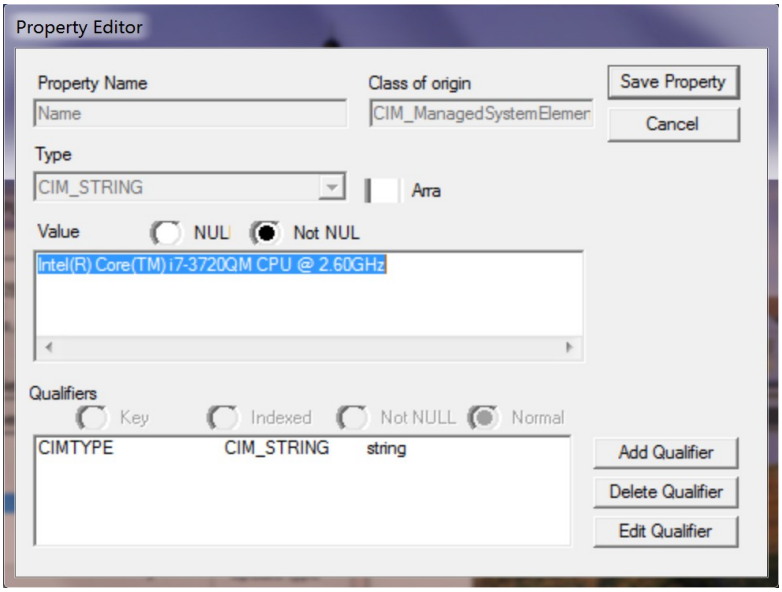
The links on the right to the respective quality meter bar link to reports that help improve the network discovery result for each operating system. The suggestions are based on the diagnostic rule results and JDisc's experience in designing network discovery tools.



JDisc Discovery runs its diagnostic rules on all devices and creates report such as the one above. Each row displays a problem along with the number of devices that are affected. Y more detailed explanation can be obtained by the context menu. You can drill down by double-clicking on a problem to display all affected devices or select the *Explain* context menu item to display suggestions helping to resolve the problem.

Data Normalization – Why that?

Data normalization is probably the final step to improve the device information quality. You might ask, what is data normalization all about? When you are in the network discovery business, you will quickly learn that development labs of all major device and operating system manufacturers represent the same information in many different ways. Product names and naming conventions are changed frequently.
Let's have a look at a typical example that illustrates the problem when collecting physical processors using the WMI protocol. The WMI query “SELECT * FROM WIN32_Processor” returns all processors. On the computer below, the query returned “Intel(R) Core(TM) i7-3720QM CPU @ 2.60GHz” as the processor model.



The same query returned “Intel(R) Xeon(TM) CPU 3.40GHz” and “Opteron(tm) Processor 250” on two more computers. Processor hardware manufacturers add a lot of “garbage” (from a reporting point of view) to the processor models. Wouldn't be a standardized processor model, such as “Core i7-3720QM”, “Xeon” or “Opteron 250” much more usable?

The naming of processor models is only a single aspect of data normalization. You will find very similar normalization problems also with installed software titles , device models, hard-disks, etc.

Network discovery tools that do not normalize the collected data are rather limited with regards to reporting and populating other IT systems, such as CMDBs. Collecting and storing raw device data in a database isn't too difficult. However, it is the data normalization that creates usable information from raw device data!

Data normalization isn't a nice to have feature, it is important for reporting and interfacing with other IT systems, such as a CMDB for example!

Conclusion

There are many network discovery tools on the market and these tools differ in the quality of the collected data but also how they support users in troubleshooting common problems.

When you evaluate a network discovery tool, pay attention to the data quality and if the tool can measure the quality of the collected data. Finally, check if the tool assists you in analyzing and solving common problems?

When you are managing a large IT environment, your discovery tool should be capable of measuring the quality of collected data and support you to troubleshoot common problems. While it might be feasible to manually check the data quality on hundreds of computers, this will no longer work when you manage an IT environment with thousands of clients and servers. This is why it is important that your discovery tool can measure the quality of collected data and support you to troubleshoot common problems.

This article has been written by Thomas Trenz, owner and founder of JDisc. Thomas has a long history in designing and implementing network inventory and discovery solutions. 2009, he started his own business with JDisc. JDisc's mission is the delivery of automated network inventory and discovery solutions for all kind of devices and all kind of different operating system platforms. Feel free to join the journey with JDisc Discovery on http://www.jdisc.com or Thomas's blog on http://blog.jdisc.com.

A shortened version of this article has first been published on the ITAM review (http://www.itassetmanagement.net).