

# CMDB in 5 Steps

## A project guideline for implementing a configuration management database

**Authors:**

Ralf Buchsein and Tobias Müller



## Table of Contents

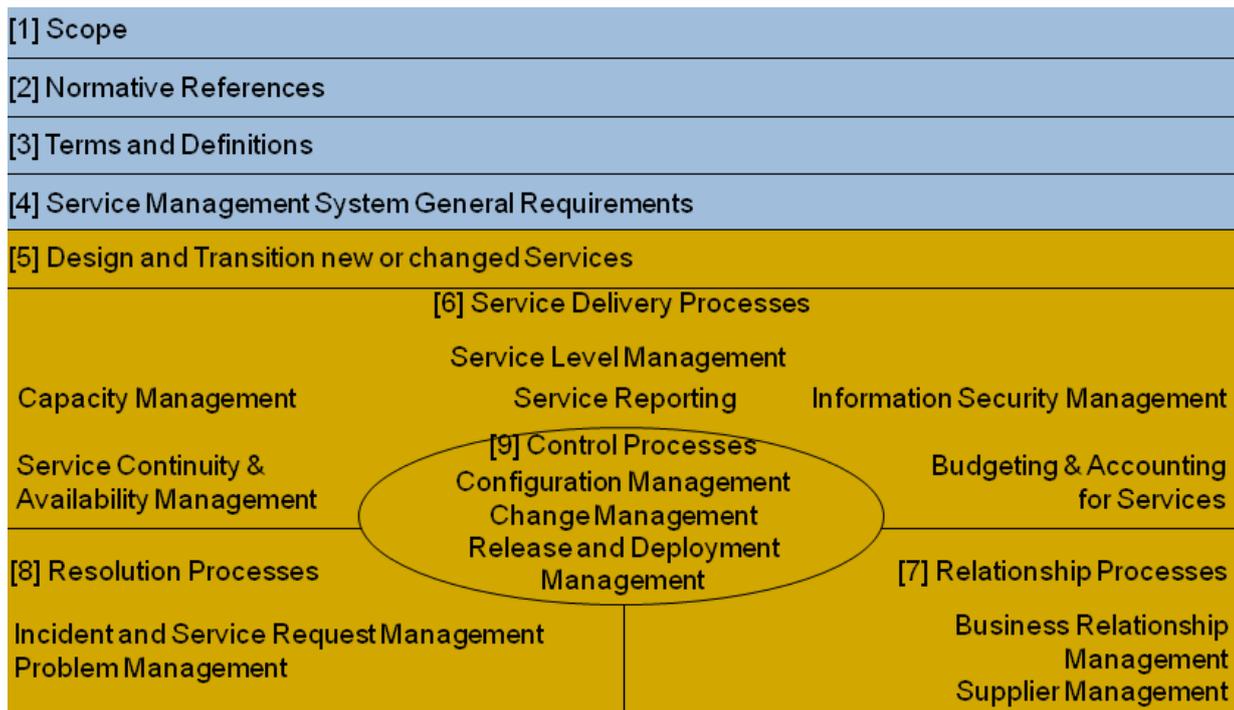
1. The Significance of the Configuration Management Database for IT Service Management	3
1.1 The CMDB as Basis for Cost Savings and Process Optimization	4
1.2 The Way to a Successful Project	4
1.3 Requirements of ISO/IEC 20000	6
2. Building a CMDB in Five Steps	6
2.1 Step 1: Define the Objectives	6
2.2 Step 2: Assign Responsibilities and Staff Motivation	7
2.3 Step 3: “Bottom-up” or “Top-down”: Select the Right Approach	7
2.4 Step 4: Define the CMDB Content	8
2.4.1 Definition: Configuration Item	8
2.4.2 Content of the CMDB	8
2.4.3 Examples of Attributes for a Hardware/Software CI	9
2.4.4 Examples of Attributes for a Hardware Component	9
2.4.5 Four Rules for Selecting the Appropriate Level of Detail	9
2.5 Step 5: Fill the CMDB with Data	10
3. Tool Support with iET ITSM and iET Inventory Center	10
3.1 Structure of the CMDB in iET ITSM	12
3.2 Products, Components and their Attributes	13
3.3 Configuration Items and their Attributes	14
3.4 Filling the Inventory Staging Area with CIs	14
3.5 Creating Links: Data Mapping	15
3.6 Filling the CMDB	16
3.7 Variance Comparison when Using CMDB	16
3.8 Summary: Building and Filling the CMDB	17
4. Glossary	18

## 1. The Significance of the Configuration Management Database for IT Service Management

Organizations adopting ITIL® based standards and processes frequently start with the implementation of incident management, followed by problem management. This stands to reason, since both processes are extensions of the original helpdesk, which usually already exists and can be enhanced and extended. In this way quick results can be achieved with regards to improved customer service. As a next step, the focus is usually on change and release management, since organizations have realized they need a configuration management database (CMDB). But the implementation of a CMDB presents many challenges, both technical and organizational.

In line with ITIL® best practices, the configuration management database is the central database of IT service management and thus the foundation for high quality customer service, stable systems and applications, as well as a controlled IT infrastructure. All ITIL® processes and other processes such as IT security management depend on the configuration management database and can only function seamlessly if the database provides current and accurate data. While configuration management is the discipline responsible for the CMDB, change management, according to ITIL®, is the process that controls the changes in the CMDB. In addition, ISO/IEC 20000 puts configuration and change management as control processes clearly in the center of its requirements.

Outline of ISO/IEC 20000 (source: KESS IT Consulting & Training)



## 1.1 The CMDB as Basis for Cost Savings and Process Optimization

Today, legal requirements or software audits demand a detailed knowledge of the IT infrastructure, which can only be gained with a CMDB. In addition, organizations can assess potential risks of changes in advance and thus minimize negative effects on the business operations with the help of a CMDB.

### ITIL® describes Configuration Management as follows:

“Configuration management creates a logical model of the infrastructure and the available services. This happens by identifying, controlling, maintaining and verifying the versions of all existing configuration items (CIs). The objectives of configuration management are:

- Provide information about all IT components and configurations within the enterprise and its services.
- Make information available about configuration and its documentation for support of all other IT service management processes.
- Provide a solid base for incident, problem, change and release management.
- Verify the configuration documentation as compared to the infrastructure and address deviations if applicable.“

The ITIL® best practices describe the CMDB as “a database that contains all relevant details for each configuration item (CI) as well as details concerning the important relationships between the CIs.” With regards to the depth and detail of information in the CMDB, ITIL® simply states “everything you wish to keep under control.”

“Configuration management creates a logical model of the infrastructure and the available services. This happens by identifying, controlling, maintaining and verifying the versions of all existing configuration items (CIs).”

## 1.2 The Way to a Successful Project

Many organizations try hard to show their entire IT infrastructure, including all of the detailed information, in the CMDB. This flood of information can lead to significant delays and may result in a failure to achieve the original objectives within the agreed time, quality and cost constraints.

The CMDB can lead to improvements in virtually all IT processes and can support organizations in their compliance efforts with regards to legal requirements. It can help meet these legal requirements since installed software instances are also maintained as CIs in the CMDB.

Table 1 on the next page shows which functions are supported by the CMDB in the relevant IT service management processes.

Process	Supported Functions
Incident management	Incidents can be prioritized more accurately based on the relationships between affected CIs. Incidents can be resolved faster, because the entire CI information for each user is visible.
Problem management	Detailed information for problem analysis is available.
Change management	Supports the analysis of potential ramifications to the production environment after changes take place.
Release management	Availability of information for release planning and execution, so that rollouts do not have a negative effect on the business.
Configuration management	The CMDB is the database in which the IT infrastructure is recorded and described, delivering a detailed model of the IT infrastructure for all processes.
Service level management	Availability of information about the CIs that support an IT service. Without this data, SLAs cannot be thoroughly created and adhered to.
Financial management for IT services	CI information is enhanced by financial data, which is necessary for the cost and service billing.
Availability management	Measures and controls the availability of the configuration items and the delivery of information to identify weak points.
IT service continuity management	Based on the CIs, emergency contingency plans and baselines are created, which should be available at an alternative location.
Capacity management	The CMDB delivers information for capacity planning.
Security management	The CMDB includes classifications from security management for trustworthiness, integrity and availability, and provides information for risk management.

Functions supported by the CMDB

### 1.3 Requirements of ISO/IEC 20000

While the ITIL® best practices only represent recommendations, an international standard has been defined that specifies the minimum requirements for IT service management. Among other things ISO 20000 demands “All CIs must be uniquely identifiable and recorded in the CMDB, ( ... ) for a CI, the related changes and problems must be available.”

## 2. Building a CMDB in Five Steps

A multitude of sources deliver data to the CMDB. This includes information which is redundant, meaning it is not necessary or it duplicates or contradicts other information sources. IT organizations need to define a process that determines the necessary data, how data storage and data updates are handled, and which sources are reliable. Furthermore, objectives must be defined and process responsibilities must be clearly assigned.

Requirements for a successful implementation are:

- Definition of objectives from a business perspective
- Acceptance and support from users and management
- Reasonable project timeline with measurable goals and swift integration into the daily business

#### CMDB in 5 Steps

- 1. Define the objectives
- 2. Assign responsibilities
- 3. Select the right approach
- 4. Define CMDB content
- 5. Fill the CMDB with data

### 2.1 Step 1: Define the Objectives

A formalized process, additional tools to implement configuration management, and a supporting CMDB should be introduced in order to deliver specific business benefits. Ideally there should be reliable information on the current status and performance of the IT organization. From this information, and knowledge of the organization, goals can be defined that are desirable, measurable and achievable. Questions to be addressed include:

- Which current difficulties, bottlenecks, etc. are the trigger for this project?
- Which benefits do you expect? Which services will be improved?
- Who will use the CMDB? What kind of information will the system users search for? What expectations do they have?
- What are your measurable criteria for success?

The reasons for building a CMDB vary and can provide important pointers towards the most useful implementation procedures (see chapter 2.3).

## 2.2 Step 2: Assign Responsibilities and Staff Motivation

Change is uncomfortable and results in additional work. Make sure all employees are brought on board from the start, and put together an interdisciplinary team. The team needs to participate in defining goals relevant to the CMDB project and dividing the project into phases to deliver demonstrable short-term success.

Be clear about responsibilities. There must be a process owner for configuration management. By clearly assigning responsibility you can help guarantee the success of the project and ensure that the CMDB delivers powerful and accurate information. Someone should also be responsible for the various information sources for the CMDB, to ensure that interfaces are correctly implemented and maintained.

Configuration management has to go hand in hand with the establishment of change management and the designation of a change manager. As soon as the CIs are recorded, the maintenance process has to be in place. Otherwise, the accuracy of the CMDB cannot be guaranteed and the investment may be wasted.

Employees in the following departments are ideal candidates for the project team:

- Administrator of the IT infrastructure (often described as asset management or inventory management)
- Application management
- Staff members from all processes, especially incident, problem and change management.

Communication is an important element for success. Intranet sites are a proven tool where project goals and achievements can regularly be communicated. Reports in employee newsletters, round table discussions and informational events enhance employee awareness of the significance of the project and keep the team motivated.

## 2.3 Step 3: “Bottom-up” or “Top-down”: Select the Right Approach

“Bottom-up” or “Top-down” – the best procedure strongly depends on the size of the organization, number of configuration items and overall objectives. With a bottom-up approach, one has to be careful not to lose sight of the goal. Usually many “local CMDBs” already exist in the form of files, databases, etc. which need to be consolidated. The challenge here is to successfully consolidate the administration of these local CMDBs while meeting the dual objectives of providing sufficient detail for specialist users without the CMDB becoming too large and unwieldy to be an effective tool. Replacing local tactical information sources maintained by individuals can be politically sensitive since it can be construed as a reduction in their responsibilities.

Furthermore, you will be swamped with data, which for the most part is not necessary. When combining information from software distribution solutions and user administration, several thousand relationships can occur in organizations with 1000 PC workstations and 30 applications, and they will all need to be maintained in the CMDB. The danger exists that you will establish a central database with a vast number of IT assets which will never fulfill the characteristics of a CMDB, such as accurate relationships or restriction to important configuration items.

Therefore, iET Solutions recommends the “Top-down” approach. Instead of recording every configuration item with all configurations, focus should be on the elements that have the greatest impact on the delivery of the IT services. Check your goals: What benefits are expected? Which services will be enhanced?

In reality, a scan of the IT infrastructure will be performed with the help of a discovery solution. The amount of discovered information will be reviewed according to which configuration items are supporting enterprise-critical services and which ones will be recorded in the CMDB. It is not desirable to record every CI with every configuration attribute in minute detail. A frequent successful approach is to start with two or three of the most important services and then gradually expand the CMDB. Often the product catalogue of an organization will be used as a guideline for building the CMDB.

The requirements of ISO 20000 provide tips for the size and level of detail of a CMDB: “(...) it must be assured that the degree of control is appropriate for the business requirements, error risks and criticality of the service.” This references the top-down approach. Organizations should take a look at their defined IT services and analyze which components and systems are necessary for performing the services and which detailed information is required. The challenge lies in the critical review of the necessary components, but this review also provides the opportunity to limit the size right from the start in order to quickly use the CMDB in a production setting.

## 2.4 Step 4: Define the CMDB Content

Within configuration management, the CIs, their relationships, and their relevant attributes are defined. The main question here is “Which information do we need to successfully deliver our IT services?”

### 2.4.1 Definition: Configuration Item

According to ITIL® a configuration item is “a part of the infrastructure that is required for the delivery of systems, applications and services.” The depth of information is virtually unlimited. For instance, a laptop can be recorded as one CI, or each of its components can become a single CI (processor, RAM, network card, drive ...). Decisions on the level of granularity, and the level of detail at each level of a CI must be made based upon what information the process participants need to deliver the services and comply with existing legal requirements.

For example, if desktop PCs are leased for a fixed period and covered by an external support and maintenance agreement there may be no need to record child CIs or components. On the other hand, if the IT staff frequently swaps components and upgrades systems during their lifetime then additional detail would be justified.

### 2.4.2 Content of the CMDB

The content of the CMDB is structured as follows:

#### 1. Configuration Items (CI)

Elements of the IT infrastructure, which support a service, such as hardware and software.

- Hardware: PCs, laptops, printers, servers
- Network: Cables, switches, router, hubs
- Software: Operating systems, applications
- Documentation and contracts

#### 2. Components

Elements of the IT infrastructure, which should not be recorded as CIs, to reduce the data volume of the CMDB and to simplify the maintenance.

- Graphic or network cards

### 3. Attributes:

Technical and economic data, which describe a configuration item or a component in more detail, such as:

- Physical attributes delivered by the discovery solution, e.g. manufacturer, model and serial number
- Logical attributes which are recorded manually or by connecting to another system such as cost centre, purchase price, etc.

### 4. Relationships

This refers to CI relationships as well as relations to services:

- Service to CI relationship
- CI to CI relationship
- CI to ITSM processes, for instance, to incidents and changes

#### 2.4.3 Examples of Attributes for a Hardware/Software CI

- CI-name / name of component (= unique identifier)
- Type
- Serial number (hardware) or license number (software)
- Model number (hardware) or version number (software)
- End date of warranty (hardware), end date of maintenance (software)
- Location (hardware)
- Owner / user
- Source / provider
- Installation date
- Current status (e.g. for hardware ordered, received, in use, archived, not used, in stock, in repair)
- Notes

#### 2.4.4 Examples of Attributes for a Hardware Component

- Name of the component (= unique identifier)
- Type
- Model number
- End date of warranty
- Source / Provider
- Current status (ordered, received, in use, archived, not used, in stock, in repair)

#### 2.4.5 Four Rules for Selecting the Appropriate Level of Detail

- Record all that will be needed (also in the future)!
- Don't record what will not be needed!
- Only record what can be measured!
- Watch the effort regarding data maintenance!  
(The effort for delivery of information must be consistent with its usefulness.)

## 2.5 Step 5: Fill the CMDB with Data

An organization is unlikely to enter all information manually in the CMDB. This is not manageable considering the complexity of today's IT environments. Moreover, the data would not be complete and accurate at any one point in time. Inventory discovery solutions gather most of the configuration items by scanning the IT infrastructure and delivering the data to initially populate the CMDB and subsequently update the CMDB.

The quality of the CMDB increases if the following is done:

- Determine the “leading systems”, from where data will be imported, e.g. user administration, employee personnel data from HR systems.
- Automate data import and synchronization.
- Define the maintenance process and use tools to implement it.
- Reduce the number of interfaces to easily support administration and maintenance.

The CMDB always represents the nominal inventory and also contains regulatory compliance information such as software licenses in use. Therefore, any changes to the CMDB must be made in a controlled fashion, and be authorized and verified by change management. As a result, the database of an inventory discovery solution never represents the CMDB, but only the actual inventory. Variances between nominal and actual inventory should be automatically marked, but a potentially necessary change in the CMDB must be performed manually and in a controlled way. An inventory discovery solution must never change the nominal inventory without verification. Just because the solution finds games like “Solitaire.exe” that come automatically with some programs, does not mean they represent CIs and they do not belong in the CMDB.

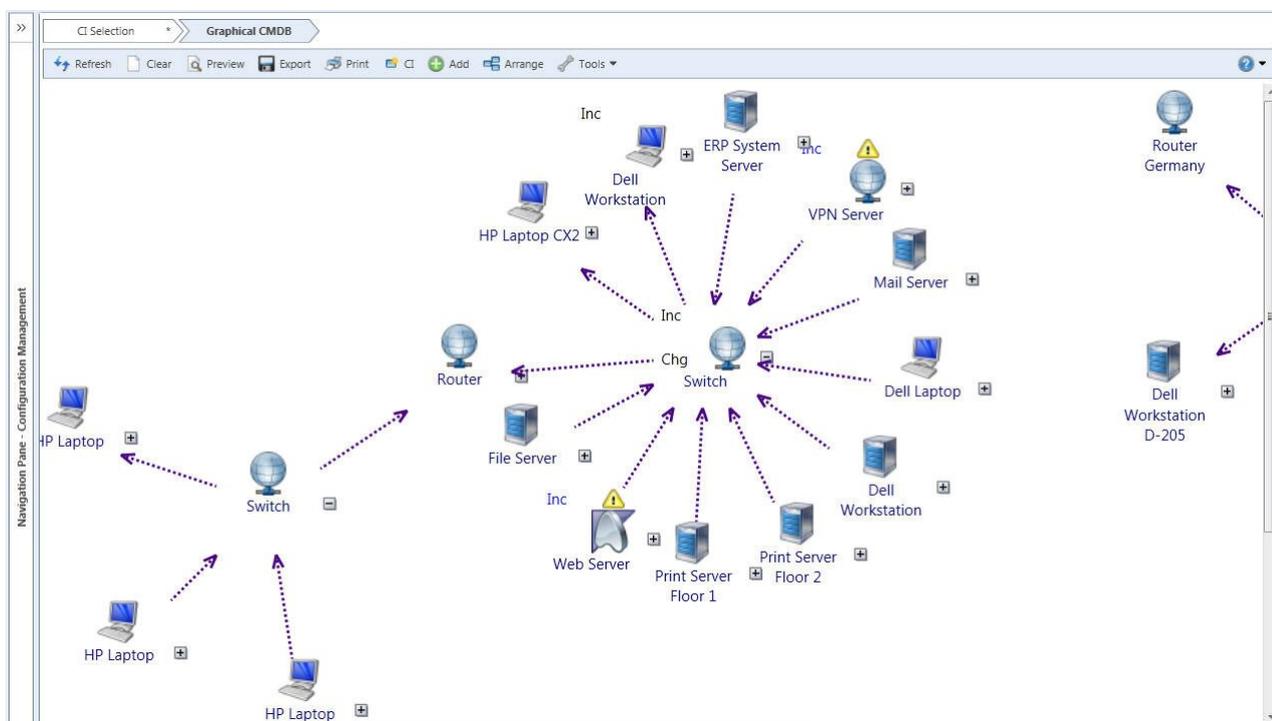
Increasingly stringent regulations and higher penalties associated with licensing infringements are additional factors that make it more critical than ever to maintain accurate and in-depth information on configuration items. These requirements appear at first glance to focus on accounting and auditing. But data management in general and financial reporting in particular is almost completely reliant on the performance of the IT infrastructure, such as networks, software, servers, and desktops.

The regulations require documented controls and processes for the IT hardware and software used in the financial reporting system and, where configuration items are material to financial statements, for those CIs themselves. Most companies have a huge number of CIs involved in financial reporting, ranging all the way from the computers used to track inventory, to the ones used to prepare the income statement. Knowing where all of them are, what safeguards are in place to prevent unauthorized access, and that you will be able to detect any changes, is key. Managing authorization processes and providing an audit trail to track changes with the help of the CMDB and change management is especially important when adopting legal requirements.

## 3. Tool Support with iET ITSM and iET Inventory Center

The configuration management database is part of iET ITSM, the IT service management offering from iET Solutions. In addition to the CMDB, iET ITSM includes applications for incident, problem, change, release, configuration, and service level management. A special feature of iET Solutions' iET ITSM focuses on the relationships between the configuration items as well as the graphical representation of the related incidents, problems, changes, and releases.

With its additional modules, iET Inventory Center and iET CMDB Intelligence, iET Solutions offers enhancements to iET ITSM, which provide additional automation to support the change and configuration management processes. iET Inventory Center enables the agentless discovery of complete networks from a central point. It scans physical and virtual devices including all their components such as model, processor, memory modules, drives, sound and video adapter, screens and installed USB devices. Moreover, it inventories additional hardware such as switches, routers, print servers, etc. Of course, data on all installed software packages, apps, hotfixes and the configuration of Windows systems will also be gathered. Via existing APIs, iET Inventory Center can reach even for more devices running Mac OS-X, Linux, VMware ESX, etc. with SNMP or SSH interfaces. All data is stored in a central database so that you can analyze it immediately by running dynamic, freely configurable and clearly arranged reports. These may be used as documentation for audits and for controlling purposes within asset management.



The graphical CMDB in iET ITSM displays relationships of CIs. The relations to an incident are indicated in the graphic with “Inc” and to a change with “Chg”.

Example for...	Laptop	Desktop	Monitor	Printer	Operating System
Product	Every model that is approved by company policy will be maintained as a product, e.g. type laptop	Every model that is approved by company policy will be maintained as a product, e.g. type desktop	Every model that is approved by company policy will be maintained as a product, e.g. type monitor (CRT, flat screen)	Every model that is approved by company policy will be maintained as a product, e.g. type printer	All available, tested, licensed and company policy approved operating systems, e.g. type operating system
Attributes for which values are predefined at the product level. Values are automatically inherited by each CI assigned to this product.	Model Manufacturer Supplier CPU RAM Hard drive (number and capacity)	Model Manufacturer Supplier CPU RAM	Model Manufacturer Supplier Resolution Screen size	Model Manufacturer Supplier	Manufacturer Supplier Version (Professional, Standard, Enterprise Edition) Language
CI	A particular unit with unique identifiers such as inventory number, owner, location and status.	A particular unit with unique identifiers such as inventory number, owner, location and status.	A particular unit with unique identifiers such as inventory number, owner, location and status.	A particular unit with unique identifiers such as inventory number, owner, location and status.	A particular software with unique identifiers such as serial number, user and status.
Attributes which are different for each CI. Values are therefore added at CI level.	Inventory number Serial number Host name IP address MAC address Netmask	Inventory number Serial number Host name IP address MAC address Netmask	Inventory number Serial number	Inventory number IP address Host name MAC address	Serial number
Relationships	Incidents Problems Changes Releases Related hardware and software CIs Contracts Documentation	Incidents Problems Changes Releases Related hardware and software CIs Contracts Documentation	Incidents Problems Changes Releases Related hardware and software CIs Contracts Documentation	Incidents Problems Changes Releases Related hardware and software CIs Contracts Documentation	Incidents Problems Changes Releases Related hardware and software CIs Contracts Documentation

### 3.1 Structure of the CMDB in iET ITSM

The CMDB is an integral element of iET ITSM and is structured to allow the definition of products, CIs, components and attributes.

The example above does not show components for reasons of clarity.

## 3.2 Products, Components and their Attributes

iET Solutions understands a product as a template, which serves as the starting point for the recording of CIs. Products are based on the product catalogue that already exists in many IT organizations. A decision must be made as to which products should be administered in the CMDB. This can be the entire product catalogue or a subset of products.

The use of products simplifies the recording of CIs. By creating a product, you create a template that defines general attributes, and default values for those attributes where required. Those attributes can be matched with values that are identical for each CI. Attributes and values are automatically inherited for each CI that is assigned to the product.

iET ITSM allows for the definition of components. Components are elements of a product which are not listed as CIs to keep the process simple and to avoid excessive maintenance efforts. The use of components is optional. Depending on need and objectives, you can also define the elements as CIs instead of components. The higher maintenance effort, however, needs to be carefully assessed in advance.

The definition of products and components is the most important part when building a CMDB. In this phase, you decide which CIs will be administered later in the CMDB and how much detail will be recorded. Depending on the size of the organization and the number of existing IT assets, this process can be a big part of the project.

### Example for a product:

Product	Laptop
Attribute, which is relevant for each CI, <b>with</b> a predefined value	Attribute: Manufacturer Predefined value: Toshiba will automatically be inherited at CI level
Attribute, which is relevant for each CI, <b>without</b> a predefined value	Serial number The actual value, i.e. the serial number itself, will be recorded at CI level.
Relationships	To services and components will automatically be inherited at CI level

### Example for a component:

Component	Graphic card "Nvidia"
Attribute for a component	RAM of graphic card
Relationships	To products

### 3.3 Configuration Items and their Attributes

Unlike products which only serve as templates, or definitions of objects, the CIs represent actual instances of hardware and software as well as documentation (e.g. contracts). Attributes and components are inherited from the product as well as component attributes. These are populated with actual values when the CI is created. As mentioned above, the most efficient way of creating and describing the CI is with the inventory discovery tool and iET CMDB Intelligence.

CI	Toshiba Laptop Satellite Pro
Attribute - Serial Number	Value = 4711
Component – Graphic Card	Value = GeForce
Attribute RAM for the component graphic card	Value = 12 GB
Relationships to services and contracts	Will be inherited from the product
Relationships to CIs	<ul style="list-style-type: none"> <li>■ To software CIs, e.g. installed software “Windows 11”</li> <li>■ To hardware CIs, e.g. monitor</li> </ul>

#### Example:

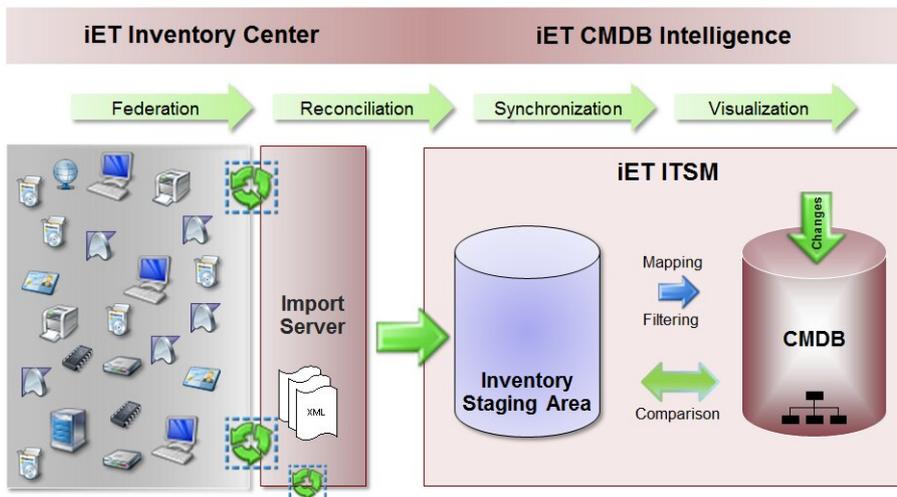
For software CIs, iET Solutions offers the ability to use the so-called Gathered CIs. With a Gathered CI, a particular type of software, such as Microsoft Windows Word 2013, will only be recorded once with all its details. All computers that have Microsoft Windows Word 2013 installed will be displayed via the CI relationships. This significantly reduces the quantity of CIs that are administered in the CMDB. On the other hand, specific data such as different serial numbers can no longer be maintained. The use of Gathered CIs does not comply with the guidelines of ITIL® and has to be carefully assessed prior to implementation. The decision to use this feature will depend on data volume and the objectives of the CMDB implementation at a specific organization.

### 3.4 Filling the Inventory Staging Area with CIs

XML files are created, with the help of the inventory discovery solution, that contain all detected configuration items and their details. These XML files are written to the inventory staging area via the iET Import Server. Alternatively, the data can be called via ODBC from the database of the inventory discovery solution. The inventory staging area serves as the database for the actual inventory. iET Solutions considers configuration items in the inventory staging area as “inventory CIs.” These are administered in the inventory staging area separately from the nominal CIs, which are administered in the CMDB.

Usually during the first import, also called the initial import, all data is written to the inventory staging area. Afterwards the information is reviewed and those pieces of information that will not be administered in the CMDB are filtered out. The decision is made for products, configuration items, components and their attributes. Setting filters reduces the amount of data. Filters can be set to prevent data from being written to the inventory staging area in the first place and/or to exclude data from being imported into the CMDB.

Once a filter is set, it is also valid for updates and new imports. For instance, filters allow the exclusion of data relating to detailed BIOS information or executable program files, e.g. Windows Live Update, that come with the operating system.



The actual inventory is saved in the inventory staging area before the import into the CMDB takes place.

### 3.5 Creating Links: Data Mapping

iET Solutions defines mapping as the linking of products and their attributes, as well as components and their attributes, with their equivalents in the CMDB. The benefits of mapping are:

- Identical products may have different labels in the staging area because information has come from more than one source. They can be mapped to one product in the CMDB.
- Detailing of products, components and attributes can be controlled.
- Descriptions can be unified: for instance, different inventory discovery solutions show the labels RAM and RAM-element, which can be combined as RAM in the CMDB.

#### Example of mapping:

Mapping is possible at the product, component and attribute levels. The following example illustrates the mapping of a product:

- The inventory discovery solution finds the product “Microsoft Word” under different labels, since sometimes it was purchased from an OEM provider and sometimes directly from the manufacturer. The labels are: “OEM”-Word, Microsoft Word.
- In the CMDB all products will be combined under the same product with the label “Microsoft Word.”
- Mapping assigns the products of the inventory staging area (“OEM”-Word, Microsoft Word) to the product Microsoft Word in the CMDB.

Mapping ensures that during future updates the different product labels are automatically assigned to the appropriate product in the CMDB.

### 3.6 Filling the CMDB

If the CMDB already contains data, then the inventory CIs and CMDB CIs have to be mapped once. These mappings are maintained for later updates.

After mapping and filtering, the replication takes place. During replication, the desired inventory CIs are written into the CMDB, taking into account the defined filters and established mappings.

Users can replicate the entire inventory of the inventory staging area or individual inventory CIs that have been newly added or updated, i.e. full or incremental replication.

### 3.7 Variance Comparison when Using CMDB

During variance comparison any differences between the actual inventory (= inventory staging area) and the nominal inventory (= CMDB) will be displayed. This variance comparison allows the following:

- Avoids duplication of configuration items (CI).
- Allows the analysis and comparison of the actual IT inventory with the assets stored in the CMDB.
- Detects unauthorized changes and initiates the appropriate measures.
- Identifies authorized, but flawed or non-executed changes.

The variance comparison can be enabled or disabled all the way down to the attribute level, i.e. differences that are not relevant for the IT operation can be excluded from the analysis.

IP addresses are a good example. A DHCP server regularly assigns new IP addresses to laptops and PCs. The different IP addresses would be flagged as differences with each variance comparison. But since they have no ramifications for the IT operation, they are considered irrelevant and should not be analyzed. iET Solutions provides the ability to specify the following:

- **Exclude** the IP addresses from the CMDB update and not to record the difference or
- **Automatically overwrite** the IP address and not to record the difference.

This control ability allows organizations to reduce the administrative effort and to focus on the analysis of important changes only.

Variance Comparison of CIs - 2000160

Restore

CMDB CI ID: 8  Single CI

CI Description: Dell Monitor 1400

Product Description: Dell Monitor

Consider settings for automatic update

Consider settings for accept difference

Attributes Components Relationships Between CIs Changes

Number Inventory CI Attributes: 3 Number CMDB CI Attributes: 3

Inventory Attribute	Inventory Attribute Value	CMDB Attribute	CMDB Attribute Value	Difference Accepted	Automatic Upd...
Serial Number	380700	Serial number	342er127	N	N
Model	W2442	Model	c1400	N	N
Manufacture Week	3/2010	Install date	03/17/2005	N	N

Legend: Unmapped labels Different or missing values

Cancel

### 3.8 Summary: Building and Filling the CMDB

Here is a summary of the steps for building a CMDB in iET ITSM:

- Verification of the actual current position: Scanning the IT infrastructure and reviewing the data. Information is recorded in the inventory staging area.
- Creating products in the CMDB:
  - Decision is based on the product catalogue, which products will be administered in the CMDB.
  - Determining the detail information, i.e. defining attributes and components.
- Creating filters for
  - Products
  - Components
  - Attributes
  - Configuration items (only in the event a CMDB already exists and needs to be updated)
- Mapping of the actual inventory of the inventory staging area to the nominal inventory of the CMDB for
  - Products
  - Components
  - Attributes
  - Configuration items (only in the event a CMDB already exists and needs to be updated)
- Replication: Saving the inventory CIs from the inventory staging area as CMDB CIs in the CMDB after applying filtering and mapping. This process is performed at the initial import and then every time new CIs are automatically found by the inventory discovery solution. The following automatically occurs in this case:
  - Mapping of CIs to products
  - Filling the attribute lists that are inherited from the product with values
  - Filling components that are inherited from the product with values
- Creating relationships to other CIs, as well as services and contracts if these are not inherited from the product. iET CMDB Intelligence automatically creates the relationship between installed software and relevant hardware CI. Insofar as the appropriate inventory discovery tool supports it, the iET CMDB Intelligence also establishes the relationship between hardware CIs, e.g. PCs to printers and monitors.
- Variance comparison:
 

If the CMDB already contains data, then the automatic variance comparison shows the differences between the inventory staging area and the CMDB. This variance comparison allows the control of the change and configuration management.

## 4 Glossary

### Asset Management

Asset management administers the assets from a business perspective. Values and costs play a role.

### Inventory Management

Inventory management includes the procedures for detection and administration of the inventory of IT assets. Mostly the technical features are maintained here.

### Configuration Management

Configuration management does not replace the inventory or asset management, but compliments it. configuration management includes information about dependencies and configuration relationships between IT components and documents in the nominal inventory via the CMDB.

### Baseline

At a certain point in time, all configuration items, including their characteristics, are recorded and documented. The starting inventory is saved and cannot be changed. A baseline can be used as follows:

- a starting point for further expansion and for verification of new configurations.
- a fall-back position, in case new configurations (after changes) cause problems.
- a standard for the delivery of configurations to the user.

### Relationships

Relationships of configuration items among each other as well as to documentation (e.g. contracts) and to other ITIL® processes, in particular to incident, problem, change and release management. Relationships can be of many different types, either defined in the standard ITSM solution, or as defined by customers.

### OLA, Operational Level Agreement

Service agreement with internal departments providing a supporting service, e.g. network, application support.

### SLA, Service Level Agreement

Service agreements with end user organizations, internal or external.

### UC, Underpinning Contract

Contracts with partners, external service providers, etc.

## ABOUT iET SOLUTIONS

iET Solutions, a division of UNICOM® Global, helps large and midsize enterprises to increase the efficiency and security of their IT operations and infrastructure. Organizations around the world use software from iET Solutions for IT service management (ITSM), software asset management (SAM) and enterprise service management (ESM). The company has more than three decades of experience in service management and works with organizations across all industry sectors.

### Locations

#### Worldwide Headquarters

iET Solutions, LLC.  
UNICOM Global HQ  
UNICOM Plaza Suite 310  
15535 San Fernando Mission Blvd.  
Mission Hills, CA 91345, United States

Phone: +1 818 838 0606  
info@unicomglobal.com  
www.unicomglobal.com

#### European Headquarters

iET Solutions GmbH  
Humboldtstr. 10  
85609 Aschheim, Deutschland

Phone: +49 89 74 85 89 0  
info@iet-solutions.de  
www.iet-solutions.de

#### UK Headquarters

iET Solutions  
Macro 4 Limited  
The Orangery  
Turners Hill Road  
Worth, Crawley  
West Sussex RH10 4SS, United Kingdom

Phone: +44 12 93 87 23 00  
info@iet-solutions.co.uk  
www.iet-solutions.co.uk