

## **Configuration Management for the Rest of Us**

By [Hank Marquis](#)

Hank is EVP of Knowledge Management at Universal Solutions Group, and Founder and Director of NABSM.ORG. Contact Hank by email at [hank.marquis@usgct.com](mailto:hank.marquis@usgct.com). View Hank's blog at [www.hankmarquis.info](http://www.hankmarquis.info).



Configuration Management is probably the least understood and most important service management process. Yet most have no idea how to start and many think it requires huge investments. But you do not need to spend big bucks to get real benefits.

The IT Infrastructure Library<sup>®</sup> (ITIL<sup>®</sup>) describes Service Asset & Configuration Management (SACM) as a method for controlling infrastructure and services.

The ITIL goes into detail describing the goals and activities relating to SACM. Given these descriptions of SACM, it can appear that ITIL implementation is not possible without a mature SACM process.

This is not true, but the use of over-generalizations like "account for all the IT assets and configurations within the organization and its services" can cause new practitioners to become confused and overwhelmed.

Coupled with vendor hype and management misunderstanding, practitioners quickly lose sight of the single reason for SACM - controlling changes.

Losing focus on the purpose of SACM makes it more difficult than it would be otherwise. A clear understanding of the real purpose of SACM can make it easier to achieve, help explain its benefits, and provide a roadmap for its justification and implementation.

There are three basic approaches available to the practitioner regarding Service Asset & Configuration Management: Purchase or build a comprehensive system, or codify and enhance existing systems.

Purchasing or building a SACM product is expensive. But there is an alternate approach - using the SACM systems virtually ever organization already has in place.

Following I explain the real purpose of SACM, show you how to uncover your current SACM systems, and provide a plan to implement SACM with real benefits for little or no additional cost.

### **The Real Purpose of Configuration Management**

Regardless of what you heard, read, or seen, the purpose of SACM is to control changes through creation and maintenance of documentation. This is not the same as Change Management, which is a process for evaluating and handling change requests in the pursuit of quality of service improvement.

SACM is a process to identify, record, maintain, report on, and verify documentation. Change Management and other ITIL processes use this documentation to make better decisions. Creating and maintaining records of Configuration Items (CI) such as hardware, software, and the documentation related to these CIs is all SACM describes, nothing more and nothing less.

The activities of SACM all relate to the simple idea of creating and maintaining a database of information regarding CIs, and then inserting the usage of this database into the decision making process. This simple understanding is the path to success regarding SACM, and the establishment of that almost mythical of things, the Configuration Management Database, or CMDB.

## The Truth about the CMDB

Many think the CMDB is a single, all-powerful database. They attribute amazing abilities to the CMDB. They grant it capabilities never intended and often not required. Many vendors inadvertently contribute to this confusion with extended claims of functionality based on their unique capabilities.

If I could rename the CMDB, I would call it the Configuration Management Databank. This sounds simple, but a large part of the problem practitioners have with SACM stems from the use of the word database and what people think the word database means. The practitioner reading the ITIL sees database and immediately starts thinking of SQL, server's, administration, and new software licenses.

Even ITIL V3, which rightfully describes a Configuration Management "System," consisting of a number of CMDBs, continues to add to the confusion with its frequent references to what appears to be a singular CMDB.

Webster defines "Database" as "a usually large collection of data organized especially for rapid search and retrieval." Many have forgotten (or never knew) that databases consist of one or more banks or repositories of data. Webster defines "Databank" as an alternate term for "Database." While it is true that a database or databank can be a single entity, databanks are often independent, and often geographically distributed among several repositories.

Many think a database is a single thing, and most think it has to be software, but catalogs and index card systems are also databases. Because most think software (SQL, Oracle, etc.) when they think database the result is confusion and trepidation. These conditions lead to the mistaken belief by practitioners that SACM and the CMDB are more than they really are. Since most IT managers are not "database people" they approach the situation from the wrong angle and try to build the wrong solution for the wrong reason.

In truth, there is no such thing as a "single file" CMDB. If I were to call the CMDB anything, I would call it a type of hypertext engine. In a Hypertext system, any object, whether it describes a piece of data (e.g., text, video, audio) or a physical thing (e.g., hardware, software, person, building, document) can link to any other object. It does not matter where, physically, the object resides. Hypertext systems are particularly useful for organizing large amounts of disparate information. The World Wide Web (WWW) is a hypertext system that is clearly not a single database. The CMDB, like the WWW, is a logical construct that references many sources of information and physical objects.

This is technically called a metabase (sometimes called a metadatabase or metadata repository.) A metabase is a database for storing data that describes data. Stay with me here! For example, a metabase might include metadata about all configuration information regarding a system gathered from a number of sources. A physical metabase is one in which the metadata is actually collected into a single place before it is accessed. A virtual metabase is one in which metadata is gathered on the fly as needed. A CMDB can include both physical and virtual attributes.

So forget the idea of the single, all-encompassing, database. Start thinking about linking together all the collections of information you use in your day-to-day activities. Start thinking about how to reference and relate the repositories of CI information you already have. This is what a CMDB really does.

Now that you understand what a CMDB actually is, I hope you have an idea of how to start uncovering your own Configuration Management Databank!

## Uncovering Your Own CMDB

SACM is the process of managing the information relating to your infrastructure. Once you understand the idea of a CMDB as a metabase, you can start to think about uncovering your own repositories of CI and creating the CMDB. I say uncovering because virtually every single IT organization already has many repositories of CI information. You most likely have a huge amount of SACM data and information already in place - you probably just never thought of it this way before. Commonly, IT organizations have much of the data but lack a single process for managing and using the data. Without such management control, the data is not available to other processes or kept up to date.

Somewhere in your organization there are records regarding hardware and software models, versions, etc. Without a doubt, you have their manuals and software CD-ROMs nearby. Right now, someone has an Excel spreadsheet with records of office system configurations. Finance has an asset register tracking hardware and software purchased. Someone else is using Access to track software licenses. Over there is a shelf holding user manuals, next to that is a closet with spare hardware. Elsewhere there are diagrams of services, circuits, and systems in Visio or maybe PowerPoint.

The first step is to locate all of the sources of information relating to your hardware and software. Do not worry about a single repository for Incident, Problem, Known Error, and Change records, hardware, software, and services for now. (You probably already have a system that tracks Incidents, Problems, and Known Errors or their equivalent right now - your ticketing system, as one of the repositories within your CMDB.)

Focus instead on the CIs relevant to delivering your vital services. Locate the existing information repositories, but leave them in their current form for now. Your goal is not to impose major new projects on existing teams, but rather to locate sources of data and formalize their maintenance and control.

Once you know where your CI information resides, you must plan how to integrate the repositories. The next step is to arrange the sources of CI data into a structure - a metabase. This does not always require investments in new systems or software development. Depending on your size, maturity, and unique requirements, your CMDB solution will vary.

You can create a spreadsheet of where your configuration information resides. You can create relational systems tying together existing tools and systems such as Human Resource directories, ticketing systems, asset registers, log files, and any other appropriate source of data. I have used extremely effective systems based on wildly different methods - from paper "circuit layout records" stored in "tub files," to large-scale software solutions that automatically integrated asset discovery and service monitoring. Each was extremely effective, efficient, and economical for the tasks required.

However you proceed, beware falling down the slippery slope of thinking SACM and the CMDB are huge things you have to purchase or develop. Start small, and grow as you mature. Follow the simple steps of locating the sources of data, taking them under management control, and creating a system for knowing what is where. This simple plan gives you a real CMDB that you can use to realize the benefits of CM!

## Getting Started

Now that you know you can have a CMDB and a SACM process that delivers real value relatively quickly and inexpensively, you need a plan. Here is a plan to get you going.

**Planning** - A SACM process begins with a plan. Your SACM plan needs to include:

- Purpose and scope of SACM
- A description of the CIs and services to which the SACM plan applies
- A time line showing important SACM activities completion
- A description of current and expected SACM tools (e.g., what you have and expect to find)
- Related documentation such as existing SACM plans or plans from suppliers
- A listing of relevant documents and their interrelationships
- Policies describing SACM management activities
- The organization, responsibilities and authorities of relevant interested parties
- Qualifications and training of staff to support SACM
- Criteria for the selection of CIs
- Frequency, distribution, and control of reports

The SACM plan must explain and describe how you plan to achieve each of the following SACM process activities. The ITIL provides great detail on these, so the following are highlights only.

**Identification** - The selection and identification of CIs and their relationships. Identification includes assigning unique identifiers and version numbers to CIs, applying labels to CIs as appropriate, and entering the CI into the appropriate databank. For service-level CIs, the selection of resource-level CIs and the descriptions of their interrelationships should describe the services' structure. Good selection and identification criteria include:

- Regulatory requirements
- Criticality in terms of risks and safety
- New or modified technology
- Interfaces with other CIs
- Procurement conditions
- Support and service considerations

**Control** - Procedures that ensure no change to any CI without controlling documentation such as an updated service or

product specification, or a RFC. In your SACM plan, define how you will accurately update CMDB records. Include:

- Management authorizations and relationships of those in authority
- Procedures for control of changes to CI records within the CMDB
- Methods to communicate changes from physical CIs to their CMDB records

**Status Accounting** - Reporting on changes to CIs throughout their lifecycle. Include methods to track CIs from ordering to depreciation. Unlike Control, Status Accounting seeks to maintain a historical record for the CI. This includes baselines, linked Incident, Problems, Known Errors, etc.

- Methods for collecting, recording, processing and maintaining status accounting records
- Definition of the content and format for all configuration status accounting reports

**Verification and audit** - A series of reviews to verify the presence and configuration of CIs with their respective records within the CMDB. Include in the plan:

- A list of audits planned
- Procedures to be used
- Authorizations required (within and without IT)
- Description of report and expected contents
- Configuration care and feeding

## Summary

Remember, the real purpose of SACM and the CMDB is to control CIs. Initially, use the CMDB to enhance maintenance decisions, expand into using the CMDB to evaluate Requests for Change (RFCs) in support of Change and Release Management.

To support Change Management, identify critical components (CIs) that underpin key systems or applications. Some common examples of these types of CIs include:

- Power supplies
- Shared servers
- Central routers and switches
- High worth users
- Shared applications
- High-speed network connections
- Vital audit, security, and production systems

With the sources of configuration information located, your next step is to combine CI information into services. This requires more work, so start small. Focus on new services, or "problem" services in need of improvement. Start by defining service CIs - composed of these critical infrastructure CIs. This will help stabilize your infrastructure by reducing unexpected consequences of changes.

Following the rest of the Service Asset & Configuration Management process defined in the ITIL allows you to maintain the integrity of the CMDB as you expand it. Establish management objectives to bring other services or locations under SACM control. Perform audits to ensure integrity and find new CIs. Begin to use the CMDB for impact analysis for Changes. Relate Incidents, Problems, and Known Errors to services and specific CIs in the CMDB.

Over time, you can migrate to new systems if required. You will know what you need, and how to achieve it. You might even find it worthwhile to purchase a system - and you will have established the value proposition to justify them.

With your new understanding, and your plan in hand, you are ready to realize the benefits of Configuration Management!