How to Populate Your CMDB in Four Steps

A WHITE PAPER

By Brad Serbu

The secret to a successful CMDB lies in meaningful information

ASG®
Software Solutions

Define the Project Scope

Identify Discovery Data

Build the Information Model

Extract, Transform, & Load

How to Populate Your CMDB in Four Steps

The tire-kicking phase is over for configuration management databases (CMDBs). Spurred by early adopter success stories, which often report an ROI of 200 percent, 400 percent, or more,[1] more than 65 percent of enterprise IT shops are currently in the process of implementing a CMDB solution.[2]

Besides being billed as a must-have for IT Infrastructure Library® (ITIL®) adopters, a CMDB brings an attractive assortment of operational benefits, including improved service levels, reduced downtime, and fewer outages. These cost-saving advantages, along with the high cost of not deploying one, have incited companies to start CMDB projects.

There is, however, a flip side to this sunny scenario. A significant number of organizations have experienced disappointing results from their CMDBs. According to independent research from leading analyst firm Enterprise Management Associates (EMA), 44 percent of enterprises rated the overall success of their CMDB effort as "fair" or "poor."[3] And then there are troubling failure rates. EMA's survey of IT infrastructure and operations leaders at Fortune 500 companies shows that 77 percent of respondents experienced a partial or failed CMDB implementation 18 months after purchase.[4]

These findings have discouraged some CMDB evangelists, but unnecessarily so. Generally speaking, it's not a CMDB's technology that is insufficient, but rather, how it is populated. If the appropriate information model is not built into the CMDB before it is populated, it will fail—simply because it will contain massive amounts of irrelevant information. For CMDBs to be effective, they must be useful; they must be able to access data that answers business questions and solves business problems. CMDBs that are populated with discovery data alone do not function in this way.

Steps to a Successful CMDB Population

1. *Identify the scope of your CMDB project*
2. *Determine the discovery data this is relevant to your project*
3. *Build a business-focused information model for the CMDB*
4. *Extract, transform, and load data into the CMDB*

Prohibitive Misconceptions

Underlying many CMDB disappointments are misconceptions that are likely to misguide implementation efforts.

A CMDB is a "thing." As EMA's Dennis Drogseth emphasizes, the CMDB is "a system in which politics, culture, organization, technology, and products all come together."[5] Organizations assuming that a CMDB is a "thing" run the risk of not investing enough in pre-implementation planning, which is one of the chief reasons EMA identifies for CMDB failures. Instead of properly planning the CMDB project, IT teams frequently become overly focused on the technology.[6] As a result, their expectations for the CMDB's performance typically run high. However, as Drogseth cautions, "You can purchase software that may be strategically useful in creating a CMDB system, but it's not a matter of opening a box…and plugging in a new model train."[7]

The CMDB will solve data quality problems. Many organizations rush to implement a CMDB because they believe that having a single source of record will solve the problem of poor data quality. But even with periodic audits for refreshing data in the CMDB, the information is actionable only when it is 97 percent accurate, according to data management experts. Because change requests are constant, the data in the CMDB is able to maintain this accuracy level for brief time periods, immediately after the audits are complete. After several experiences with misleading data, many users decide that the CMDB is unreliable and stop using it. To avoid this outcome, implementation teams must achieve continuous data quality assurance before, during, and after CMDB implementation.

The CMDB must contain every configuration item (CI). The thought of populating the CMDB overwhelms some businesses because they believe they must store every piece of data in the CMDB. While autodiscovery tools take the proposition of manual population off the table, they, too, can confound professionals with the tremendous amounts of data they can capture.[8] However, not every CI is appropriate for CMDB storage; accordingly, populating the CMDB is not entirely an automatic, tool-driven process. IT organizations must determine which CIs are important to monitor for the business; these CIs belong in the CMDB and should be built into its information model.

ASG®
Software Solutions

Why are CMDBs Unsuccessful?

[M]any CMDB projects fail because the implementation team is too action-oriented, overly focused on the technology, and devotes too little time to properly plan out the effort.[9]

A New Mindset: The CMDB as a Process

The CMDB should be conceived as a process or a journey.[10] Just as the IT environment is organic, so is the CMDB. Embracing this mindset will keep IT teams from trying to do too much, too early, with their CMDB—a misstep that leads to overpopulating the CMDB with configuration data that has no business significance. Burdened by a glut of data, the CMDB cannot be used to evaluate simple configuration changes, such as the feasibility of one department switching from Oracle® to SQL Server®. Organizations that were initially in a hurry to populate their CMDBs find out later how time-consuming it is to organize the precipitating glob of data to find the information they really need.

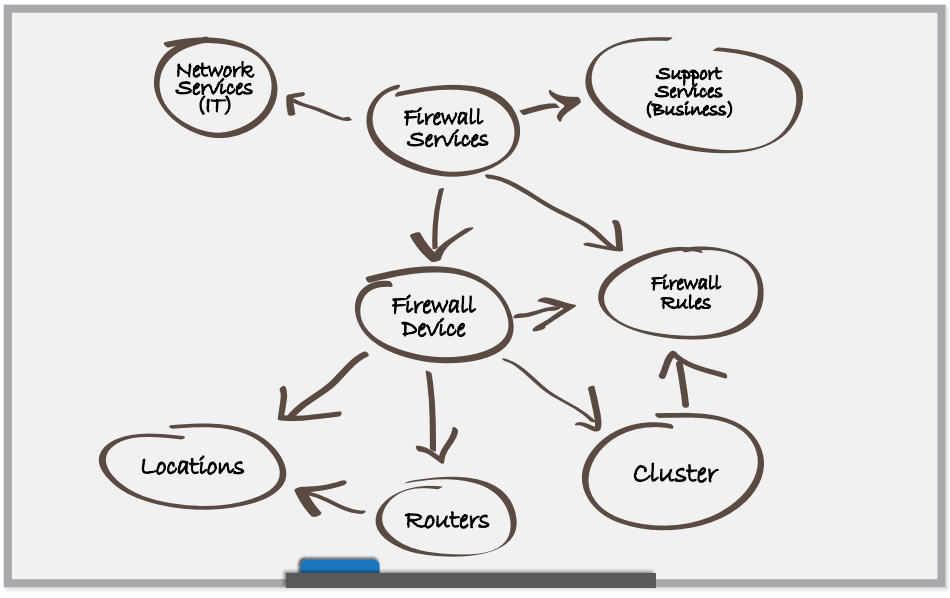
A CMDB's success hinges on having the *right* data. The best way to achieve this is to populate the CMDB gradually, following a series of planned steps that lead to building a business-focused information model. It is this model that enables the CMDB to identify relevant data and ensure the proper process for accessing it.

Four Steps, One Successful CMDB

The CMDB's ability to transform data into meaningful information is secured with its proper population, which can be accomplished in four steps.

Step 1 Define the Project Scope

A frequent problem with CMDB projects is that they are too broad and lack short-term wins—a situation that encourages scope creep and budget overruns.[11] Fortunately, this scenario can be avoided by simply narrowing the scope of expectations attached to the CMDB. Instead of attempting to model every IT component in the enterprise, which results in a purely technical view, adopt a top-down approach that begins with the business. Focus on a critical service, a given domain, or a line of business. Work with the business owners of the project to identify a specific business problem that occurs in one department or during a particular service delivery. Articulate the questions that need answers. Then, formulate objectives for the CMDB project that effectively address the problem. Specify both short- and long-term goals (e.g., six and 12 months or more, respectively).[12]



This type of high-level whiteboard diagram, created in Step One, is extremely useful for IT and business teams because it provides them with a map of the CMDB project that they can mutually refer to as the CMDB project evolves.

After the business problem and project objectives have been identified, loosely sketch how the business owners see the IT infrastructure as it relates to their individual department/service. Maintaining this high-level view, identify the core CIs (such as configuration, asset, and inventory information) necessary for achieving the business-defined project objectives. Next, draw arrows representing relationships among core CIs without delineating their attributes.

This rough sketch defines how the CMDB will be used and what it will manage. By identifying this information, organizations essentially lay the groundwork for the CMDB's information model from a business view.

Step 2 Identify Discovery Data

The goal at this juncture is two-fold: (1) to determine the relevant data sources that hold answers to the business problem articulated in Step One and (2) to identify each data source's structure, format, location, and other information needed to understand the type of adapter needed to access and transfer discovery data into the CMDB.

Start by working with the technical owners of the project to identify the managing and monitoring products (or trusted discovery data sources) that are connected to each core CI identified in Step One. Then, identify the formats of these data sources (e.g., database, log file, XML, CSV, WS, report).

At this stage, it is not important to map how data in the raw sources relates to the business view. Instead, note how each of the products, or data sources, represents and works with the data internally:

- How often is the source updated?
- How many times per day do the technical owners use the information in the product?
- Are there related products used in conjunction with the data source?
- Are there any access or security restrictions?

The answers to these questions, combined with the designation of each data source's format, will result in a basic understanding of the adapter types required to populate the CMDB.

Step 3 Build the Information Model

Next, an information model must be created in the CMDB to ensure that only CIs and relationships that are clearly tied to Step One's business objectives are loaded into the CMDB. Begin by developing a comprehensive topology of the information needed to fulfill the business objectives. Add CIs as necessary, defining CI attributes and the relationships between and among CIs.

This schema, or organizing structure, expands and further refines the business view sketch developed in Step One. For the model to be usable in impact analysis, it should contain CIs that model actual configurations, as well as related CIs that model documented configurations.

Rules for governing the integrity of the data should also be defined and incorporated into the information model to ensure that the relevant configuration data is accurate, consistent, and accessible. These business rules, which may include constraints that specify conditions or relationships that must always be true or must always be false, help identify anomalies in source data so it can be transformed appropriately.

When the CI topology and data integrity rules are completed, the CMDB will have a filter (in the form of an empty model) that can separate the business-relevant information from other, nonessential data.

Step 4 Extract, Transform, and Load (ETL)

The goal of this phase is to access and validate raw data, transferring it to the information model in the CMDB.

Extract

First, work with the technical staff to obtain, configure, or build an adapter to connect to the specific format of each data source. Be sure to identify any changes (such as added monitoring products, new or different data formats, etc.) that have occurred since the completion of Step Two.

Use the newly constructed adapters to read, copy, and move the raw data into a transient data source (or staging area) for processing.

Transform

Use the business rules that were defined in Step Three to map the raw data to the information model. Keep in mind that one raw data source can become the information source for several CIs or CI types in the CMDB's information model, as in the case of a single location functioning as an attribute of multiple firewalls and routers.

Next, perform the data transformation, using a processing engine to convert the raw data from its original state into the form required to move it from the staging area and into the CMDB. This may involve reformatting and/or cleansing the data to remove duplicates and inconsistencies.

Load

Finally, connect adapters to the processing engine, transporting and loading the transformed data results into the CMDB.

When Step Four is concluded, the raw configuration data will have business significance because it has been filtered through the information model. Both IT and business owners can then use the CMDB to determine how systems are actually configured in the field and compare these facts to the service definitions that describe how the systems *should be* configured.

The Desired Outcome

The result of a properly populated CMDB is one validated, integrated source of truth that decision makers throughout an organization can use with confidence.

The Impact of CMDB Selection

While following the four main steps outlined in this paper, organizations can accelerate their project's time-to-results by ensuring their CMDB has the following characteristics:

A metadata repository. This database model is more flexible than a relational database, allowing for adjustments (e.g., software upgrades) without completely rebuilding the core model.

Federation. A federated CMDB can access data from any source, regardless of its vendor or location; with federation, no piece of data is inaccessible for population.

Autodiscovery. This capability, which replaces manual dependency mapping, ensures that only the most accurate data is loaded into the CMDB.

Dependency mapping. A CMDB that can automatically draw relationships between CIs saves substantial time and labor when building the CMDB's information model.

ASG's metaCMDB™

ASG's metaCMDB™ possesses all of the above attributes and more. Unlike other CMDBs, it already contains the tools necessary to support each step in the population process. With ASG's metaCMDB, CMDB population becomes intuitive.

Through the built-in intelligence of ASG-Rochade®, the world's leading metadata repository, ASG's metaCMDB automatically discovers and maps relationships between CIs (identified and unidentified) in any direction. This, and the power of federated technology, makes it easier to locate and identify all CIs that are tied to any given business objective.

Most importantly, ASG's metaCMDB transforms configuration data into meaningful information. Rather than starting with a pre-built data model, which often results in loading immaterial information into the CMDB, ASG's metaCMDB's model is completely customizable. It can be tailored to business needs on the front end, eliminating the costly time and effort of undoing an ineffective CMDB population and rebuilding the data model.

The information model of ASG's metaCMDB allows the CMDB to be managed as a process. It is flexible; it does not require organizations to define what they want to know in advance. So, when the business's information requirements change, the information model in ASG's metaCMDB[13] can be easily altered without rebuilding numerous tables, as would be the case with a relational database. Accordingly, ASG's metaCMDB can adapt to IT's changing landscape, accommodating shifts in business priorities and evolving information demands.

ASG's metaCMDB

- Acts as master or "citizen" CMDB in a federated system of CMDBs and trusted data sources
- Discovers CIs automatically and dynamically
- Initiates data federation and reconciliation automatically
- Synchronizes the master and federated CMDBs
- Provides the store of record for configuration items, including their attributes and relationships
- Enables the simple classification and organization of configuration items
- Provides root cause analysis to any desired depth
- Supports impact analysis from any desired seed item
- Offers customizable configuration views and queries
- Provides data integration between diverse vendor technologies
- Supports ITIL® disciplines
- Provides an out-of-the-box Federated Portal for easy intranet access to configuration information

Conclusion

Successful CMDBs are deployed by IT and business teams that focus on managing the implementation process, rather than on activating the CMDB's technology. A critical part of this approach involves populating the CMDB incrementally, following definitive steps.

Adopting a top-down approach clarifies the business value of the CMDB at the outset. By building the information model from the business view down (as opposed to starting with the perspective of the help desk), IT teams can ensure that all data sources for the CMDB are trusted data sources, which largely removes data quality concerns from the CMDB project.

If organizations start small, and do it right, the gains achieved with their initial CMDB population will serve as a foundation for optimal data management results in other departments and other service areas.

About the Author

Brad Serbu, Vice President of Research and Strategy at ASG Software Solutions, Inc., oversees the corporation's development of enterprise-level Business Service Management (BSM) solutions. Serbu, who has been with ASG since 1999, has been instrumental in engineering many of ASG's benchmark software releases, including ASG-IMPACT® Web Enterprise, which incorporates ITIL® and other service management best practices. Serbu has also led the development of ASG's signature strategy for implementing and deploying its federated metadata repository, ASG's metaCMDB™.

Notes

(1) *A Board Room View: Understanding Your CMDB Project's ROI*. Boulder: Enterprise Management Associates, Inc., July 2008.

(2) Matney, Chris. "CMDB Requirements: The Hidden Killer." *EMA.com*. EMA Analyst's Corner. June 2008.

(3) Ibid.

(4) *A Board Room View*… Enterprise Management Associates, Inc.

(5) Drogseth, Dennis. "One more time: Why the CMDB is not a 'thing.'" *NetworkWorld.com* 04 Dec. 2006 <http://www.networkworld.com/newsletters/nsm/2006/1204nsm1.html>.

(6) *10 Best-Practice Tips to Help You Succeed with Your Enterprise CMDB Project*. Boulder: Enterprise Management Associates, Inc., June 2008.

(7) "One more time: Why the CMDB is not a 'thing.'"

(8) EMA cites autodiscovery and CMDB population as two of the most daunting challenges facing implementation teams. (*10 Best-Practice Tips*…)

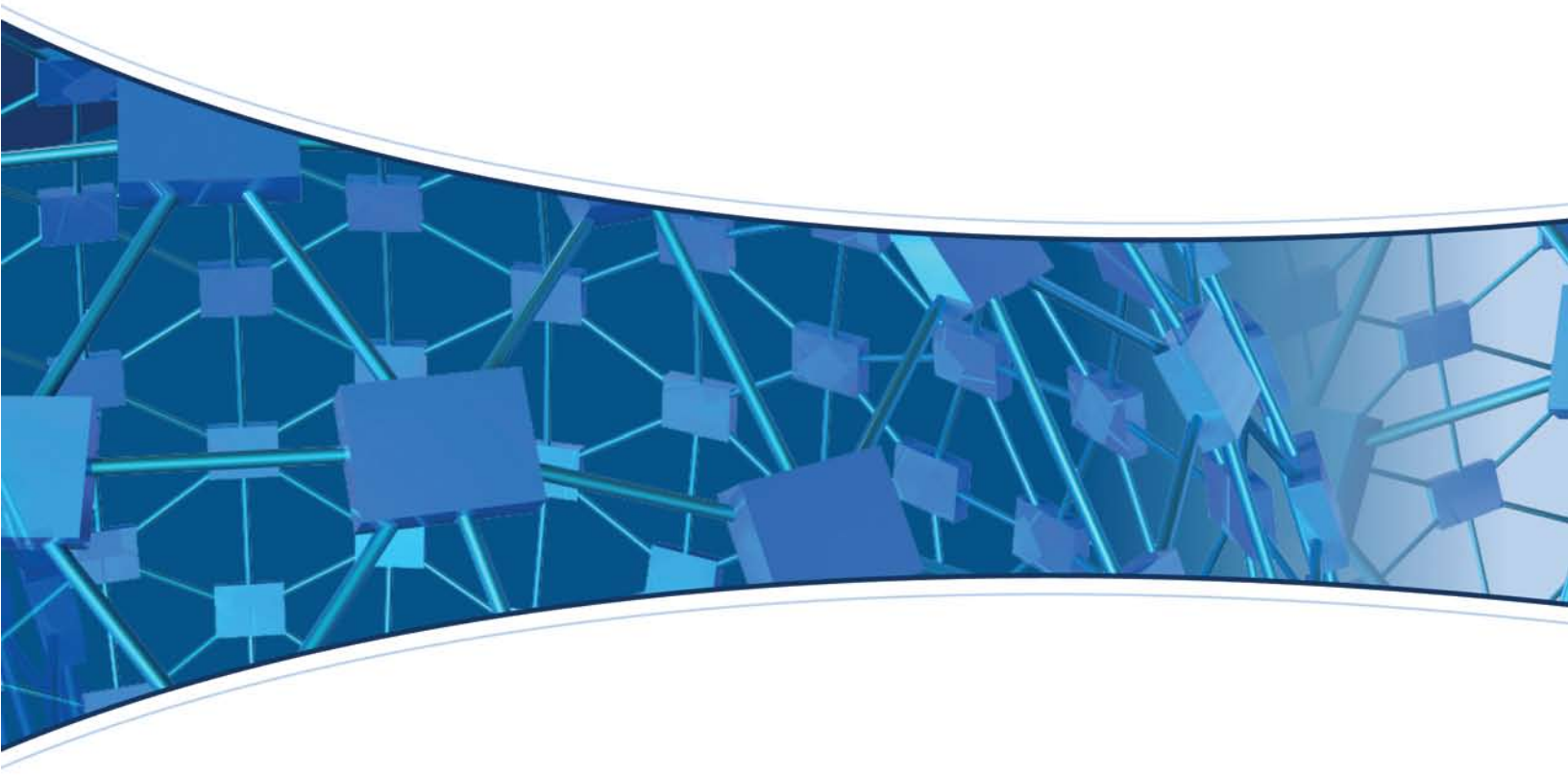
(9) *10 Best-Practice Tips*… Enterprise Management Associates, Inc.

(10) Muirhead, Richard. "CMDB: A Journey, Not a Destination." *ITJungle.com*. 18 March 2008.

(11) *10 Best-Practice Tips*… Enterprise Management Associates, Inc.

(12) Guidelines for writing detailed requirements for CMDB projects are presented in EMA's hands-on workbook, *How to Define Detailed Requirements for Your Enterprise CMDB Project* (April 2008).

(13) The meta-model of ASG's metaCMDB combines the best capabilities of network and object-oriented database models. (See Connor, John. *Choosing the Right CMDB: Smart Considerations for Strategic Decision Makers*. Naples: ASG Software Solutions, July 2008. Available at *www.asg.com*.)



ASG Worldwide Headquarters | 1333 Third Avenue South, Naples Florida USA 34102
Telephone 1.239.435.2200 or 1.800.932.5536

Business Service Portfolio | Content Management | Metadata Management | Applications Management | Operations Management | Performance Management | Infrastructure Management | Identity Management