The State of CMDB Federation

Jayson Kurisinkal, June 2015

Abstract

While *federation* is a great concept in Configuration Management Systems (CMS) architecture as defined in ITIL v3, it is not widely adopted or implemented. Once the basic CMDB is set up and made operational, perhaps the next generation of CMSs may evolve into a *federated* model. But why has this taken so long, and is it worth evolving into that state?

Background

Federation as a concept in CMDB was introduced in ITIL V3, when the term Configuration Management System (CMS) was coined. At that time, enterprises were struggling to implement CMDB and practical wisdom perhaps drove the ITIL authoring team to propose CMS, a distributed CMDB/data repositories concept. CMS architecture also made sense as it was conceivably the best way to get enterprises closer to implementing a CMDB: Do not build a monolithic CMDB that acts like a data warehouse, let the existing data repositories be; you only refer to the data in them when required, from your CMDB – it all made logical sense!

Soon after ITIL v3 was published, some of the leading ITSM tool vendors formed a consortium to establish a set of standards (CMDBf) for exchanging and referencing data between CMDB and Master Data Repositories (MDR), and among MDRs. There was no plan for having a standard data model, though. Naturally, the tool vendors who were part of the consortium added the catch phrase *CMDBf Standards* to their product flyers.

This was in 2007. We are in 2015 and let's take a look at what's happening now.

Current State

IT Organisations

Many IT Organisations have been successful in implementing a CMDB – Configuration Items (CI) have been uploaded in various classes, and relationships have been defined to form a Service or Business model, which has provided valuable "impact assessment" information, among other things. This data is maintained through periodic discoveries and automated or manual data uploads (mostly through ETL – extract, transfer and load – jobs). Also, the data is normalised and reconciled on an ongoing basis.

Even after the introduction of the CMS concept, based on the results from a survey, and from the interactions this author had with other CMDB architects, it appears that most organisations have adopted a monolithic CMDB!

Feedback on the state of Federation and its future

A survey conducted on CMDB architects from around 10 enterprises, who have implemented a working CMDB in the last 5-7 years, gave the following results:

Survey respondents' background information:

- ✓ Respondents used 4 of the market leading tools
- ✓ Half of the respondents had <10 classes in their CMDB and half of them had >25
- ✓ 25% had less than 10K records in their CMDB, 40% had between 10K and 25K, and 50% had more than 250K
- ✓ One fourth of respondents had fully automated data maintenance, half had automation of data maintenance for more than 50% of CIs

Survey Results

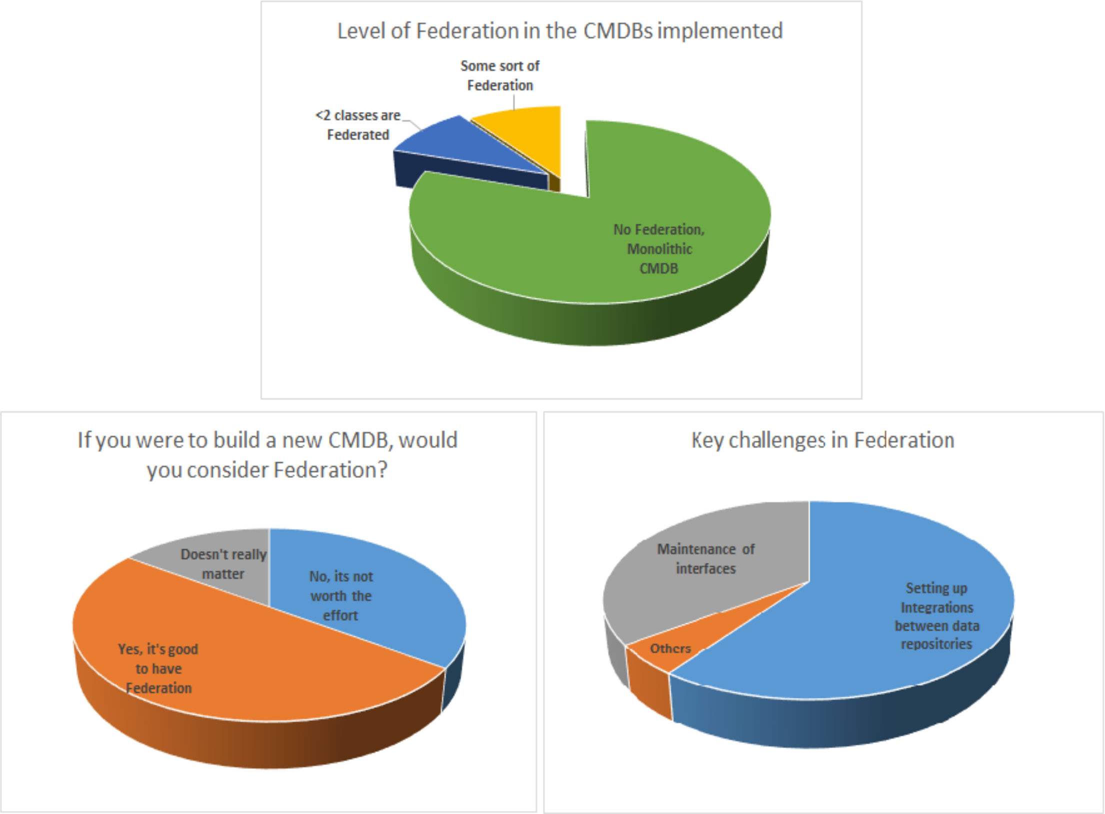

Table 1

CMDBf consortium

After publishing a few papers on the architecture and specifications, this body has now become defunct. Some of the leading ITSM tool vendors who were part of the consortium have successfully demonstrated interoperation of CMDBs by using CMDBf specifications. The standards specs are still in draft mode on the DTMF site and have no signs of revival.

Vendors

CMDB tool vendors, even those who were involved in CMDBf, obviously do not refer to the CMDBf standards anymore. Federation as a concept is by and large mentioned by most of the product vendors.

ITSM Tool Vendors' current stance on Federation

The following information was collated in June 2015 based on the publicly available documentation and references on the vendors' website. This is not intended to be a representation of the vendor's CMDB tool capability or the capability for CMDB federation. (Listed in alphabetical order.)

BMC Software

In their CMDB product documentation, you can find the following description:

> *"Federation provides data with service context across multi-vendor IT management environments, so you can take actions by automatically launching applications, such as service desk, to create a ticket."*

Also in an article titled "Six Criteria for an effective CMDB architecture", BMC lists *federation* as the number one criterion.

CA Technologies

The CMDB Product information does not mention federation, however a whitepaper published by CA in 2014 titled "IT Service modelling for CA CMDB" talks about federation, and even the CMDBf standard.

Cherwell

In the CMDB features and capabilities listing, there is no mention of federation.

HP

The product documentation states that HP Universal CMDB supports *vendor capable federations*.

ServiceNow

The product documentation of Enterprise CMDB lists federation as a feature:

> *Federation of third party discovery and configuration data is supported through standard synchronization offerings (SMS, LAN Desk, others) and through the CMDB Discovery table*.

SerivceNow has also started using a new phrase "Single CMDB", especially after the Knowledge 15 conference – this is not a reference to a monolithic CMDB, but having a single CMDB across the enterprise (IT, HR, Business etc.).

Table 2

As you can see from Table 2, the term *Federation* means different things for different CMDB tool vendors. This situation is aptly captured in the Gartner Critical Capabilities Study of CMDB (2014), where one of the key findings was that "*The meaning of the terms "integration" and "federation" varies significantly among vendors*"!

Why Federation is not popular

Federation in its true sense is to reference data or meta-data from another data repository by establishing a real time connectivity from the CMDB to the data repository. But why do organisations still get all the data from these repositories and upload them through data upload and maintain a monolithic CMDB? The possible reasons are discussed below.

Evolution of CMDB and Service Models based on business and operational value

One of the advantages of having federation is that we need not manage large volume of data. But how much data are we talking about?

In the early days, a major concern for all architects was the thought of an overwhelming amount of data to be populated into CMDB. Everyone was advised to take baby steps, which in fact is still relevant. Someone reading the CMS and Service Knowledge Management System (SKMS) architecture of ITIL v3 is sure to be overwhelmed. However, in reality, the CI records or classes of CIs that really need to be tracked or managed to deliver an IT service *without significant risks* are much lower than you would expect. The keyword is *risk*; the cost of managing the CI should not exceed the cost of risk (ITIL ST, 4.3.5.3).

Also, the number of classes or types of CIs that you define drives your need for federation – what level of details should you fetch for a CI. Certain details of CIs are best left as an attribute rather than splitting then into further CIs. For example, the physical memory of a server is best described as an attribute rather than having a CI class of its own. The advice from experts (as per ITIL) during the early days and the way tools were designed then, caused much time and effort to be spent on CI-ising everything!

Evolution of "as a service models"

While discussing benefits of federation in ITIL v3, one of the examples given (ITIL ST, 4.3.3) is that of the need to integrate or federate with the outsourced service provider's CMDB.

… [The organisation's CMS] system will derive its data input from discrete repositories – each one a CMDB – owned and maintained by the [Service providers]…

In today's "as a service" world, this is irrelevant as customers are not worried about how providers manage their internal CIs to deliver the service they are offering, as long as the utility and warranty are met. Organisations who subscribe to such outsourced services will, at the most create a Service level CI in their CMDB, with appropriate relationships to other services. These CIs will be used for incidents (ideally generated as events from the service provider or through the organisation's monitoring and event management tools) and change request's impact assessments, among other operational values of CMDB.

Hence "as a service" models avoid the need to set up federation to other service provider CMDBs.

The need for data accuracy

ITIL lists more than 10 items as value enablers of CMDB to business (ITIL ST 4.3.3). However, perhaps only two or three of these are practically visible in organisations:

- *Successful assessment, planning and delivery of changes and releases*
- *Resolution of incidents and problems within the service level targets*

Both the above points somewhat describes the value of CMDB as an "impact assessment" tool. How accurate does the data need to be for this functionality?

In the book *The IT Skeptic looks at the CMDB* [2009], Rob England proposes that for most organisations, configuration data is best, among other things, *impact assessed by expert people, not technology.* Though a lot has changed since this was written, the core message is still relevant.

ITIL defines one of the objectives of SACM as *Maintain accurate configuration information on the historical, planned and current state of services and other CIs*. (4.3.1)

Monolithic CMDBs with periodic (daily or weekly) data loads and reconciliation may not have accurate data all the time, but it is a good starting point for a change or incident's impact assessment. Obviously, impact assessment for an incident needs more accurate information. In reality, when an incident occurs, it is still beneficial for the first level support to look at the impact information available in the CMDB to get a high-level assessment done.

Let's take the example of a physical node failure in a virtual server farm. Since the last update to the CMDB, a VM could have failed over to another node. However, the first level of assessment from CMDB might not give this information which has to be obtained from the management software of virtual servers. This is usually done by the relevant support team ("*expert people"*), and the impact is formalised and communicated. This is pretty much what is happening in most organisations with a monolithic CMDB, which in most cases meets business requirements.

However, with more automation and orchestration of incident resolution and service provisioning, the need to have accurate information has increased. In such cases, the automation tool directly references the data repository to get the accurate information (in case the CMDB does not have the current state).

There is also a penetration of Robotic Process Automation in IT operations recently, which literally mimics the actions of an IT Support analyst and executes high volume, repetitive activities. This could include Service Request fulfilment and recurring incident restoration. As these are pre-defined activities that require no decision making, the dependency on accurate information in the CMDB can be avoided during the creation of scripts, by pointing it to a reliable source.

Federation through launching the interface of the data repository

In cases where building a federated interface is difficult, a link (URL, an executable, or even plain instructions on how to access) is provided as an attribute to a CI. This will enable the consumer of a CI to launch an interface of the data repository that has the accurate information or more details. For example, for an Oracle Database CI, a link to the Oracle Enterprise Manager interface is given as an attribute and the user can launch the OEM interface to find the most current state of the database configuration during incident restoration or the change record creation process.

Technology challenges and business case

As illustrated in the survey results in Table 1, the major challenge of implementing federation is the technical complexity and the overhead in maintenance. Though most of the data repositories support Web services, JDBC connectors, Database layer connectors or a similar technology, the cost in designing and implementing federated interfaces between CMDB and data repositories, and between repositories does not often justify the value it provides. Also there is a challenge and cost in maintaining these interfaces.

Conclusion

Implementing CMDB Federation is not unachievable. Leading tools in the market provide the option of implementing federation, though not always in a standard way. Most organisations are satisfied with the extract, transfer and load (ETL) process of updating a single CMDB rather than having real time accuracy through federation to data repositories where accurate information and more details of CIs are stored. Changes in the service models of outsourced providers (“as a service”) also have reduced the need for federation.

References

ITIL Service Lifecycle books, AXELOS, 2011
www.axelos.com

The CMDB imperative, Glen O’Donnell and Carlos Casanova, 2009

DTMF CMDB Specification, 2009
http://www.dmtf.org/sites/default/files/standards/documents/DSP0252_1.0.0.pdf

BMC Atrium CMDB product data sheet, BMC Software
http://documents.bmc.com/products/documents/01/68/200168/200168.pdf

Six Criteria for an effective CMDB architecture, BMC Software, 2011
http://documents.bmc.com/products/documents/66/58/186658/186658.pdf

IT Service modelling for CA CMDB, CA Technologies, 2014
https://support.ca.com/phpdocs/0/8165/8165_whitepaper_IT_Serv_Modeling_for_the_CA_CMDB_v6.pdf

CMDB Features and Capabilities, Cherwell Support site
https://cherwellsupport.com/WebHelp/en/5.0/15854.htm

HP ITSM product information
http://www8.hp.com/au/en/software-solutions/configuration-management-system-database/

ServiceNow product information page
http://wiki.servicenow.com/index.php?title=Enterprise_CMDB#Federation&gsc.tab=0

CMDBf: IT's Next Moonwalk, Dennis Drogseth, 2009
http://www.cioupdate.com/trends/article.php/3831831/CMDBf-ITs-Next-Moonwalk.htm

The IT Skpetic Blog
http://www.itskeptic.org/blog

Critical Capabilities for Configuration Management Database, Ronni J. Colville, Jarod Greene, 2014
http://www.gartner.com/technology/reprints.do?id=1-1XERE7B&ct=140715&st=sb (Link from HP site)