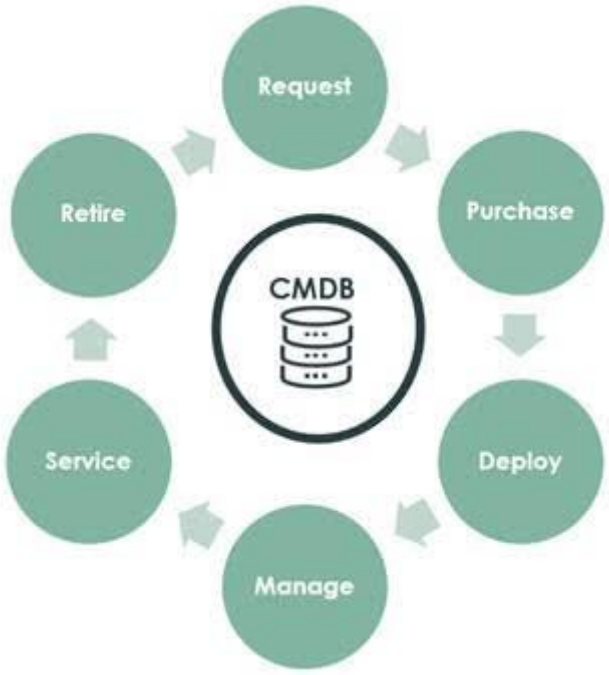
Using SAM to Create & Maintain your CMDB

By Rory Canavan, SAM Charter

Introduction

SAM Charter has been a principle advocate of companies creating and maintaining Software Asset Management Systems to manage software – indeed, our CEO (Rory Canavan) lobbied Working Group 21 to consider making ISO 19770-1 a Management Systems Standard; thereby making alignment to other Management System ISOs that little bit easier to achieve – and it was greeted with some delight to see this achieved in November 2017.

However, once we have built our IT/ Software Management System, aside from generating Effective Licence Positions, (ELPs) we also have to consider how we work and play well with other disciplines in the IT operational lifecycle. Service Management is the acceptable face of IT as far as the business is concerned, particularly in the managed portion of the life cycle. Core to that face appearing relaxed and in control, is the creation and maintenance of a CMDB (Configuration Management Database) which the IT life cycle orbits around.



This though, presents Service Management with a problem: What strategy do we adopt to ensure our CMDB remains accurate and scalable to our needs? Perhaps (just perhaps) SAM has the answer.

The problems we are trying to fix

The Bermuda Triangle of SAM

To qualify the problem that lies before us, let's consider a use-case of bad practice we at SAM Charter like to refer to as "The Bermuda Triangle of SAM". A project manager is tasked with standing up a new service for a business, and that service is going to make the business a lot of money - The service is supported by a technology stack, comprised of hardware and software.

Consequently, our project manager doesn't wish to be weighed down by standard operating procedures and such "out-dated" concepts as due-diligence, and so circumvents any existing IT request routine and goes straight to procurement with their budget code.

Procurement (having few checks and balances in place) are really only concerned that the project manager does not exceed their budget. So, after a cursory check, executes the purchase and our project manager is happy – i.e. The purchase phase of IT life cycle is wilfully circumvented.

In tandem with going through the procurement activity, the project manager will approach the team responsible for deploying software and ask them to configure and roll out the software they are seeking to buy. With them taking the Project Manager on trust (and not wishing to blow their deployment SLAs) they oblige, not giving any regard to assessing whether the entitlement is in place to justify the installation. The provisioning via the IT change procedures portion of the IT life cycle is once more wilfully circumvented.

Fast-forward six months, and the project manager is on a new project, procurement can't remember what they bought last week, never mind six months ago, and the deployment team are in the same boat. And then a software vendor knocks on the door and asks you to demonstrate what you have bought compared to what you have installed.

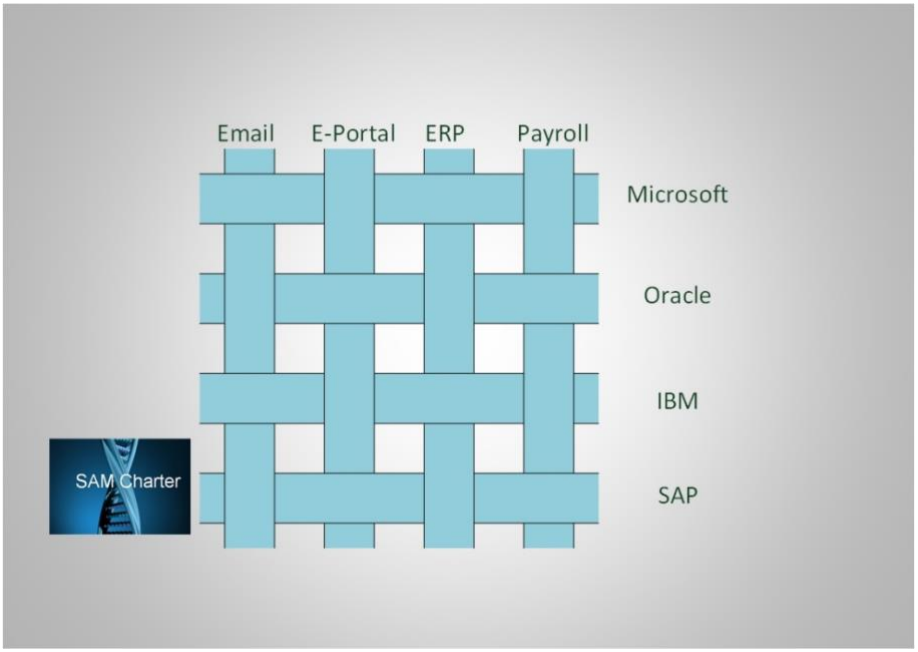
The air-gaps that exist between the request, purchase and deployment activities will set you up for the perfect storm of a software vendor audit. No CMDB means a skewed visibility into compliance health, which means there is no mechanism to quantify the problem or remediate the situation. The company thus finds themselves in the Bermuda Triangle of SAM, unable to put faith in the data they are presented with.

For a 3-minute video characterising this use-case, please go to:

https://www.samcharter.com/awesome-explainer-video/

SAM and Your Technology Weave

Picture a piece of cloth and imagine this represents your IT estate. SAM will view the IT estate from an East to West point of view, with each strand representing a software vendor – at the end of each strand, if you are able to perform a comparison calculation of contract and purchase data against installation data, then you have a fair chance of arriving at an ELP. The business though, views IT in a North to South fashion – i.e. representative of the IT services that interface with the business: e.g. Messaging, ERP or CRM. Once an ITAM function is capable of generating ELPs in a systematic fashion, it should then start looking to report on the cost of the technology stacks that are comprised of software and hardware.



➔ *Sidebar: I appreciate that this is a simplification of what a technology-stack could or should look like; I would always look to bench-mark a technology stack against the ISO 7-layer reference model, so that all components are captured and accounted for – however, for the purposes of this article, we'll be keeping our attention on software.*

For a fuller dissection of this viewpoint, please go to the link below:

https://www.samcharter.com/sam-understanding-technology-weave/

So: keeping in mind that we have a trestle-like view of our IT estate, how can we reasonably set about managing the integrity of both perspectives, (Service Management & SAM) without harming one another?

We need a CMDB initiative that enables a company to capture software assets coming into an organisation from the point of inception through its lifecycle to the point of disposition. Leveraging the CMDB, companies can gain visibility to their Technology Weave. SAM cleanses CMDB data, which can be surfaced in places like an application portfolio management (APM) solution that also leverages the CMDB. An application portfolio solution can be used to view critical business services, the applications that support those business services and the software that comprises those applications.

If we can apply the various use-cases before Service Management that a CMDB enabled request portal should address, then we stand a much better hope of curtailing wild-west purchases and deployments, and having to re-label them as Shadow-IT which permits bad practice to be branded as the new-normal.

➔ *Sidebar: My problems with Shadow-IT are:*

1. *Shadow IT doesn't play to economies of scale (purchases won't typically be made under a volume contract)*
2. *Shadow IT will not magically add itself to any CMDB once it has been deployed, though solutions for spend detection are starting to find their way into the market*
3. *The first time Shadow IT is introduced to the business is (invariably) when it stops working properly; Service Management is always playing catch-up understanding what purpose the Shadow IT is serving*
4. *This is how bad projects are run!*

Our IT request portal should accommodate:

- Software Requests
- Requests made by end users
- Requests made by program/ project managers
- Manager approval capability for specialty software
- Auto-deployment for titles deemed safe/ low on the SAM radar
- Other best-practice checks (IT approval, Licence Pool check)

Top tip: This entire paper is based on the understanding that ***ALL*** personnel requesting software uses the portal to request those assets. Without it, the business will not have visibility of demand, and so a primary step in the Service Management and IT Asset Management lifecycle is broken.

Like any chain, your lifecycles are only as strong as their weakest links.

What does our IT request portal need to capture to make this notion different from any other?

First and foremost, we should be making use of and creating a few primary keys:

- The primary key for the request
- A primary key for the IT service we are aligning our software to
- A primary key for the business service we are aligning our software to

Ideally, our IT request portal is backed up with a Supported Software Catalog that sits within a CMDB, guiding users to pick tested and approved software. If something is requested that is not on the list, then we should offer the requestee the chance to get newly-requested software to be added to the Catalog. If you wish to split views for the request portal so that end users see something different and easier to use than the view presented to Project Managers, then that could be a great idea if done well.

Why all these primary keys? The information needs to be captured so that we can install a software tag on/ alongside the software assets to be deployed. ISO19770-2 (Software ID - SWID tags) allows for the customisation of such data. If such information is embedded into the installations of software, then SAM suites that support -2 tagging will be able to highlight which software technologies support which business or IT services.

XML tags constructed as per the guidance offered by ISO 19770-2 are machine (and human) readable pieces of data that uniquely identify software installations. Adoption by major software vendors is "patchy" to say the least, however you can create your own tags, and one of the best places to start in seeing how to do this would be via the Tag Vault website.

For those titles that are not a physical install on someone's device, then the tag placement could be re-allocated to a system such as Active Directory or the CMDB itself. Whichever system is used, ensure that it can offer this data into your SAM suite. This then places the CMDB (or Active Directory) at the heart of your Joiners, Movers and Leavers Process as a golden source of truth, and one that should interact with your HR function.

The Benefits

The approach above does not come without some work; however, a "do nothing" approach will be costly to your IT estate.

What are the benefits of following the above-mentioned approach?

Building & Maintaining your CMDB: Over time, once the request funnel has been established and adhered to by all staff, then you will start to see the creation (and maintenance) of your CMDB (at least from a software point of view). If you are feeling bold, you could expand your software request portal to include other elements of IT, but for now, your software stack for IT services is nothing more than a report away – and is self-sustaining with perpetually clean data.

Value-Chain Analysis: Providing financial data is aligned to software installs, then running the "IT service" query in the SAM Suite (where IT Service = "X") will automatically calculate the cost of the IT service. From here, the business is in a much better position to calculate the cost of a technology-stack against a business service – so that that cost of that service can be compared to the revenue it generates.

Following on from that idea:

Scaling up and down: Calculating the upscale or downscale of IT services will be made that much easier, as the existing technology will be tagged, and aligned, to pricing data. The blending of contract terms and conditions against proportion of usage will offer a greater deal of certainty for any intended changes to the existing architecture.

Compliance: With ever-increasing complexity concerning the context in which software is used (as opposed to one install = one licence) tags that align major software titles will offer a richer comprehension of the manner in which software is being used. This is never more apparent in the case of OEM/ Embedded software. A database or report package might be included as part of a larger suite of software in a given technology stack; and so be licensed as part of that larger suite.

However, had we adopted an "East-to-West" view of our IT estate, we could be sending the company scrambling to apportion a licence to something we have already paid for.

Another aspect that has recently come into focus in the world of SAM, is indirect licensing – granting users of one system to features and functionality of another system. The point at which access is granted from the first system to a second system is the point at which both systems should share the same IT service tag or have appropriate access rights allocation accounted for in the CMDB.

This, aligned with any bridging/ connector technology that is deployed will offer a ready flag to the SAM Team to scrutinize whether or not existing licenses permit this technological use-case.

Technology not associated with a Service: This approach to managing your IT estate from a SAM & ITSM perspective is evolutionary – not revolutionary; and will require the lifecycle of IT change to churn through your IT estate, but what you will see over time is the steady building of the IT services like pieces of pie re-assembling on the same plate.

The point at which you believe you have all of your IT services tagged is when you can start to interrogate the remainder of your IT estate for rogue installations or blacklist candidates. Think about it: if it's not tagged, what purpose is it serving? A mop-up exercise of tagging legacy IT might be required, but equally, a rationalisation of the IT estate can also ensue with an application portfolio tool mentioned above.

Conclusion

Hopefully the contents above will have offered you food for thought on how best to implement scalable and consistent SAM, while leveraging a CMDB to facilitate the IT lifecycle. Stood alone, disciplines like the discussed SAM, APM, ITSM and HR will get so far, but combined, they make for a powerful and compelling direction in which to take IT operations.

To discuss your CMDB and SAM alignment, reach out to SAM Charter today:

Email: rory.canavan@samcharter.com

LinkedIn: https://www.linkedin.com/in/rorycanavan/

About SAM Charter

With a technical background in business and systems analysis, Rory has a wide range of first-hand experience advising numerous companies and organisations on the best practices and principles pertaining to software & IT asset management. This experience has been gained in both military and civil organisations, including the Royal Navy, Compaq, HP, the Federation Against Software Theft (FAST) and several software vendors. Rory is the Research & Development Director for ITAMOrg, and a member of WG21 – the working group responsible for defining and promoting the ISO 19770 family of ITAM standards.