Automated Mapping of Services From CMDBs

David Cuthbertson, AssetGen Ltd

Mapping services and systems can be difficult and time consuming due to the complexity of modern IT environments, so techniques that reduce the workload involved and re-use the maps for other purposes are in demand. This white paper covers how CMDB (Configuration Management Data Base) data can be more easily captured and used more effectively for service management using AssetGen SysMap.

The Value of Mapping Infrastructure

A service map makes it much clearer than a list of how services, applications and IT components combine together to deliver a service. The same applies to any form of mapping technique – you create a map to enable comprehension so that decisions, actions and risks are better understood. It could be a network diagram, a service map, a data flow chart, a batch process diagram and so on. A city A-Z atlas shows how we have data in list forms (the indexes) linked to maps that show varying degrees of detail, often overlaid with services (tube, theatres, hospitals, congestion zone, etc). The maps are used by the public for planning journeys, reacting to incidents, determining journey times, knowing alternative routes, etc. It is a reference source for utilities, government and emergency services to plan activities, coordinating tasks and projects. Using the same approach for mapping IT services and systems seems like common sense – it saves everyone time, reduces risk and enables management processes – it could even be best practice! Our white paper "Mapping Services, Systems and Servers using AssetGen SysMap" has more specific detail around the benefits of service mapping (available on our web site).

Prepare for Success

If we continue with the A-Z analogy, key success factors that enable delivery of a city street map are shown in the table below. The first issue to notice is that until there is ownership, there will be no common source of data and no agreed update process. Hence the reason within service management for establishing a configuration management (CM) function to set policy, develop and manage processes, standards and conventions.

If you are thinking about setting up a configuration management function, evaluating service desk toolsets for CM, creating project plans to capture data on services, or developing CM interfaces to technical teams then hopefully we can give a tangible vision of the CMDB service mapping output as users might see and use it.

Success Factors	City Mapping	Service Mapping
Ownership of a common data & update process	Government	CM
Ownership of standard and custom output lists and maps	A-Z company	CM / reporting / users
Common terminology and symbols	A-Z company	CM
Production of specific maps with overlays	A-Z company	CM/ reporting / users
Used for planning & operational processes, risk control	Public, transport, leisure, health, emergency etc.	Projects, change, incident, problems, continuity, capacity, SLA monitoring

As all managers know, it is easier to implement strategic change if you have a "vision" which everyone understands. In our projects to deliver service mapping, we always focus on the development of a prototype or demonstrator (normally within 3-5 days), as it is a useful tool to communicate what service mapping can deliver and what it looks like. It reduces the amount of organisation time devoted to education, requirements gathering, and objective setting.

For the purposes of this white paper we will assume that you already have assigned ownership or governance, or intend to do so in the near future. We'll concentrate on the mechanics of capturing, producing and re-using service maps in conjunction with CMDB data held in a service desk.

The Existing Service Mapping Process

One of the first issues you face when starting a mapping project is how you get the data that describes the relationships between services and systems. To get an end to end view across technologies will often involve specialists agreeing what is a good perspective. Typically, a white board is used to capture information, with a few iterations to get the accepted "service view". From this we produce a picture and a 'parts' list to ensure that the Configuration Items (CIs) are in the CMDB and we manually connect them with the appropriate relationships. In the case of asset and other simpler data, we may be able to create the relationships direct from another source that already exists. Then you need to validate the dependencies are correct by manually checking a sample of the CMDB contents. Once agreed, we now have a service view in the CMDB!

Responsibility can then be assigned to a service owner to keep it up to date and review on a regular basis. In theory, the service owner can use the original diagram to check for accuracy and update it in line with changes.
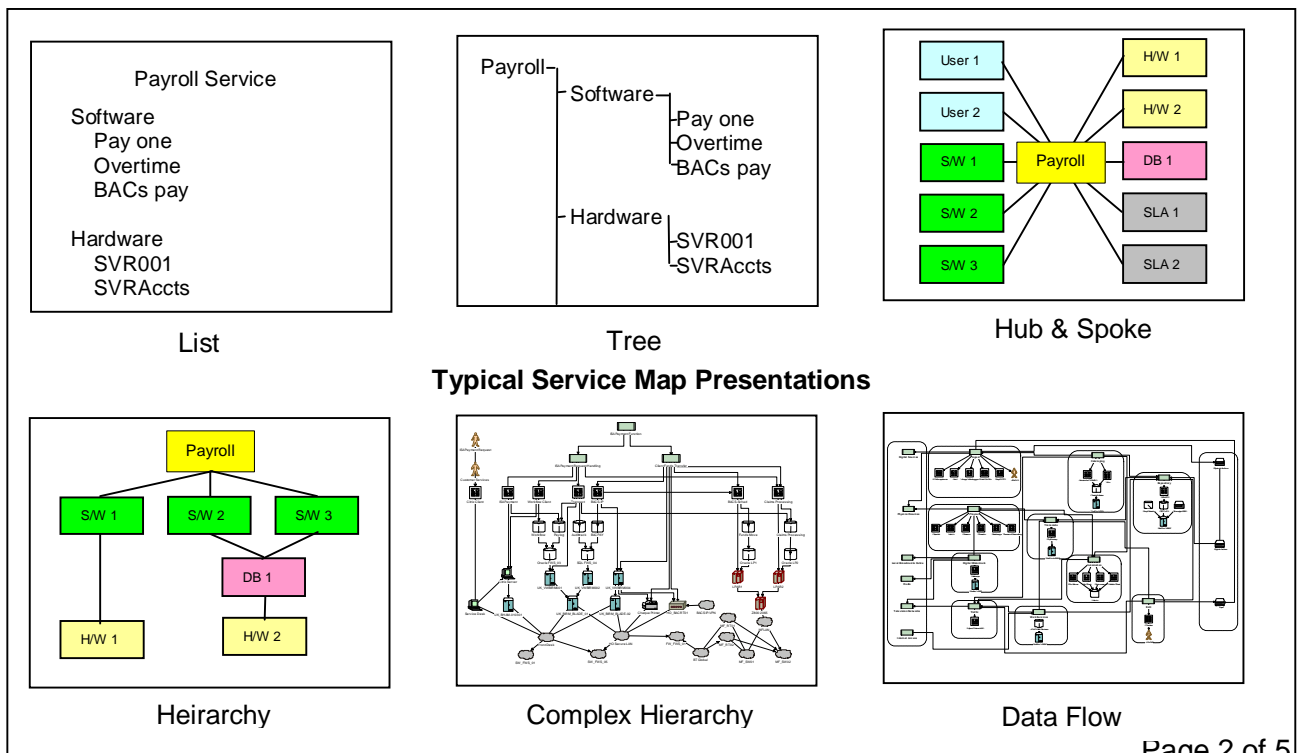
Does it work in practice? Well it can, if changes to systems and services are notified to service owners. For instance, it is a common problem to know who to notify when a server is virtualised as it could affect multiple services, but which ones?

The existing maps and CMDB data often diverge over time, as reverse engineering a map out of the CMDB is laborious and time-consuming for

CI Hierarchy	
Business Process	CRM, Accounts Payment, HR
Service	Payments, Customer Complaints, Payroll
Systems	Collections of Software Applications
Software	Specific applications, data or databases
Software Infrastructure	DBMS such as Oracle, SAP
Virtual Infrastructure	Virtualised Servers, PCs, Storage, VLANs
Hardware	Servers, PCs, Networks, Storage, Networks
Environment	Locations, Cabling, Power

anything but the simplest environment. In addition, there are many ways to show dependencies between the layers in a CI hierarchy, all of which may be a required output. In other words, another map!
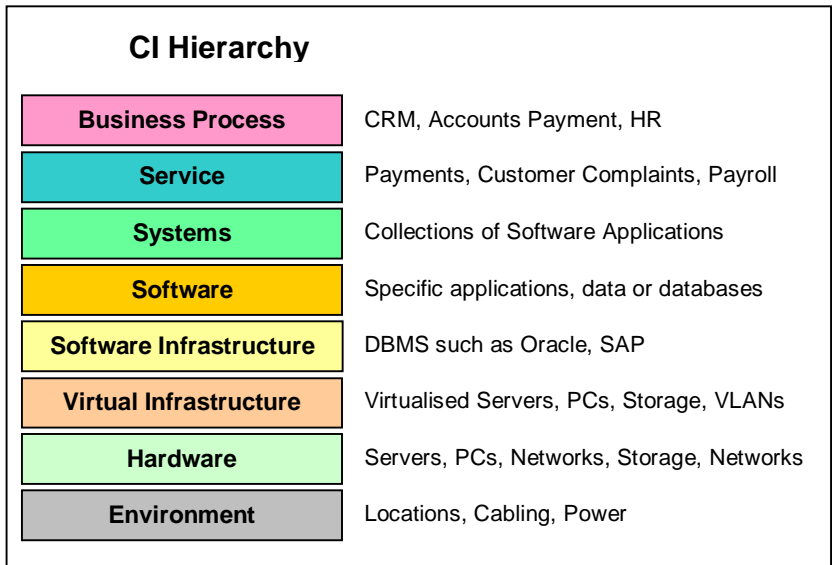
## Different Presentation Styles of Service Maps

Below are some examples of different methods which are often used to display relationships between CIs, all providing different types of understanding from a common set of CI data.

Payroll Service

Software
    Pay one
    Overtime
    BACs pay

Hardware
    SVR001
    SVRAccts

List

Payroll
  Software
    Pay one
    Overtime
    BACs pay
  Hardware
    SVR001
    SVRAccts

Tree

Hub & Spoke

Typical Service Map Presentations

Heirarchy

Complex Hierarchy

Data Flow

As the maps become more technical in nature and describe functional relationships, it becomes increasingly difficult to satisfy all needs. For instance, to understand how software transactions could be affected by a new version of code (e.g. a batch process), detail on hardware, networks and other technologies may be irrelevant. On the other hand, to understand the impact of rebooting a server, seeing which pieces of software and services are impacted is critical – so having multiple technologies or CI groupings on a page can be highly desirable. The trouble is that the detailed diagrams are not easily understood by non-experts. So in producing service maps, we have to cover multiple types of output – a key reason why AssetGen SysMap was developed.

In practice, we recommend the A-Z approach and provide the capability to produce maps with differing levels of detail to suit different needs, drawing from a common set of data.

The screenshot on the right shows another type of output available from AssetGen SysMap, which is a simple form that can be launched directly or from within other applications. Within this form, we are analysing the relationships between CIs for simple what-if type queries. "If I take this server down, what is the impact on systems, users, business functions, etc?" The CMDB has the data that describes the complexity of service delivery, but it still needs to be used by anyone wanting to plan, react or predict a change.
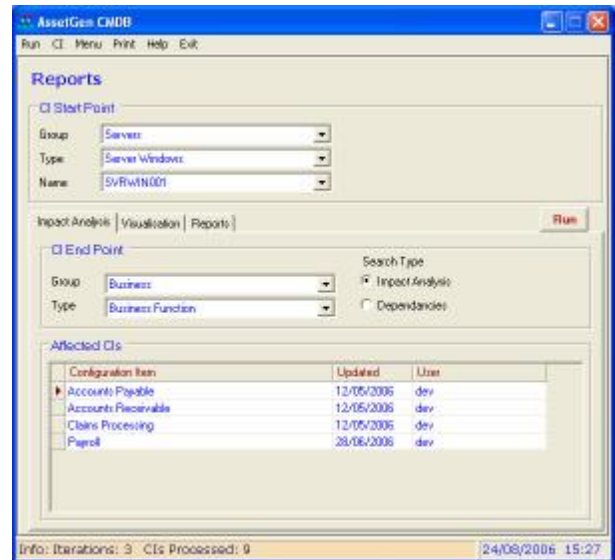

We have found that the majority of CMDB users are not really interested in looking at detailed maps, they just want a quick answer to a question, with the supporting service map being available if needed.

The CMDB approach of having a common knowledge base supporting multiple processes can deliver significant benefits in terms of control, predictability and the ability to manage more effectively during incidents. However, it does require you to make sure that system users get the information they need in a format they can use and understand. We can go further however and extend a CMDB for other mapping techniques.

## Enhance the Value of Your CMDB

If the CMDB has a maintained list of CIs, with process around them to ensure they are accurate, why don't we use it as a base for mapping the IT systems from other perspectives?

- Service mapping
- System and SOA architectures
- Batch processes
- Release and test models
- Network & Storage architectures
- Data flow diagrams
- Data mapping of critical systems
- Risk and recovery plans

Data Flows

Network Diagram

System Architecture

Service Map
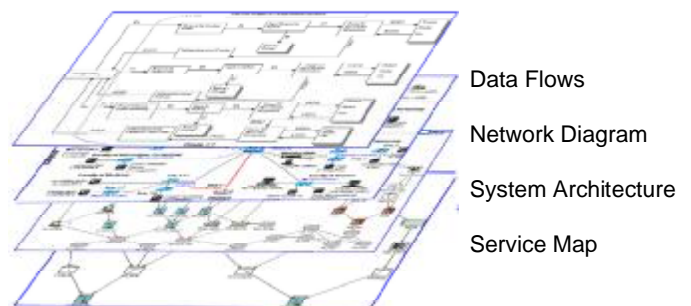
For example, the batch processes that link different systems together are typically represented as a flow chart. Trying to understand which pieces are involved in delivering each part of the process is a variation of mapping. The methods outlined previously probably duplicate existing data, rather than mine it from existing systems such as a CMDB. A technique we use to help accomplish the above in AssetGen SysMap is to filter on CI and relationship groups. In effect, you can overlay multiple maps on the same infrastructure – just like an A-Z. So if you need to reboot a server or upgrade a database, the potential impact could be understood from different perspectives, with automation helping to deliver regularly updated maps as required.

© AssetGen Limited

On the right is another technique for use with service maps, where we overlay incident, problem or change data from the service desk on a service map. By maintaining CI naming conventions, any CMDB derived information can have other data associated to show hot spots. In the example on the right, there are changes being implemented by server and software teams on a common platform. The same technique can show related incidents, problems and changes.
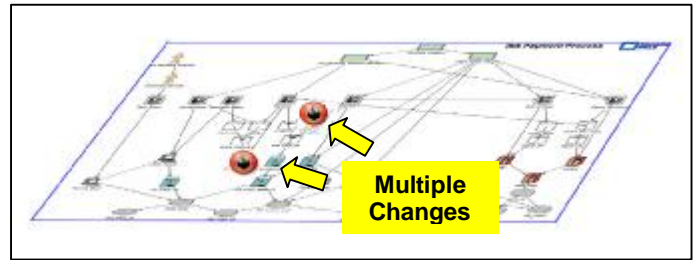

A fundamental issue is that a key CI, such as a server, will appear in many maps. If the existing CMDB helps to reduce the multitude of spreadsheets and ad hoc databases in the service management environment, why not use the same data to reduce other data sets? In reality, delivering a CMDB also delivers standards, processes, ownership and other benefits which other teams often cannot achieve by themselves.

Creating the Service Maps

Automation in practice takes the form shown on the right where we use offline data from the service desk CMDB and other sources, bringing it into AssetGen SysMap. Our approach can use a variety of existing data sets ranging through spreadsheets, databases, and CMDBs embedded in standard service desks.

Service maps can be created in MS Visio™ on demand, focused around top down, bottom up or CI centric views. In addition all CIs on a service map are hyperlinked in Visio back to a web impact analysis form, so that additional understanding can be gained such as reporting and audit trails. This hyperlinking can also be incorporated in other toolsets such as monitoring systems and data centre management toolsets (AssetGen Connect) relatively easily.
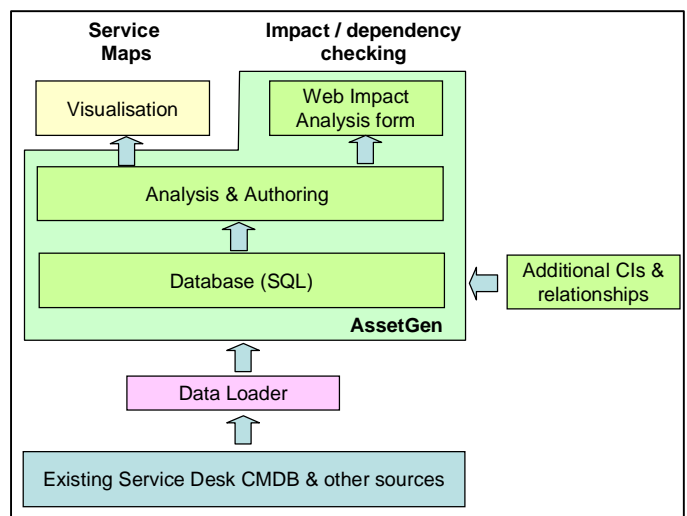
The data loader is a piece of middleware which can incorporate filtering as well as reconciliation across other sources. Any additions or modifications to CIs or attributes are captured, with audit trails to help maintain integrity of data. There are a number of advantages when implementing AssetGen SysMap apart from its technical functionality, many of which are not technology specific, but are to do with the real world issues that face IT departments and managers implementing CM. For example:

- Quick delivery of service maps in Visio covering many 1000s of CIs
- Validation of existing CMDB and service desk data, where visualisation quickly shows data errors
- No impact on service desk performance when creating service maps, or analysing change impact
- Service maps can incorporate additional CI data and relationships not held in the service desk
- Support environments with multiple service desks or outsourcers, where multiple data sources exist
- Enable service mapping to be delivered and maintained when service desks are upgraded/migrated
- Initial CI data and processes can be developed outside of the service desk until finalised/validated

As a principle, we wouldn't recommend CI data is held outside of the CMDB, but there are situations where it is not practical to try and encompass all needs within a standard service desk CMDB. Its primary purpose may have been to manage incidents, rather than hold detailed information on systems and dependencies.

If an organisation had 50 services, delivered by 1000 servers, then to provide service and server centric views would require 1050 diagrams. It is common sense to produce 50 standard service views and the others can be generated as required in a few seconds. Scale up the number of servers, add software, networks and storage and the potential ways of viewing CIs rises exponentially – with automation being the only practical way.

Automating Service Maps from the CMDB

The focus of this white paper is to show how a CMDB can be used to generate different forms of service maps, helping technology and service management teams understand dependencies between systems.

If you already have a populated CMDB based on a service desk, then little work has to be carried out to provide service maps. A quick demonstrator or proof of concept may be all that is needed to give the confidence that benefits are achievable.

Most organisations we come across have disparate information of varying degrees of breadth and depth, so there are ownership, process and technology issues to overcome. Our technology can accelerate the programme, as well as deliver a lower cost of ownership compared to manual production of service maps. In addition we have experience and integration skills to complement the project team.

The next time you see a service map being drawn on a white board, ask yourself if it is because:

       a. the CMDB can't provide it, but we have the data
       b. we don't have the data in a CMDB
       c. our systems are so complex, only a white board is suitable

There are now the technology and techniques to avoid the workload of manually mapping your services, data, systems, architectures, data flows, test and release models. Take advantage of them.

The Author

David Cuthbertson is a director of AssetGen Limited, a UK software company focused on applications for infrastructure management. He is a regular industry speaker at BCS and ITSMF events on best practices and the application of configuration management techniques to ICT and data centre infrastructure. Previously, he was the chairman and a founding member of the BCS Service Management Specialist Group (SMSG).

AssetGen Limited
3 Church Street, Cirencester
Gloucestershire, United Kingdom GL7 1LE
Tel +44 (0)8707770717 Fax +44 (0)8707 518852
www.assetgen.com

North America   Tel +1   301 947 6714
Asia Pacific (including Australia)   Tel +61 (0)417 231726