

# Mapping Services, Systems and Servers using the CMDB

David Cuthbertson, Square Mile Systems

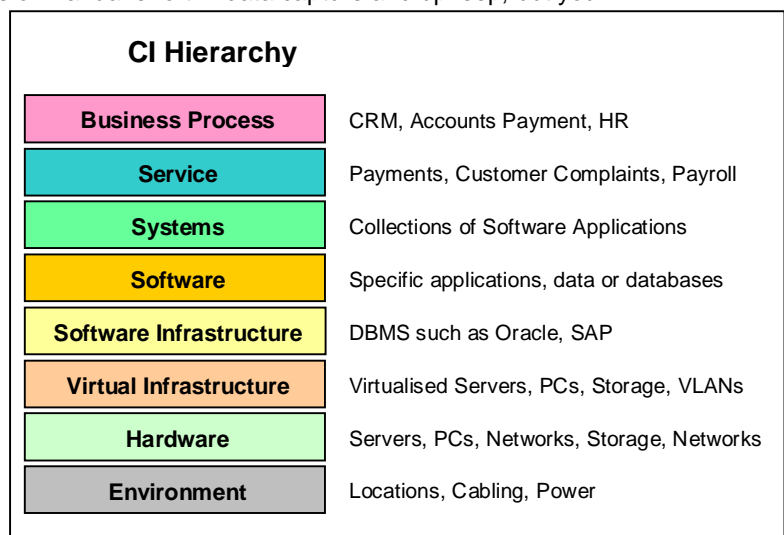
***Understanding the dependencies between IT components is critical to managing change effectively. The relationships between services, software and hardware should also be known when creating risk and recovery plans. However, the creation and maintenance of service maps is often problematical due to scope, the amount of manual effort involved in data capture, the limited skill sets available and a suitable data repository, or CMDB.***

## Why Create and Maintain Service Maps

Many organisations get by without mapping their service infrastructure, but there is a growing need to create a CMDB (or equivalent) if you want to get processes more integrated. We know that trying to document complex IT systems is painful and involves lots of manual effort in data capture and upkeep, but you will have to create service maps at some stage.

The hierarchy diagram on the right shows typical types of layers used for service mapping. We need to understand the dependencies between and within layers when managing change to critical services. In addition we may map documents and other data to the different Configuration Items (CIs) with greater levels of detail.

For many, this model is too complex for general use so often the layers are bundled together. The systems layer may comprise all software, databases, and external feeds for instance. You have to make a design choice – too many layers in the CMDB results in more complexity to manage, too few and it is of limited value.



There are many practical reasons why you create service maps.

1. It's the easiest way to record relationships so that they can be input into a CMDB.
2. The relationships between CIs in an existing CMDB can't be validated easily without a service map.
3. Without services and component dependencies mapped, IT teams will often categorise changes and impacts inaccurately, resulting in unnecessary disruption and inconsistent reporting. You can't see "hotspots" on a service map if the map does not exist.
4. Reacting to an incident or unforeseen event is made more difficult if you can't determine dependencies and potential risks of any actions you intend to take (or worse, have already taken).
5. Service reporting becomes of limited use as incidents and problems can't be assigned easily as far down the hierarchy tree as possible to reflect the root cause.
6. Problem management is time consuming and less effective if you have to create a service map for each problem review! This is one of the reasons why there are white boards in incident rooms.
7. Service continuity, availability, capacity, financial and other management processes become too difficult and costly to manage easily as single points of failure; duplication and other factors are too difficult to comprehend. So each team creates its own knowledge base not correlated with others.
8. Information on major incidents or problems can be "distorted" by individuals as detailed knowledge of system interaction is released economically to support opinions, commercial or political objectives.
9. There is a contractual obligation, or specific objective to show evidence of how you map services and maintain them. Somebody wants to see the evidence, so it has to be done!

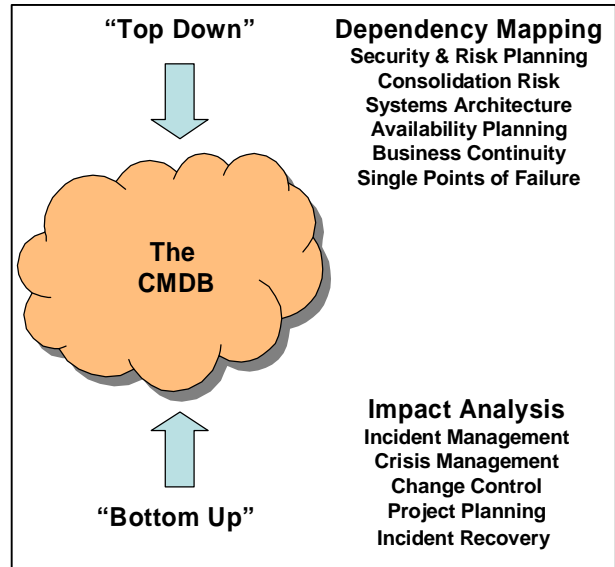
In addition to these, you may have your own requirements, ranked with priorities.

## What types of Service Maps are there?

There are many ways to look at service mapping. Different types of maps are often found within service desks and other support tools, so a decision must be made on the type of view you need to deliver first. Are you looking for a “top down” service map – ie starting at a service and looking at underlying or supporting CIs.

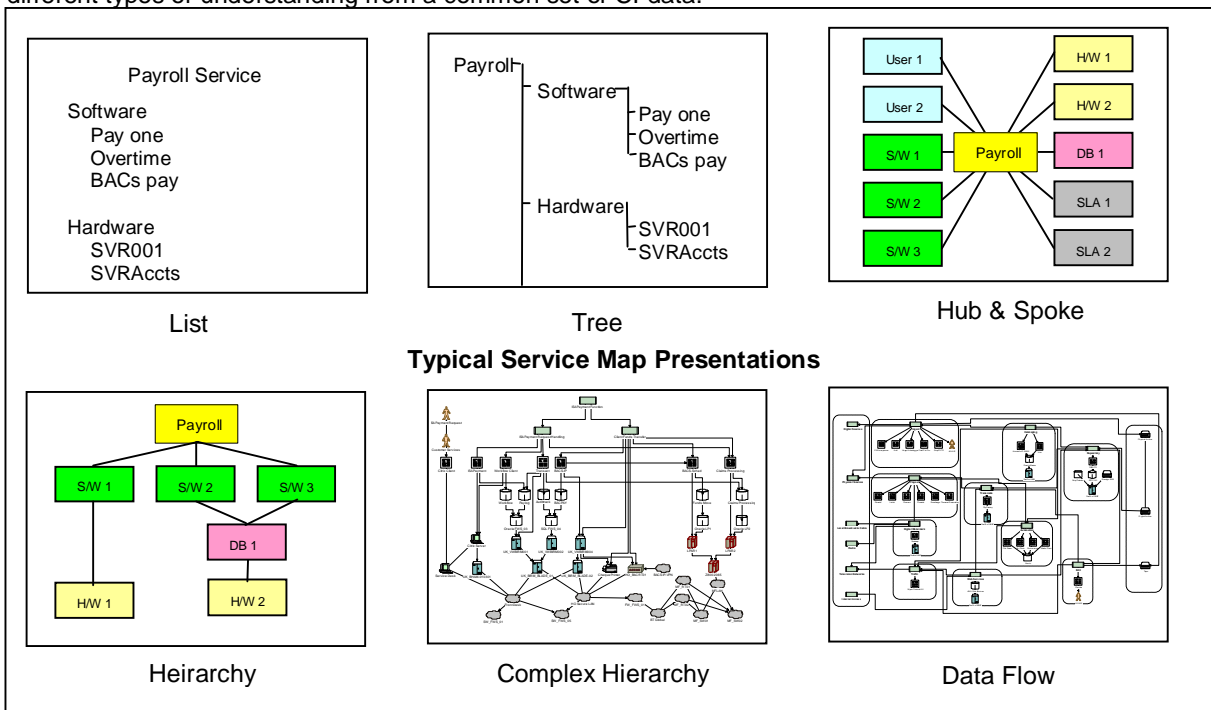
Or do you need a “bottom up” view starting at a component such as a server and looking at the software and services affected? The picture on the right emphasises typical uses of the CMDB and the type of mapping best suited.

When you collect the information it may in different formats, with the benefit of the CMDB that they will all be normalised into a common data structure. When you want to understand how CIs are linked then the presentation may also take various forms. Each has its own benefits, with some not scaling well for the bigger, or more complex environments.



CMDB Views

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



The top three views are relatively easy to automate with software as the display structures are simple, but are not suited to large numbers of CIs, or multiple relationships. The bottom three provide more information about service and system dependencies, but are more manual in nature. Many of the more modern CMDB systems try and automate the production of these service maps, though there is often a manual tidy up required to enable comprehension. After all, a map is designed to suit a need and we may need just an overview, a detailed technical view or a mixture of both. And we still have to get the base data to begin with!

In the authors experience, the complex hierarchy view should be the primary aim as the information is suited to multiple uses. For instance, the same service map will help teams categorise changes at the best level. It will enable a service continuity manager to determine which CIs are delivering the critical parts of a service. Billing for services can be easily determined depending on which CIs are shared or dedicated to business groups. Not an issue if you have only a few servers, but a definite benefit if you have hundreds of servers spread across multiple equipment rooms. We can more easily determine capacity, recovery and the impact of failures with a complex hierarchy map, than with the list and tree views.

## **The Service Mapping Project**

The ideal way to start is to have available details of your services, hardware, software and mapping standards so that the project team can focus on understanding relationships between CIs. For many, these details do not exist so a more pragmatic approach is required – one that delivers some tangible results in a fixed timescale.

Our recommendation is that you start with a few key services to help drive the definition of standards, naming and display. It also shows up the quality and depth of operational data, making it easy to focus attention on what should be audited and how. Auto-discovery packages can help initially with device details, but as much of the service mapping data is manual in nature, they are of limited use.

As the majority of services are mapped, the ownership of each service map should be assigned, often to a business service owner or service performance manager. It becomes their responsibility to regularly validate the service maps for which they are responsible. The configuration manager, or a team colleague should have the skills and tools to generate service maps from the underlying CMDB, supporting validation needs as well as specific project or management requirements.

The end goal is to have a populated, maintained CMDB from which service maps can be derived and updated to help the various process teams in their roles. As service desks are not primarily selected for their ability to present service maps, it often requires some reporting software or middleware to extract CMDB data, so service maps of the various types can be easily produced by visualisation tools.

A newer technique that is being increasingly adopted is to link service maps to incidents, problems and changes to reduce the discovery time during incidents or change reviews. It then becomes easy to see where there are “hotspots” or potentially conflicting changes across a large environment. It does of course assume that you already have a service map to link to...

## **The Author**

David Cuthbertson is a founding director of Square Mile Systems, a UK computer services company focused on configuration management solutions. He is a regular industry speaker at ITSMF events on best practices and the application of configuration management techniques to ICT and data centre infrastructure. He is the current chairman of the BCS Service Management Specialist Group (MSG).

### **Worldwide Headquarters (Europe & US)**

Square Mile Systems  
3 Church Street, Cirencester  
Gloucestershire, United Kingdom GL7 1LE  
Tel +44 (0)870 950 4651 Fax +44 (0)870 751 9268  
[www.squaremilesystems.com](http://www.squaremilesystems.com)



### **Asia Pacific (including Australia)**

Tel +61 (0)417 231726