What is Service Configuration Management?

Traditionally, configuration management dealt strictly with management and measurement at the component level. Today, service configuration management looks at all stages of the service lifecycle in order to provide information about an organization’s performance.

Managing service configuration is critical in understanding the lifecycle of a service. The Service Configuration Manager correlates views while tapping into discovery and mapping capabilities that provide new insight and visibility into the network, thereby providing the ability to manage the service and its lifecycle.

Service configuration management is composed of:

* **Configuration Management Database (CMDB):** Created by mining various configuration sources. A Configuration Management Database brings related information from multiple sources into one view, to create a CMDB without having to build a physical database.
* **Service Catalog:** Created by modeling business/technology services. A Service Catalog is a systematic list of service definitions and documentation complete with relationship mapping between services delivered as well as their underlying components.
* **Service Definition:** Augments the Service Catalog with dynamic links to underlying technology components.
* **Reports:** Assist with change/impact analysis, capacity planning, reconciliation, auditing, and data center consolidation or license usage.
* **Business Service:** Configuration alerts that foster proactive management.

# Key SCM Features

The key features of SCM are:

* Auto-discovery that detects dependencies, baselines “normal,” and identifies scheduled and unscheduled changes
* Integrated mapping of element relationships across an enterprise
* Integration of asset, configuration, and change sources into a CMDB
* Building and maintenance of Service Views automatically and dynamically, alleviating the need to create, update, and maintain views (models) manually

# SCM Methodology

Implementing a service configuration is a multistep process that allows leveraging and integrating important data and technology into a unified view. This process leverages, reuses, and combines configurations to parse data and views in a customized way.

The process of implementing SCM can vary depending on requirements, but the basic process includes these key areas and their implementation steps:

1. **Technology Integration:** Define and create adapters to integrate technology, data, and discovery. This can include integrations with network and application management systems, trouble ticketing systems, discovery tools, and databases.
2. **Service CatLog:** Define Service Views using the **Services** hierarchy in the console.
3. **1st Generation Service Configuration:** Build a correlated CMDB to unify all objects by object name.
4. **2nd Generation Service Configuration:** Create additional SCM definitions that will leverage previous definitions to do any of the following:
   * Slice and dice the views in a functional way using class, purpose, or use case.
   * Link back to the correlated CMDB or combine hierarchical views as required. Use a discovery tool to reveal dependencies.