ITIL – A guide to service asset and configuration management

The goal of service asset and configuration management

The goals of configuration management are to:

- Support many of the ITIL processes by providing accurate configuration information to assist decision making, e.g. the authorisation of changes, the planning of releases, and to help resolve incidents and problems faster
- Minimise the number of quality and compliance issues caused by incorrect or inaccurate configuration of services and assets
- To define and control the components of services and infrastructure and maintain accurate configuration information on the historical, planned and current state of the services and infrastructure

The purpose and objectives of service asset and configuration management

The purpose of service asset and configuration management is to:

- Identify, control, record, report, audit and verify service assets and configuration items
- Account for, manage and protect the integrity of service assets and configuration items through the service lifecycle by ensuring that only authorised components are used and only authorised changes are made
- Ensure the integrity of the assets and configurations required to control the services and IT infrastructure by establishing and maintaining an accurate and complete configuration management system.

The scope of service asset and configuration management

Asset management covers service assets across the whole service lifecycle. It provides a complete inventory of assets and who is responsible for their control. It includes:

- Full lifecycle management of IT and service assets, from the point of acquisition through to disposal
- Maintenance of the asset inventory. Configuration management ensures that selected components of a service, system or product (the configuration) are identified, baselined and maintained and that changes to them are controlled. It also ensures that releases into controlled environments and operational use are done on the basis of formal approvals
- The scope covers interfaces to internal and external service providers where there are assets and configuration items that need to be controlled, e.g. shared assets

Value to business of service asset and configuration management

Optimisation of the performance of service assets improves the overall service performance and optimises the costs and risks caused by poorly managed assets, e.g. service outages, correct licence fees and failed audits.

Service asset and configuration management provides visibility of accurate representations of a service, release, or environment that enables:

- Better planning of changes and releases
- Improved Incidents and problems resolution
- Service levels and warranties to be delivered
- Better adherence to standards, legal and regulatory obligations (less non-conformances)
- Changes to be traceable
- The ability to identify the costs for a service

Service asset and configuration management policies

Service asset and configuration management policies, objectives, scope and principles and critical success factors (CSFs) need to be developed. These policies are often considered with the change and release and deployment management policies as they are closely related.

There are significant costs and resources implications to implementing service asset and configuration management and therefore decisions need to be made about the priorities to be addressed.

Many IT organisations focus initially on the basic IT assets (hardware and software) and the services and assets that are business critical or covered by legal and regulatory compliance, e.g. software licensing.

Service asset and configuration management principles

The main policy sets out the framework and key principles against which assets and configurations are developed and maintained. Typical principles include:

- Ensuring that asset and configuration management operations costs and resources are controlled
- The need to deliver corporate governance requirements, e.g. software asset management
- The requirement for available, reliable and cost effective services
- The transformation from *find and fix* reactive maintenance to *predict and prevent* proactive management
- The level of control and requirements for traceability and audit
- Provision of accurate asset and configuration information for other business and service management processes
- Level of automation to reduce errors and costs

Basic concepts of service asset and configuration management

The configuration model

Configuration management delivers a model of the services, assets and the infrastructure by recording the relationships between configuration items. This enables other processes to access valuable information, e.g.:

- To assess the impact of proposed changes
- To assess the impact and cause of incidents and problems
- To plan and design new or changed services
- To plan technology refresh and software upgrades
- To plan release and deployment packages and migrate service assets to different locations and service centres

Configuration items

A configuration item (CI) is an asset, service component or other item that is, or will be, under the control of configuration management.

Configuration items may vary widely in complexity, size and type, ranging from an entire service or system including all hardware, software, documentation and support staff to a single software module or a minor hardware component.

Configuration items may be grouped and managed together, e.g. a set of components may be grouped into a release. Configuration items should be selected using established selection criteria, grouped, classified and identified in such a way that they are manageable and traceable throughout the service lifecycle.

The Configuration Management System (CMS)

To manage large and complex IT services and infrastructures, service asset and configuration management requires the use of a supporting system known as the Configuration Management System (CMS).

The CMS holds all the information for CIs within the designated scope. Some of these items will have related specifications or files that contain the contents of the item, e.g. software, document or photograph. For example, a

service CI will include the details such as supplier, cost, purchase date and renewal date for licences and maintenance contracts and the related documentation such as Service Level Agreements and underpinning contracts.

The CMS maintains the relationships between all service components and any related incidents, problems, known errors, change and release documentation and may also contain data about suppliers, locations and business units, customers and users.

The Definitive Media Library (DML)

The Definitive Media Library (DML) is the secure library in which the definitive authorised versions of all media CIs are stored and protected. It stores master copies of versions that have passed quality assurance checks.

This library may consist of one or more software libraries or file storage areas, separate from development, test or live file store areas. It contains the master copies of all controlled software in an organisation. The DML should include definitive copies of purchased software (along with licence documents or information), as well as software developed on site. Master copies of controlled documentation for a system are also stored in the DML in electronic form.

The DML will also include a physical store to hold master copies, e.g. a fireproof safe. Only authorized media should be accepted into the DML, strictly controlled by SACM.

The exact configuration of the DML is defined during the planning activities. The definition includes:

- Medium, physical location, hardware and software to be used, if kept online – some Configuration Management support tools incorporate document or software libraries, which can be regarded as a logical part of a DML
- Security arrangements for submitting changes and issuing documentation and software, plus backup and recovery procedures
- Capacity plans for the DML and procedures for monitoring growth in size
- Audit procedures
- Procedures to ensure that the DML is protected from erroneous or unauthorized change (e.g. entry and exit criteria for items)

Definitive spares

An area should be set aside for the secure storage of definitive hardware spares. These are spare components and assemblies that are maintained at the same level as the comparative systems within the controlled test or live environment. Details of these components, their locations and their respective builds and contents should be comprehensively recorded in the CMS. These can then be used in a controlled manner when needed for additional systems or in the recovery from incidents. Once their (temporary) use has ended, they are returned to the spares store or replacements are obtained.

Service asset and configuration management process activities

Management and planning

The management team and configuration management should decide what level of configuration management is required for the selected service or project that is delivering changes and how this level will be achieved. This is documented in a configuration management plan.

Often there will be a configuration management plan for a project, service or groups of services, e.g. network services. These plans define the specific configuration management activities within the context of the overarching service asset and configuration management strategy.

Configuration identification

When planning configuration identification it is important to:

- Define how the types of assets and configuration items are to be selected, grouped, classified and defined by appropriate characteristics to ensure that they are manageable and traceable throughout their lifecycle
- Define the approach to identification, uniquely naming and labelling all the assets or service components of interest across the service lifecycle and the relationships between them
- Define the roles and responsibilities of the owner or custodian for configuration item type at each stage of its lifecycle, e.g. the service owner for a service package or release at each stage of the service lifecycle.

The configuration identification process activities are to:

- Define and document criteria for selecting configuration items and the components that compose them
- Select the configuration items and the components that compose them based on documented criteria
- Assign unique identifiers to configuration items
- Specify the relevant attributes of each configuration item
- Specify when each configuration item is placed under configuration management
- Identify the owner responsible for each configuration item

Naming configuration items

Naming conventions should be established and applied to the identification of all CIs. Individual CIs should be uniquely identifiable by means of the identifier and version. The naming conventions should be unique and take into account the existing corporate or supplier naming/numbering structures.

The naming conventions should include the management of:

- Hierarchical relationships between CIs within a configuration structure
- Hierarchical or subordinate relationships in each CI
- Relationships between CIs and their associated documents
- Relationships between CIs and changes
- Relationships between CIs, incidents, problems and known errors

Labelling configuration items

All physical device CIs should be labelled with the configuration identifier so that they can be easily identified. Plans should be made to label CIs and to maintain the accuracy of their labels. Items need to be distinguished by unique, durable identification.

A standard policy on labelling hardware is beneficial at the users/service desk, e.g. if all hardware is labelled in the bottom left-hand corner of the left side, it is much quicker and easier to explain to the user where they will find the required information.

Attributes for configuration items

Attributes describe the characteristics of a CI that are valuable to record and which will support service asset and configuration management and the ITSM processes it supports.

Typical attributes include:

- Unique identifier
- CI type
- Name/description
- Version (e.g. file, build, baseline, release)
- Location
- Supply date
- Licence details, e.g. expiry date
- Owner/custodian
- Status
- Supplier/source
- Related documents
- Historical data, e.g. audit trail
- Relationship type

Relationship types

Relationships describe how the configuration items work together to deliver the services.

These relationships are held in the CMS – this is the major difference between what is recorded in a CMS and what is held in an asset register.

The relationships between CIs are maintained so as to provide dependency information.

For example:

- A CI is a part of another CI
- A CI is connected to another CI
- A CI uses another CI
- A CI is installed on another

Configuration control

Configuration control ensures that there are adequate control mechanisms over CIs while maintaining a record of changes to CIs, versions, location and custodianship/ownership.

No CI should be added, modified, replaced or removed without an appropriate controlling documentation or procedure being followed.

Policies and procedures need to be produced for:

- Licence control, to ensure that the correct number of people are using licences and that there is no unlicensed use and no wastage
- Version control of service asset, software and hardware versions, images/builds and releases
- Access control, e.g. to facilities, storage areas and CMS
- Build control, including the use of build specification from the CMS to perform a build
- Promotion, migration of electronic data and information
- Taking a configuration baseline of assets or CIs before performing a release (into system, acceptance test and production) in a manner that can be used for subsequent checking against actual deployment
- Deployment control including distribution
- Installation
- Maintaining the integrity of the DML

Often there are many procedures that can change a CI. These should be reviewed and aligned with the CI types where possible as standardisation prevents errors. During the planning stage it is important to design an effective configuration control model and implement this in a way that staff can easily locate and use the associated training products and procedures.

All configurations are controlled through the change management process.

Status accounting and reporting

Each asset or CI will have one or more discrete states through which it can progress. The significance of each state should be defined in terms of what use can be made of the asset or CI in that state. There will typically be a range of states relevant to the individual asset or CIs.

A simple example of a lifecycle is:

- Purchased
- Request for change assigned
- In build
- In test
- Live
- Declining
- Retired/withdrawn
- Disposed

The way CIs move from one state to another should be defined and at each lifecycle status change the CMS should be updated with the reason, date time stamp and person that did the status change.

Service asset and configuration reports

Typical reports include:

- A list of product configuration information included in a specific configuration baseline
- A list of configuration items and their configuration baselines
- Details of the current revision status and change history
- Status reports on changes and deviations
- Details of the status of delivered and maintained products concerning part and traceability numbers
- Revision status
- Report on unauthorised usage of hardware and software
- Unauthorised CIs detected
- Variations from CMS to physical audit reports.
- Status reports of assets for a business unit or software licence holdings are often required by financial management for budgeting, accounting and charging

Verification and audit

The activities include a series of reviews or audits to:

- Ensure there is conformity between the documented baselines (e.g. agreements, interface control documents) and the actual business environment to which they refer
- Verify the physical existence of CIs in the organisation or in the DML and spares stores, the functional and operational characteristics of CIs, and to check that the records in the CMS match the physical infrastructure
- Check that release and configuration documentation is present before making a release

When to carry out verification and audit

Before a major release or change, an audit of a specific configuration may be required to ensure that the customer's environment matches the CMS.

Before acceptance into the live environment, new releases, builds, equipment and standards should be verified against the contracted or specified requirements.

There should be a test certificate that proves that the functional requirements of a new or updated CI have been verified, or some other relevant document, for example a request for change.

Unregistered and unauthorised items that are discovered during configuration audits should be investigated and corrective action taken to address possible issues with procedures and the behaviour of personnel. All exceptions are logged and reported.

Configuration audits check in addition that change and release records have been properly authorised by change management and that implemented changes are as authorised. Configuration audits should be considered at the following times:

- Shortly after changes to the CMS
- Before and after changes to the IT services or infrastructure
- Before a release or installation to ensure that the environment is as expected
- Following recovery from disasters and after a *return to normal* (this audit should be included in contingency plans)
- At planned intervals
- At random intervals
- In response to the detection of any unauthorized CIs

If there is a high incidence of unauthorised CIs detected, the frequency of configuration audits should be increased.

Service asset and configuration management relationship with other processes

As the single virtual repository of configuration data and information for IT service management, service asset and configuration management supports and interfaces with every other process and activity to some degree.

- Change management – identifying the impact of proposed changes
- Financial management – capturing key financial information such as cost, depreciation methods, owner and user (for budgeting and cost allocation),maintenance and repair costs
- ITSCM – awareness of assets the business services depend on, control of key spares and software
- Incident/problem/error – providing and maintaining key diagnostic information; maintenance and provision of data to the service desk
- Availability management in detection of points of failure
- Etc!

The relationship with change and release and deployment is particularly important, with these processes benefiting greatly from a single coordinated planning approach.

Key performance indicators and metrics for service asset and configuration management

The following measures are applicable:

- Percentage improvement in maintenance scheduling over the life of an asset (not too much, not too late)
- Degree of alignment between provided maintenance and business support
- Assets identified as the cause of service failures
- Improved speed for incident management to identify faulty CIs and restore service
- Impact of incidents and errors affecting particular CI types, e.g. from particular suppliers or development groups, for use in improving the IT services
- Percentage re-use and redistribution of under utilized resources and assets
- Degree of alignment of insurance premiums with business needs
- Ratio of used licences against paid for licences (should be close to 100%)
- Average cost per user for licences (i.e. more effective charging options achieved)
- Achieved accuracy in budgets and charges for the assets utilised by each customer or business unit
- Percentage reduction in business impact of outages and incidents caused by poor asset and configuration management
- Improved audit compliance

Other measures include:

- Increased quality and accuracy of asset and configuration information
- Fewer errors caused by people working with out of date information
- Shorter audits as quality asset and configuration information is easily accessible
- Reduction in the use of unauthorized hardware and software, nonstandard and variant builds that increase complexity, support costs and risk to the business services
- Reduction in the average time and cost of diagnosing and resolving incidents and problems (by type)
- Improvement in time to identify poor performing and poor quality assets
- Occasions when the *configuration* is not as authorised

- Changes that were not completed successfully or caused errors because of poor impact assessment, incorrect data in the CMS, or poor version control
- Exceptions reported during configuration audits
- Value of IT components detected in use
- Reduction in risks to the business/organisation due to early identification of unauthorised change