CWDS
Child Welfare Digital Services

Child Welfare Digital Services Project

Service Asset and Configuration Management Plan

September 2017

Revision History

Revision / Version #	Date of Release	Author	Summary of Changes
.01	3/9/2015	D. Serpa	Original Document
.02	03/17/2015	D. Serpa	Updated to reflect decision to point to document management process with the transition to SACM in 6 months.
.03	4/1/2015	D. Serpa	Incorporate feedback and new content into draft SACM based on review feedback from CB and LB.
.04	6/22/2015	N. DeVriend	Edited based on discussions with PMO team.
.05	9/10/15	S. Tanniru	• Formatting changes • Provided the expanded forms of several acronyms • Added Document naming convention Added Document Versioning
.06	10/09/15	S. Tanniru	1 Incorporated changes suggested by QA, IV&V, Project Manager, and the Project Director.
1.0	10/09/2015	S. Tanniru	2 Submitted Version 1.0 for base lining.
2.0	8/5/2016	C. Bratton	Changed from SDLC Waterfall to Agile
2.1	8/22/2016	C. Bratton	Incorporated changes suggested by QA
3.0	09/20/2017	Arpit Patil, Chad Bratton	Updated introduction, assumptions, risks, controlled assets

TABLE OF CONTENTS

1 Executive Summary

Information Technology (IT) services are made up of several individual components: Things like Processes, Data, Hardware, Software, Systems, Services and Connectivity to name a few. From the Information Technology Infrastructure Library (ITIL), Service Asset and Configuration Management (SACM) is all about proper planning and management of the relationships and attributes of all these components across every service in our infrastructure.
As the DevOps team at CWDS, we are in process of implementing the controls to keep track of all aspects of our assets and configurations. This plan will lay out a draft for the approach we will adopt to implement the SACM plan.

1.1 Purpose

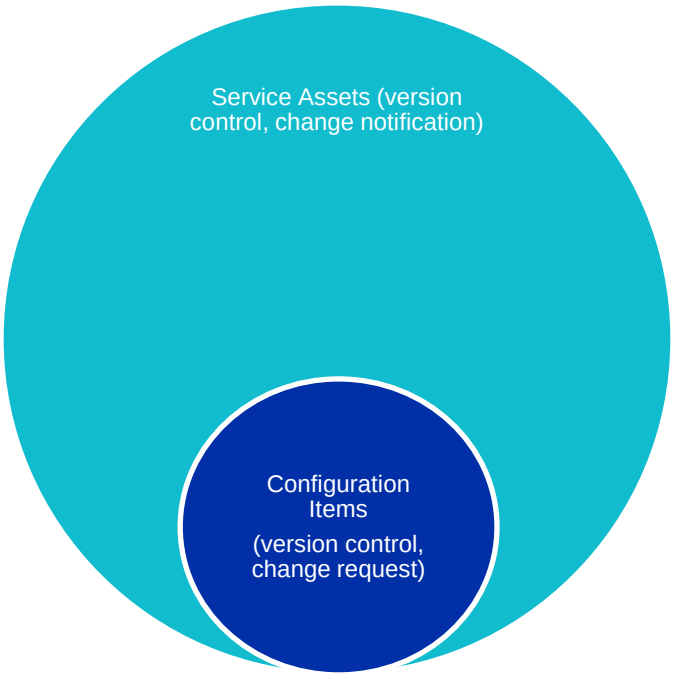
This document describes the Configuration Management (CM) Plan (hereinafter referred to as the "plan") for the Child Welfare Services – New System (CWS-NS) Project (hereinafter referred to as the "Project"). The purpose of this plan is to document how the Project will identify State service assets and configurable items (hereinafter referred to as "SA's", and "CI's") including internal deliverables and work products, and State-owned software products, so that our service assets maintain an accurate configuration record and decisions can be made using accurate information.
The term "Service Asset" refers to any resource or capability. Assets can be one of the following types: management, organization, process, knowledge, people, information, applications, and infrastructure.
The term "Configuration Item" refers to any component that needs to be managed in order to deliver an IT Service, and anything that needs to follow a formal change control process. Information about each CI will be recorded in a configuration record within the Configuration Management Database (CMDB) and will be maintained throughout its lifecycle. CIs are under the control of change management. Project CIs will include IT services, infrastructure, software, source code, and some formal documentation.

1.2 Strategy

In order to successfully complete the implementation of this plan, our strategy is to implement the key activities of SACM in a phased manner in alignment with ITIL best practices for planning, identification, control, status accounting and reporting, verification and audit, and managing information.

1.3 Scope

The scope of this plan covers SAs (Service Assets) that the DevOps Engineering team shall manage – including internal deliverables and work products, software products, infrastructure, and other CIs (Configuration Items) – across the whole service lifecycle from initiation through operations. All CIs are considered service assets, but not all service assets are considered CIs.



1.3.1 In-Scope

The scope of configuration management includes:

Configuration Identification

- Selecting, grouping, and classifying SAs and CIs (Service Assets and Configuration Items)
- Identifying SAs and CIs
- Naming convention for SAs and CIs
- Labeling SAs
- CI relationships and dependencies

Configuration Control

- Change Management
- Version Control
- Baseline
- Repository and Access Control
- Software Licensing

Status Accounting and Reporting

- Status Accounting
- Reporting

1.3.2 Out of Scope

- Project IT Hardware Asset Management is currently not in scope for this plan.
- Legacy System (CWS/CMS) processes are out of scope for configuration management:
- IBM is responsible for configuration management of software, hardware, licenses and source code for CWS/CMS.
- IBM provides source code to the repository (change occurs as a result of the release process).
- The CWDS organization won't be changing the way that IBM performs configuration management – we need to define our processes for the project.
- The State manages the contract for IBM, but IBM manages all of the configuration. The CWS-NS project will have a separate CM process than the current IBM process.

1.4 Objective

The objective of the Service Asset and Configuration Management Plan (SACM) is to document and inform project stakeholders about Configuration Management (CM) with the project, what CM tools will be used, and how they will be applied by the project to promote success. The CWDS SACM Plan defines the project's structure and methods for:

- Identifying, defining, and baselining CIs (configuration items)
- Controlling modifications and releases of CIs
- Reporting and recording status of CIs and any requested modifications
- Ensuring completeness, consistency, and correctness of CIs
- Controlling storage, handling, and delivery of the CIs

1.5 Assumptions and Constraints

Assumptions: A Microsoft Excel workbook is the current repository for configuration items that will be used until another solution is implemented
Constraints: Managing an excel workbook requires lots of manual labor, and we are limited to the existing functionality of Microsoft Excel.

1.6 Risks

Title	Description
No CMS/CMDB (Configuration Management System/Configuration Management Database)	Without a CMDB, we are spending lots of time manually updating CI attributes and there is a higher chance of human error.

1.7 Integration with other CWDS Plans

Availability Management Plan: The Availability Management Plan relates to the SACM plan by tracking and managing the availability of identified CI's, which is extremely important to ensure system up-time for critical infrastructure.

IT Operations Management Plan: The IT Operations Management Plan relates to the SACM plan by tracking and managing the related operations and processes or procedures for identified CI's.

Capacity Management Plan: The Capacity Management Plan relates to the SACM plan by tracking and managing the capacity of identified CI's, which is extremely important to ensure critical infrastructure capacity is consistently within defined thresholds.

Service Continuity Management Plan: The Service Continuity Management Plan relates to the SACM plan by applying continuity and recovery standards to all identified CI's by ensuring a seamless user experience in the event of a disaster.

Change Management Plan: The Change Management Plan is the process of managing changes to project artifacts, application code, deliverables, or baselines at a strategic, tactical, and operational level. The purpose of this Change Management Plan is to establish a standard approach for the approval and tracking of proposed changes for the project.

Document Management Plan: The purpose of the Document Management Plan is to capture how document management is defined for the project, as well as how internal Project documentation is developed and reviewed. Document Management is the process of organizing, storing, protecting, and sharing documents. This plan describes how to manage the hard copy and electronic repositories of documents, historical information, and provides a consistent approach to the creation, update, and format of documents.

1.8 Document Maintenance

The CWDS Service Asset & Configuration Management (SACM) Plan will be updated as processes and procedures change. A minor version change does not change the intent of the document and consists of spelling, grammatical and minor corrections. A major version is when a document's content is changed and represents a change in intent, change in process, or procedures.

2 Roles and Responsibilities

The following tables describe the roles and responsibilities of the CWS-NS Project stakeholders in the SACM Plan arena.

Roles	Responsibilities
Configuration Management Analyst	• Sponsor, design and manage the configuration process and its metrics • Define the service assets that will be treated as configuration items • Communicate process information or changes to ensure awareness. • Provide process resources to support activities required throughout the service lifecycle • Updates the CMDB (Configuration Management Database) as needed • Address issues with the configuration process operation • Work with Project Librarians to plan and coordinate all process activities • Plan and manage support for CM (Configuration Management) tools and processes • Coordinate Interfaces between CM and other processes, especially change management, release and deployment management, and knowledge management • Periodically review the process strategy to ensure that it is still appropriate • Work with service owners and other process managers to ensure the smooth running of services • Monitor and report on process performance. • Propose scope of the configuration management plan • Define the structure of the configuration management system, including CI (Configuration Item) types, naming conventions, required and optional attributes and relationships • Train staff in CM principles, process and procedures • Perform configuration audits • Respond to quality audit findings and work with the Quality Manager for process improvements/corrective action • responsible for disposal of software
Project Librarian (Can be performed by	• Control the receipt, identification, storage, and withdrawal of all supported CIs • Maintain status information on CIs and providing this as appropriate • Archive superseded CIs

Configuration Analyst)	• Assist in conducting configuration audits • Identify, record, store and distribute issues relating to service asset and configuration management • Work with the Quality manager to conduct and report on Configuration Audits.
Application Development, Data, Technical Architecture Teams	• Follow the processes to identify and change Configuration Items. • Use the Configuration Management tools in the course of development, enhancements, and addressing defects in the CWS-NS project. • Indicate when changes are ready for a build.
Quality Manager	• Review the work of the CM process owner and other project teams as defined in Quality Management Plan to ensure compliance with this plan. • Conduct quality audits of the CM process and tool.
Change Initiator	• Fill out Change Request (CR) for a New/Changed/Retired CI
Change Developer	• Develop planned change (includes implementation, verification, back-out plan if required)
Change Manager (Can be performed by Configuration Analyst)	• Evaluate impact of resources required for implementation, costs and schedule • Assess CR to consider business impact, technical impact, options, recommendations and (as needed) cost. • Records CR updates, facilitates CCB meeting • Perform post implementation review to verify that change does what the request stated it would do
Change Implementer (Can be performed by Change Manager)	• Implements change • Validates that change is successful
DevOps Engineer	• Helps with CI identification • Responsible for collaborating with state staff and defining standards for the Service Asset and Configuration Management Plan and associated processes • Responsible for Implementing and maintaining CMDB Infrastructure • Automation of logging, reporting, auditing of CIs
Users	• Depend on CMDB to reflect accurate and current information to make informed decisions.

3 Plan Approach and Methodology

3.1 Transition to Agile

As a result of the transition from a pure SDLC (Systems Development Life Cycle), waterfall change management approach to a more agile approach, the CWDS project has incorporated the following agile-like principles into the new changes in this document:

- Project Documentation (Project Management Plans and Processes) are no longer considered configuration items. They are considered service assets.
- Removing redundancy and complexity from all aspects of development: from designs, from requests and requirements, and even from and processes and tools.
- Question what really is needed and remove the things we don't really need. Avoid overly complex or laborious tools and processes. These extraneous elements add complexity and/or redundancy that can create more friction than forward motion on development projects. The result is that they add little if any value to the main goal of developing quality software.
- Agile Configuration Management is more about accepting and embracing change than preventing or controlling it. Agile methods acknowledge that too much "control" can harm development efforts of both software and business teams.

3.2 Planning

A Service Management Plan will be created for each service that is offered by the project. This plan typically covers the scope and objective of a service, the activities and procedures as well as roles and people required. This SACM (Service Asset and Configuration Management) plan is in fact the first part of the strategy.

3.3 Identification and Classification

This stage involves creating a complete inventory of all the CIs (Configuration Items) in our infrastructure. In this activity, we will essentially record every bit of data about our CI that is necessary for effective operation. In order to do so, all CIs are given unique identifiers and maintain a record of all relevant attributes of the CI including the owner. Each CI is assigned to a related CI class, that groups similar CIs together.

Note: The tables below containing service assets and configurable items is not yet complete. This list will be updated as needed to support the SACM process.

Service Assets (SAs): requires version control and change notification, not change control)	
Category	Method and/or Situation
Project Documentation (Project Management Plans and Processes)	The Project Management Office (PMO) has decided that project documentation would not be considered a CI under change control. Project documentation is a service asset. The authority for making changes to project documentation is at the subject matter expertise (SME) level and does not require going through a CCB to approve changes.
User Documentation (user manuals, user guides, cheat sheets, help or FAQs)	Each service team is responsible for the development, revision, management, approval and storage of user documentation.
Vendor Documentation (Sprint Status Report)	Contract Management will handle the approval of the status reports in comparison to contractual obligations.
Technical Documentation produced by the State (architecture roadmap, data roadmap, infrastructure roadmap)	Technical teams are responsible for the development, revision, management, approval and storage of technical documentation.
Enterprise-wide Hardware/Software/Licenses purchased by the State	Goes through a workflow approval process for installation and configuration.
Standard Hardware/Software/Licenses purchased by the Vendor	Vendors are expected to bring their own laptops and set up their own development environments. If they bring software, the State needs a plan to transfer the software and licenses to the State.
Role Based Access	Role based access is a service asset that does not require change control but goes through a formal workflow approval process.

Table 3-3 Service Assets Overview

Configuration Items (CIs): requires change and version control	
Category	Method and/or Situation
Cost Increase	Requires additional cost to implement change
Digital Service Standards	Adverse impact to planned standards / compliance
Business Process and System Functionality	Requires modification to system functionality or business process
Schedule	Requires extension of scheduled deliverable dates

Configuration Items (CIs): requires change and version control	
Category	Method and/or Situation
Bidders' Library	All documentation within the bidders' library
Source Code	• IBM manages source code for the CWS/CMS independently. • CWS-NS source code will be deposited into GitHub with appropriate version control. • Becomes a CI when the source code is moved into the State's environment(s). • The GitHub repositories will have the mechanism to control who has access to the code and note who has checked the code in and out. Control of software code varies depending on environment, action, and responsible party.
Systems Development	• Test Automation Scripts • Ansible scripts • Anything that has a high impact to the system, application, or infrastructure. • Patch - A patch is a specific fix that addresses a single problem or a change resulting from a legislative or policy/program change • Minor Release - A minor release is a release for system fixes (inclusive of all previous fixes and patches), system enhancements, and new functionality. Minor releases occur between major releases. • Major Release - A major release includes new as well as improved features and functionalities.
Non-Standard Hardware/Software/Licenses purchased by the Vendor	If a vendor brings non-standard HW/SW that they want to install in a State environment, the State must allow for a formal change control process.
Configuration Baseline	A configuration baseline of CIs must be taken before performing an implementation/release (into system, acceptance test and production) in a manner that can be used for subsequent checking against actual deployment.

Configuration Items (CIs): requires change and version control	
Category	Method and/or Situation
Snapshots	• Snapshots will be taken of the production environment prior to being rolled out. In the progress of identify anomalies, a final snapshot is created and compared with the target baseline to verify any discrepancies. • Snapshots will be taken before every release or major / minor upgrade, update, bulk update, schema changes and database migration and with a retention period of 90 days.

Table 3-4 Configuration Items Overview

3.4 Control

CIs must follow a strict process for being added, changed, and removed from our CMS or CMDB (Configuration Management system or Configuration Management Database). The goal is to ensure that changes don't occur without following approved procedures such as change management. We will create policies and procedures for controlling software licenses, controlling access to facilities and systems, and how we will properly capture the baseline of assets and CIs before releases, to give an accurate way to verify the success of deployments.

3.5 Status Accounting and Reporting

Throughout the lifecycle of every CI we will keep track of the complete status — including what changes have been proposed, the status of approved changes, and other relevant information related to the CI. Being able to view and provide status reports gives you important insight into both the current and historical state of your CI's.

Reports include many different types of information, such as an inventory of CIs and their baseline configurations, itemizations of any unauthorized Cis, updates on recent changes or exceptions, or an itemization of hardware and software assets.

Service Asset & CI Statuses:

- Purchased
- Received
- Awaiting approval of CR
- Retired/Withdrawn
- Live (CI may be used)
- Disposed
- Spare

- Under Repair
- In Build/Dev
- In Test

3.6 Verification and Audit

Regular reviews and audits are essential, and assures that data is accurate on all of CIs (Configuration Items). Verification is an ongoing activity, consistently ensuring that the CMDB (Configuration Management Database) accurately reflects all CIs. An audit is a more formal deep dive to confirm not only that records are accurate, but that processes are being followed and standards are being met.
If there is a high incidence of unauthorized CIs detected, the frequency of configuration audits will be increased.
The activities include a series of reviews or audits to:

- Ensure there is conformity between the documented baselines and the actual business environment to which they refer
- Verify the physical existence of CIs in the organization or in the DML (Definitive Media Library) and the locked site, the functional and operational characteristics of CIs, and to check that the records in the CMS (Configuration Management System) match the physical infrastructure
- Check that release and configuration documentation is present before making a release
- Review the version control structures in active use
- Verification and audit will be carried out:
- Before a release to ensure the environment is as expected
- Following recovery from a “disaster” and after “return to normal”
- Shortly after changes to the CMDB (addition of fields), as well as changes to database itself
- Before and after changes to the CIs
- At planned and random intervals
- In response to the detection of unauthorized CIs

3.7 Managing information

As part of the asset and configuration management process, we will regularly back up the CMS (Configuration Management System), keep detailed records about archived and historical CI (Configuration Item) versions, and take appropriate measures to ensure data integrity across the lifecycle.

CM Process
Configuration Management utilizes the typical request, track, build, test, approve, implement, control and monitor approach in order to utilize the processes already in place.

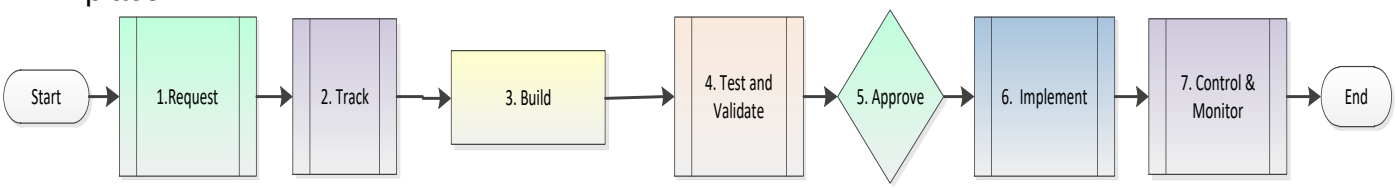


Table 3-1 CM Process Overview

The high-level steps to the CM process are listed in table 3-2.

Step #	Description	Role
1.	**Request:** Analyze if a change is needed. Follow Change Management Process by creating a Change Request (CR) for adding, changing, or retiring/archiving a CI.	Change Initiator for CR CCB for Approval
2.	**Track in the Configuration Management Database (CMDB):** Configuration Analyst updates the CI with status from CR if tool does not automatically do so and ensures there is a back-up/snapshot prior to Change Control Board (CCB) approval.	Configuration Analyst or Project Librarian
3	**Build:** Build or implement changes	Change Developer
4.	**Testing and Validation:** Test process and validate assumptions and constraints	Change Developer
5.	**Approve:** Approve the change for release after successful completion of testing and validation	CCB for Approval
6.	**Release & Implementation:** Train, roll-out, and support	Change Implementer
7.	**Control & Monitor:** Update CI, update status, put back under change control, monitor any issues or unauthorized changes, and report.	Configuration Analyst

Table 3-2 SACM Process Overview Steps

3.8 Decommissioning Assets and Configuration Items

Decommissioning a Configuration Item requires a change request. Follow the Change Request Process and indicate the reason for retiring the asset.
All software that is no longer to be used in state service shall be disposed of in an appropriate manner.

The Configuration Analyst is responsible for disposal of software (organization/classification).

Documentation will be retained per the retention policy and by utilizing the CWDS Document Management Process.

3.9 Controlled Assets

We will use the example of Elasticsearch to describe how CIs (Configuration Items) are controlled under the SACM plan. Elasticsearch versions and relevant dependencies will be logged as a CI in the CMDB (Configuration Management Database), and all changes will be logged for reporting, verification, and auditing. The repository for Elasticsearch versions is an AWS (Amazon Web Services) S3 Bucket. Our Ansible deployment scripts pull the versions out of the S3 bucket based on the version number that is required for a specific environment. If an upgrade is needed to a newer version of Elasticsearch, a user story is created to perform the necessary work. The user story can be accepted or rejected by the product owner. If accepted, the upgrade can be implemented, and a new record is added into the CMDB. If rejected, the story returns to an in-progress state until it can be re-evaluated by the product owner. Changes to controlled assets will always include a change request.

3.10 Recording & Reporting

We are using an excel workbook to provide status information and deployment history to project staff. The workbook is updated as needed, and resides here on the CWDS DevOps Engineering SharePoint site.

3.11 CI Changes

All changes against CIs will be logged within the Change Request, including a detailed description and an analysis of impact for the specific change, as well as other potentially important information relating to the change. Please refer to the CWDS Change Management Plan for specific processes relating to change requests.

3.12 Tools & Procedures

Excel is the current tool used for tracking changes against CIs, and will be used until a CMS is implemented. These requirements are for an integrated tool set, not for a single tool.

The tool shall:

- Provide security controls to limit access to CWS records on a need-to-know basis
- allow CI attributes fields; such as unique identifier, type, name, description, version, location, supply date, license details, owner, status and others depending on the type of CI

- Produce management reports from any of the data fields that are held within the CMS without the need to purchase additional products or consultancy services
- Facilitate the production of management reports from historical records
- Provide an audit trail for record information and updates; i.e., IDs of individuals or groups opening, updating and closing records; dates and times of status and activities updates, types of activities
- Automate notification and escalation to keep IT and users informed of potential issues or progress
- Allow old CI records to be deleted or archived
- Support CIs with different formats for model numbers, version numbers and copy numbers, such as for hardware (e.g., serial no. for Dell, HP & IBM) MS Office software and documentation such as ISBN number on books or an edition number on an SLA)
- Automatically validate input data to ensure that all CI record details are unique
- support the addition of the relationship with other CIs at the time of entering the record
- Show the current status of any CI, such as 'live' or 'withdrawn'
- support the control of software through all stages of its lifecycle, from design stage through to live operational running
- Support the management and use of baselines that can be used for reverting to trusted versions
- Verify that correct and authorized versions of CIs exist
- Support linking definitive media libraries to the CMS/CMDB
- Allow CI inventory reports to be produced to facilitate configuration audits
- Maintain the historic details of all CIs, such as installation date, records of changes and locations
- Display data in the form of models and maps relationships between CIs
- Support the hierarchic and networked relationships between CIs; i.e, a capability that is needed for management reporting, managing incidents, problems and changes (impact analysis)
- Automatically identify other CIs affected when any CI is the subject of an incident, problem, known error record and RFC
- Automatically update the version number of a CI if the version number of any component CI is changed
- Provide a documented procedure and checklist for manual updates to configuration data, which are also recorded in a change log

- Prevent CI records being updated without appropriate change approvals and procedures being followed, including documentation
- produce a report showing unauthorized additions

4 Appendix A: Glossary

Term	Definition
CMS (Configuration Management System)	A set of tools and databases that are used to manage Configuration data. The CMS also includes information about Incidents, Problems, Known Errors, Changes and Releases; and may contain data about employees, Suppliers, locations, Business Units, Customers and Users. The CMS includes tools for collecting, storing, managing, updating, and presenting data about all Configuration Items and their Relationships. The CMS is maintained by Configuration Management and is used by all IT Service Management Processes.
CMDB (Configuration Management Database)	A logical database containing all relevant information about IT infrastructure components, as well as the relations between those components. Each component is referenced in the CMDB as a Configuration Item (CI).
CI (Configuration Item)	Any component or other service asset that needs to be managed in order to deliver an IT service. Information about each configuration item is recorded in a configuration record within the configuration management system and is maintained throughout its lifecycle by service asset and configuration management. Configuration items are under the control of change management. They typically include IT services, hardware, software, buildings, people and formal documentation such as process documentation and service level agreements.
SA (Service Asset)	Any resource or capability of a service provider.
CN (Change Notification)	Change Notification is a notification to stakeholders that details important information related to a change.
CR (Change Request)	A formal proposal for a change to be made. It includes details of the proposed change, and may be recorded on paper or electronically. The term is often misused to mean a change record, or the change itself.
Status	The name of a required field in many types of record. It shows the current stage in the lifecycle of the associated configuration item, incident, problem etc
History	Information about all changes made to a configuration item during its life. Change history consists of all those change records that apply to the CI.
Attributes	A piece of information about a configuration item. Examples are name, location, version number and cost. Attributes of CIs are

	recorded in a configuration management database (CMDB) and maintained as part of a configuration management system (CMS). See also relationship; configuration management system.
Dependencies	The direct or indirect reliance of one process or activity on another.
Configuration Item Class	All CIs with the same nature are grouped within classes. All CIs within a CI class have the same behavior, for example the lifecycle. Some typical CI Classes: Application, Network Gear, Server, Documentation.
Configuration Audit Report	Results from a formal inspection and verification to check whether a standard or set of guidelines is being followed, that records are accurate, or that efficiency and effectiveness targets are being met.
ITIL (Information Technology Infrastructure Library)	ITIL, formally an acronym for Information Technology Infrastructure Library, is a set of detailed practices for IT service management (ITSM) that focuses on aligning IT services with the needs of business. In its current form (known as ITIL 2011), ITIL is published as a series of five core volumes, each of which covers a different ITSM lifecycle stage. ITIL describes processes, procedures, tasks, and checklists which are not organization-specific, but can be applied by an organization for establishing integration with the organization's strategy, delivering value, and maintaining a minimum level of competency. It allows the organization to establish a baseline from which it can plan, implement, and measure. It is used to demonstrate compliance and to measure improvement.
Relationship	A connection or interaction between two people or things. In business relationship management, it is the interaction between the IT service provider and the business. In service asset and configuration management, it is a link between two configuration items that identifies a dependency or connection between them. For example, applications may be linked to the servers they run on, and IT services have many links to all the configuration items that contribute to that IT service.