



MANUAL TRANSMITTAL

Department of the Treasury
Internal Revenue Service

2.150.2

APRIL 18, 2025

EFFECTIVE DATE

(04-18-2025)

PURPOSE

- (1) This transmits revised IRM 2.150.2 Configuration Management (CM) Process.

MATERIAL CHANGES

- (1) IRM 2.150.2.1.1.1 (5) Process Description, moved to 2.150.2.2.5.2 (2) Configuration Identification..
- (2) IRM 2.150.2.1.3 Roles and Responsibilities, added new CMDB roles.
- (3) IRM 2.150.2.1.6 (1) Terms and Acronyms, added "Technical Service".
- (4) IRM 2.150.2.1.8 Training, updated training courses.
- (5) IRM 2.150.2.2.2 Inputs, replaced Configuration Item with IT Asset.
- (6) IRM 2.150.2.2.3 Outputs, added Configuration Item.
- (7) IRM 2.150.2.2.5.1 (2) Management Planning, updated exclusions to ACIO CM Plan and SDLC approach.
- (8) IRM 2.150.2.2.5.2 (3) Configuration Identification, updated CM baseline tables based on OneSDLC Compliance Checklist and Technology Insertion Checklist.
- (9) IRM 2.150.2.3 Configuration Management in the Cloud, added new content.
- (10) IRM 2.150.2.5.2 Document Versioning Guidelines, revised instructions for clarity.

EFFECT ON OTHER DOCUMENTS

IRM 2.150.2 dated April 2, 2024 is superseded.

AUDIENCE

The Configuration Management Process is applicable to all Information Technology (IT) organizations, contractors, and other stakeholders having responsibility for configuration, management, oversight, and successful day-to-day operations of the IRS IT enterprise hardware, software, and applicable documentation.

Rajiv Uppal
Chief Information Officer

2.150.2

Configuration Management (CM) Process

Table of Contents

2.150.2.1 Program Scope and Objectives

2.150.2.1.1 Background

2.150.2.1.1.1 Process Definition

2.150.2.1.2 Authority

2.150.2.1.3 Roles and Responsibilities

2.150.2.1.4 Program Management and Review

2.150.2.1.5 Program Controls

2.150.2.1.5.1 Controls

2.150.2.1.5.2 Metrics

2.150.2.1.5.3 Tailoring Guidelines

2.150.2.1.6 Terms and Acronyms

2.150.2.1.7 Related Resources

2.150.2.1.8 Training

2.150.2.2 Configuration Management Process

2.150.2.2.1 Main Process Diagram

2.150.2.2.2 Inputs

2.150.2.2.3 Outputs

2.150.2.2.4 Activities

2.150.2.2.5 Guidelines for Configuration Management

2.150.2.2.5.1 Management & Planning

2.150.2.2.5.2 Configuration Identification

2.150.2.2.5.3 Configuration Control

2.150.2.2.5.4 Configuration Status Accounting

2.150.2.2.5.5 Configuration Verification & Audit

2.150.2.3 Configuration Management in the Cloud

2.150.2.4 Configuration Identification Index (CII) and Document Versioning Guidelines

2.150.2.4.1 CII Guidelines

2.150.2.4.2 Document Versioning Guidelines

2.150.2.5 Configuration Documentation Classification Guide

2.150.2.1
(04-18-2025)
**Program Scope and
Objectives**

- (1) **Purpose.** This IRM section describes the formal process for implementing the requirements of the Configuration Management (CM) process. It provides an operational definition of the major components of the process and the requirements for each of the process components. This document also describes the logical arrangements of steps that are essential to successfully completing the process and achieving its desirable outcome. Configuration Management is a systems engineering process for establishing and maintaining consistency of the software product's performance, functional, and physical attributes with its requirements, design, and operational information ensuring consistency among physical and logical assets in an operational environment. Configuration Management is also one of the components in the Information Technology (IT) Service Support under the IT Infrastructure Library (ITIL) framework. It is an IT Service Management (ITSM) process, and its primary goal is to identify, maintain, and verify information on IT configuration assets, or configuration items (CIs), in a configuration management database (CMDB) that is required to deliver an IT service. It covers identifying, recording, and reporting of IT components, including their versions, constituent components, and relationships. Items that should be under the control of Configuration Management include hardware, software, and associated documentation. Additionally, where IT security and IT operations meet is where it blends the key practices and establishes a security-focused Configuration Management (called Secure Configuration Management (SecCM)), such as vulnerability assessment, remediation and configuration assessment. For the purposes of this document, the process scope and objective for this Configuration Management is primarily based on supporting the software development and operations environment. Secure Configuration Management process and requirements are documented in IRM 10.8.1 Information Technology (IT) Security, Policy and Guidance.
- (2) **Audience.** The Configuration Management process is applicable to all IT organizations, contractors, and other stakeholders having responsibility for configuration, management, oversight, and successful day-to-day operations of the IRS IT enterprise hardware, software, and applicable documentation.
- (3) **Policy Owner.** Demand Management & Project Governance (DMPG) Division, within Enterprise Operations (EOps) - IT.
- (4) **Program Owner.** Governance & Resource Management Branch, within DMPG - EOps - IT.
- (5) **Primary Stakeholders.** IT organizations having responsibility for establishing an internal or local Configuration Management process and/or managing, changing, and controlling their IT system and/or system components are stakeholders in the Configuration Management process.
- (6) **Contact Information.** To recommend changes or to make any suggestions to this IRM section, email the IT CM Program Management Office (PMO): *it.cm.process@irs.gov*

2.150.2.1.1
(08-19-2020)
Background

- (1) Information systems are typically dynamic, causing the system state to change frequently because of upgrades to hardware, software, or modifications to the surrounding environment in which a system resides. Industry standards and best practices including those issued by the Government Accountability Office and the Office of Management and Budget stress that information systems (e.g., general support systems, major applications, and minor applications) must document and assess the potential impact that proposed system changes

may have on the operational processes and security posture of the system. The IT industry best practices recognize this as an essential aspect of effective system management, as well as being part of the continuous monitoring and maintenance of security accreditation of federal systems.

- (2) Configuration Management is a critical control for ensuring the integrity, security, and reliability of the IRS information systems. Absent a disciplined process for controlling configuration changes, management cannot be assured that its systems will operate as intended, or that systems' maintenance will be performed in a cost-effective or timely manner.

2.150.2.1.1.1
(04-18-2025)

Process Definition

- (1) **Process Description.** The information set below describes the characteristics of Configuration Management.

- Configuration Management is the process responsible for providing accurate and complete configuration information about related software and IT infrastructure components, including their attributes and relationships, to support other software engineering and service management processes.
- Businesses require quality IT services to be provided in an economical manner. To be both efficient and effective, all organizations need to have control of their IT services and infrastructure.
- Configuration Management provides a model of a service's IT infrastructure or the entire IT infrastructure by identifying, managing, maintaining, and verifying the configuration items in the operational environment, their attributes and their relationships with other configuration items and the services they enable and support.

- (2) **Process Goal.** The goal of Configuration Management is to provide accurate information on the current state of the software and IT infrastructure, the attributes of IT service-related components, and their relationships to enhance the effectiveness of other software development and service management processes. This is accomplished through the identification, control and verification of those items declared to be within scope of the process. The key to this process is the identification of the relationships that exist among configuration items.

- (3) **Process Objectives.** Process objectives describe material outcomes that are produced or achieved by the process. The following is a list of objectives for this process:

- Help provide accurate assessment of the risk and impact of a change to a configuration item.
- Improve the assessment of the impact of a configuration item failure by readily identifying the services that it supports.
- Provide accurate configuration information to other software development and service management processes.
- Provide a snapshot of a known state of the IT environment or baseline of configuration items.
- Create a standard method for introducing, updating and tracking components or aggregated configuration items in the IT environment.
- Provide a standard automated technology and location for storing information about the configuration item.

- Ensure that all changes to the software and IT infrastructure, configuration items or component configuration item, that need to be managed are reflected on the software libraries and repositories.
- (4) **Disciplines.** Configuration Management is performed within the context of the 5 common disciplines that are applicable in development, operations, and security environment, as illustrated in the figure below. Although Configuration Management has a shared goal and objective in identifying, managing, and controlling the configuration items across the environments, its purpose and application is slightly different but complimentary.
- a. Under the software development environment, Configuration Management ensures that the development and release processes are controllable and repeatable through identification, tracking, and protecting the project's deliverable or software products from unauthorized change.
 - b. Under the operational environment, Configuration Management ensures the integrity of configurations required to control the services and IT infrastructure by establishing and maintaining accurate and complete configuration item records in a CMDB.
 - c. Under the security environment, Configuration Management ensures management and control of secure configurations for an information system to enable security and facilitate management of risk.

The following figure illustrates the 5 Common Disciplines.

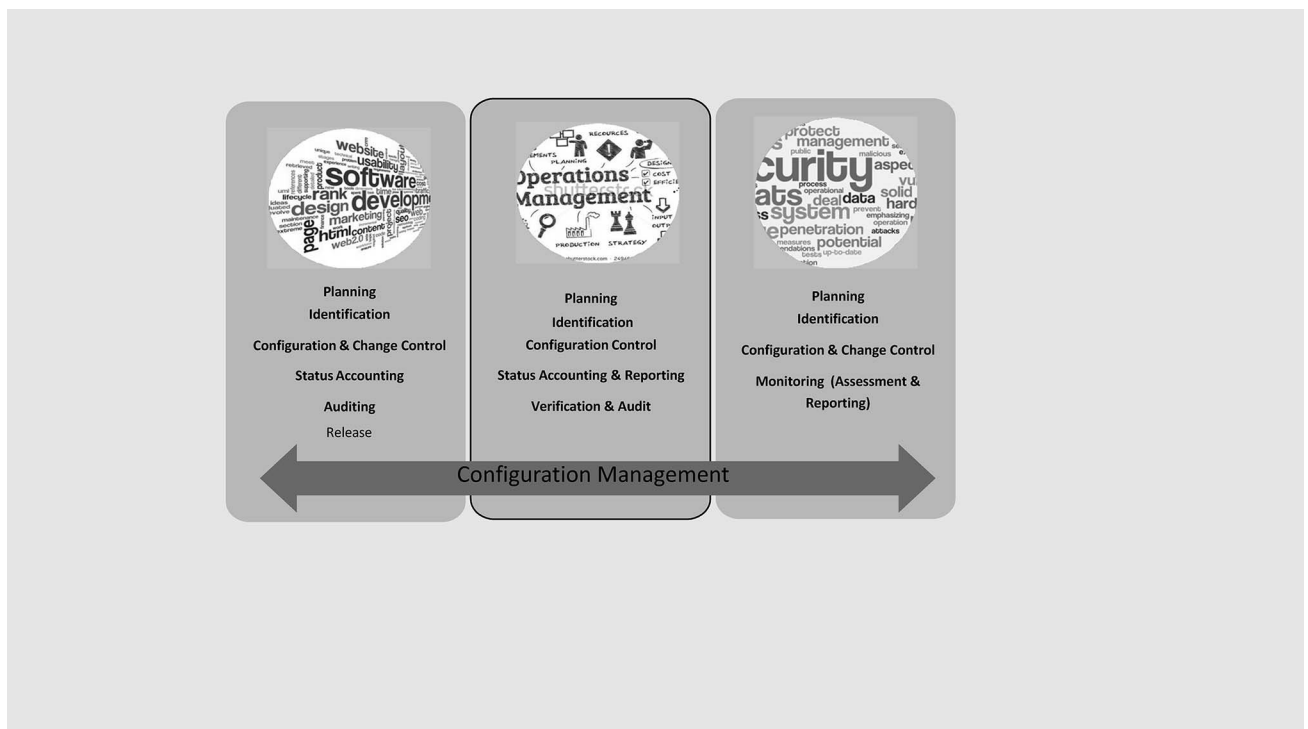


Figure 2.150.2-1

2.150.2.1.2
(08-19-2020)

Authority

- (1) The IRMs listed below establishes the authority for this process to be established:

- IRM 2.150.1 **Configuration Management Policy**
- IRM 2.125.1 **Change Management Policy**
- IRM 10.8.1 **Information Technology (IT) Security, Policy and Guidance**

2.150.2.1.3
(04-18-2025)

Roles and Responsibilities

- (1) Each process defines at least one role. Each role is assigned to perform specific tasks within the process. The responsibilities of a role are confined to the specific process. They do not imply any functional standing within the hierarchy of an organization. For example, the process manager role does not imply the role is associated with or fulfilled by someone with functional management responsibilities within the organization. Within a specific process, there can be more than one individual associated with a specific role. Additionally, a single individual can assume more than one role within the process although typically not at the same time. The table below describes the generic Configuration Management and related roles and responsibilities.

| Process Role | Description |
|------------------|--|
| CM Process Owner | <p>The CM Process Owner is the single point of contact for the process at the enterprise level and is accountable for the overall quality of the process, ensuring that the process is performed as documented and is meeting its objectives. The CM Process Owner's responsibilities include sponsorship, design, review and continual improvement of the process and its metric. Specific responsibilities include—</p> <ul style="list-style-type: none"> • Defines the overall scope, mission, goal, and objectives of the process • Ensures consistent execution of the process across the IT organization • Ensures that the process, roles, responsibilities and documentation are regularly reviewed and audited • Reports on the effectiveness of the process to senior leadership • Accountable for implementation and review of improvement actions • Ensures that sufficient resources are in place to support implementation of the process • Ensures all relevant staff have the required training in the process and are aware of their role in the process • Ensures that the process is in alignment with the process automation tool(s) • Resolves any process and cross-functional (departmental) issues |

| | |
|-----------------------------------|---|
| CM Process Manager | <p>The CM Process Manager supports the CM Process Owner and is responsible for the operational management of the process. The CM Process Manager's responsibilities include planning and coordination of all activities required to execute, monitor and report on the process. Specific responsibilities include—</p> <ul style="list-style-type: none"> • Designs, develops, and manages improvements to the process; including plans, principles and its implementation • Provides training and communication on the standards, policy, and process to Configuration Management stakeholders • Plans, facilitates and organizes reviews, assessments, and audits on the process • Ensures alignment of the Configuration Management tools with the process; including evaluating Configuration Management tools and their design, requirements, proposed changes, and implementation • Develops, coordinates, and maintains the interfaces to other processes • Defines process metrics for measurement, reporting, and improving the process • Escalates any issues with the process • Maintains scope of the process, function, and configuration items that are to be controlled, and information that is to be recorded • Ensures that configuration data is available when and where it is needed to support other IT processes • Agrees on the structure of the CMDB, including CI Types, naming conventions, required and optional attributes and relationships • Designs and generates configuration status reports; including management reports |
| Configuration Control Board (CCB) | <p>An authoritative body, consisting of senior level managers or executives, that is responsible for managing and authorizing changes to baseline configuration items and components. The CCB control major issues such as schedule, function, and configuration of the system as a whole.</p> <p>Note: A CCB may be composed of lower-level CCBs, called Software CCB (SCCB) , consisting of technical management that is responsible for control and management of configuration items within their area of their scope and control.</p> |
| Technical Review Board (TRB) | <p>A technical team that supports the CCB or SCCB and is responsible for evaluating proposed system change requests including recommendations of alternatives or evaluation of alternatives.</p> |
| Configuration Manager | <p>Responsible for identifying configuration items and ensures Configuration Management is followed.</p> |

| | |
|----------------------------|---|
| CI Owner | <p>The CI Owner is an ITIL role and is accountable for all activities that directly affects their configuration items. This role is also responsible for all ITSM process activities associated with the maintenance and support of the configuration item. Specific responsibilities include –</p> <ul style="list-style-type: none"> • Ensures that all their configuration items are recorded, and the associated attributes and relationship information is accurate • Ensures configuration audits are performed on the CMDB • Provides input to the CI Librarian/ Analyst into what attributes and relationships need to be tracked within the CMDB • Accountable for correcting errors associated with their configuration items |
| CMDB Administrator | <p>The CMDB Administrator has broader permissions to manage the entire CMDB structure, including defining its scope, designing the data model, and overseeing overall data integrity across all configuration items.</p> |
| CMDB Service Administrator | <p>The CMDB Service Administrator focuses on managing specific configuration items within the CMDB, and is primarily responsible for keeping their assigned CIs accurate and updated.</p> |
| CMDB Editor | <p>The CMDB Editor have create, read, update, and delete access to CMDB records but are not allowed to change policies such as data manager, class manager, etc.</p> |
| CMDB User | <p>The CMDB User have read-only access to CMDB data and basic user interfaces such as reports, dashboard, etc.</p> |
| Configuration Auditor | <p>The Configuration Auditor supports the CM Process Manager and is responsible for conducting and/or providing oversight to a configuration audit. Specific responsibilities include –</p> <ul style="list-style-type: none"> • Conducts and/or oversees periodic CMDB audits to check the accuracy, completeness, compliance and security of records against the baseline • Conducts reviews and assessments on the CM process to ensure process compliance • Consolidates the observations and non-conformances • Provides reports to the CM Process Owner, CM Process Manager, and other relevant stakeholders • Ensures that reviews are performed by the CMDB Editor and CI Owners • Validates accuracy of CMDB data and report discrepancies • Ensures audit reports are distributed to the CM process stakeholders and CI Owners and recommends improvements |
| Software Developer | <p>Responsible for development and maintenance of computer applications. This includes:</p> <ul style="list-style-type: none"> • Coordinates identified issues/problems/defects with other testing or project stakeholders or provide a workaround • Documents all coding • Participates in peer reviews of coding and documentation • Performs unit testing on the created/changed code • Notifies project manager of testing status • Provides appropriate artifacts to the next phase of testing/deployment • Creates, updates, and maintains appropriate artifacts for testing phases |

| | |
|----------------------------|--|
| Software Tester | Responsible for testing, analyzing, compiling data, and generating reports. This includes: <ul style="list-style-type: none"> • Creates test related work products (test cases/scripts, test data sets, etc.) • Prepares any required reporting documentation for the respective testing activities • Executes and documents test activities • Manages testing requirements, creates, duplicates, and executes test cases/scripts, identifies and documents testing problems, and reports testing status • Analyzes appropriate documentation to extract project requirements |
| Software Quality Assurance | Responsible for providing an independent review and evaluation of software products through its software processes to ensure that the product or configuration item conforms and delivers to its intended purposes. |
| Business Lead | Responsible for creating, communicating, coordinating, and interpreting the business requirements, including approving various artifacts |
| Project Manager | The Project Manager is responsible for the planning and execution of a project. Specific responsibilities include- <ul style="list-style-type: none"> • ensures that the project/product is developed within the defined scope, schedule, and cost • monitors the progress of the development and addresses issues in the Configuration Management process • generates reports and provide status of the project deliverable and software system • ensures that the policies and processes are followed for changes |
| Project Stakeholder | Members of a project such as project sponsor, process owners, executives, customers, and users. |
| Product Manager | The Product Manager is responsible for managing the Product Roadmap and supporting the Product Team as needed. They communicate the state of the initiative and manage expectations from leadership and the team. The Product Manager also collaborates with the Product Owner and Product Sponsor to validate assumptions and refine strategies to achieve product Objectives and Key Results. |
| Product Owner | The Product Owner manages and prioritizes the Product Backlog and Mid-range Backlog. They are responsible for defining stories and prioritizing the Iteration Backlog to streamline execution. The Product Owner has a significant role in maximizing the team's value by ensuring stories meet the user's needs and comply with the Definition of Done. They collaborate with the Product Manager, Product Sponsor and Business to validate assumptions, refine objective and key results, and prioritize backlog items to achieve product goals. |

Note: Some of the roles may not be applicable for other IT projects.

2.150.2.1.4
(08-19-2020)

**Program Management
and Review**

- (1) The IT CM PMO shall manage and evaluate the process based on the following guiding principles:
 - a. **Process Management.** Configuration Management will have a single Process Owner and a separate Process Manager, responsible for implementing and ensuring adherence to the process. The process will be reviewed regularly to ensure that it continues to support the business requirements of the enterprise. Process metrics will be focused on providing relevant information as opposed to merely presenting raw data.
 - b. **People.** Roles and responsibilities for the process must be clearly defined and appropriately staffed with people having the required skills and training. The mission, goals, scope and importance of the process must be clearly and regularly communicated by upper management to the staff and business customers of IT. All IT staff (direct and indirect users of the process) shall be trained at the appropriate level to enable them to support the process. It is imperative that people working in, supporting or interacting with the process in any manner understand what they are supposed to do. Without that understanding Configuration Management will not be successful.
 - c. **Process.** Modifications to the process must be approved by the Process Owner. The design of the process must include appropriate interfaces with other processes to facilitate data sharing, escalation and workflow. The process must be capable of providing data to support real-time requirements as well as historical/trending data for overall process improvement initiatives. The process must be fully documented, published and accessible to the various stakeholders of the process. The process will be reviewed on a periodic basis to ensure it continues to support organizational goals and objectives (continuous improvement). The process must include Inputs, Outputs, Controls, Metrics, Activities, and Roles and Responsibilities along with documented process flows. The process will be kept straight forward, rational, and easy to understand. The process must meet operational and business requirements.
 - d. **Technology and Tools.** All tools selected must conform to the enterprise architectural standards and direction. Existing in-house tools and technology will be used wherever possible, new tools will only be entertained if they satisfy a business need that cannot be met by current in-house tools. The selection of supporting tools must be process driven and based on the requirements of the business. Selected tools must provide ease of deployment, customization and use. Automated workflow, notification and escalation will be deployed wherever possible to minimize delays, ensure consistency, reduce manual intervention and ensure appropriate parties are made aware of issues requiring their attention. Technology and tools should be used to augment the process capabilities, not become an end themselves.

2.150.2.1.5
(08-19-2020)

Program Controls

- (1) Program controls are driven by the policies and guiding principles on how the process will operate.

2.150.2.1.5.1
(04-18-2025)

Controls

- (1) Controls provide direction on the operation of processes and define constraints or boundaries within which the process must operate.

| Name | Description |
|----------------------------|---|
| Baselines | Documented agreed descriptions of the attributes and/or specifications of a configuration item, at a point in time, which serves as the basis for defining change. |
| Change Management Policies | Policies and mandates for change control of configuration items. |
| Configuration Audits | An examination of a configuration item to determine whether it conforms to its design and requirements including the integrity of its record. |
| Configuration Reports | Status of configuration items reported and distributed on a defined frequency. |
| Model | A defined structure and approach to recording relationships between configuration items that includes the level of detail that the organization wants to trace the relationships. |
| Plan | A documented plan that will define the scope, objective, resources, change authority, and activities for Configuration Management. |
| Taxonomy | Defined standards for naming and classifying configuration items including terms and definitions. |

2.150.2.1.5.2
(08-19-2020)
Metrics

- (1) Metrics are used for the quantitative and periodic assessment of a process. They should be associated with targets that are set based on specific business objectives. Metrics provide information related to the goals and objectives of a process and are used to take corrective action when desired results are not being achieved and can be used to drive continual improvement of process effectiveness and efficiency.
- (2) Management will regularly set targets for process performance, gather quantifiable data related to different functions of Configuration Management, and review that data to make informed decisions and take appropriate corrective action, if necessary. All measurements must have a defined data dictionary, map to the organizational strategic goals, and be documented in the Configuration Management Process Measurement Plan.
- (3) Enterprise and local Configuration Management processes, including Configuration Management tool owners, must produce metrics and measurement reports to measure the effectiveness and efficiency of the Configuration Management process.

2.150.2.1.5.3
(08-19-2020)
Tailoring Guidelines

- (1) The tailoring guidelines identify the allowable variations of the IT organization's standard process as needed for adjustments (adding, deleting, modifying) relative to specific operational or functional needs of another organization. Process tailoring is about roles and procedures, not the standard process or major activities defined in this process. All tailoring request, with supporting rationale, must be submitted in writing to and approved by the CM Process Owner.

2.150.2.1.6
(04-18-2025)

(1) This table lists commonly used terms in Configuration Management.

Terms and Acronyms

| Term | Definition |
|--|--|
| Application Service | An IT service that provides the application capabilities required to support business capabilities. It consists a set of interconnected applications and hosts which are configured to offer a service to the organization, such as email or web portal. |
| Business Service | A service that is delivered to the business customers by business units, such as collections and exam. |
| Configuration Audit | Audits conducted to confirm that Configuration Management records and configuration items are complete, consistent, and accurate. |
| Configuration Item | A collection and combination of hardware, software, and documentation that is used to deliver a product or service. |
| CI Attributes | Physical characteristics of the configuration item that describes and distinguishes it from other configuration items. Configuration item attributes are used to identify, manage, and report on its status. |
| CI Relationships | Dependency of one configuration item to one or more configuration items. Each relationship has a starting and target point. |
| CI Type | A broad classification of different IT assets under Configuration Management which configuration items are categorized. See Figure 2.150.2-1. |
| CI Subtype | A detailed classification of a configuration item. For example, a computer configuration item is further divided into the following CI Subtypes: mainframe, server, appliance. |
| Configuration Management Database (CMDB) | A CMDB is a database that contains all relevant information about the hardware and software components used in an organization. It includes the components, the IT services they support and the relationships between those components. A CMDB provides an organized view of configuration data and a means of examining that data from different perspectives. |
| Functional Configuration Audit (FCA) | Audits conducted on baseline components to verify that the development of a configuration item has been completed satisfactorily, that the item has achieved the functional and quality attribute characteristics specified in the functional or allocated baseline, and that its operational and support documents are complete and satisfactory. |
| Infrastructure Service | An infrastructure service is a service that is required for IT to deliver the business services but is not directly consumed by the customer; infrastructure services are usually consumed by other applications or services. Services such as backup, directory management (LDAP/AD), and security are common examples of infrastructure Services. |
| Model | A model is a reusable collection of configuration item instances that define a business entity, such as a business service or line of business with its supporting applications and infrastructure. |

| | |
|------------------------------------|---|
| Physical Configuration Audit (PCA) | Audits conducted on baseline components to verify that a configuration item, as built, conforms to the technical documentation that defines and describes it. |
| Service | <p>A means of delivering value to customers by facilitating outcomes that customers want to achieve, without ownership of specific costs and risks. A service consists of the following:</p> <ul style="list-style-type: none"> • Has a customer • Has a service provider • Delivers value • Enhances performances or reduces constraints • Increase the probability of achieving desired outcomes |
| Technical Service | Technical service is associated with service owners and are typically layered under one or more business application services. A technical service may have one or more technical service offerings. |

(2) This table lists acronyms used in this IRM.

| Acronym | Description |
|---------|---|
| CCB | Configuration Control Board |
| CI | Configuration Item |
| CM | Configuration Management |
| CMDB | Configuration Management Database |
| COTS | Commercial-Off-the-Shelf |
| CSP | Cloud Service Provider |
| FCA | Functional Configuration Audit |
| GOTS | Government-Off-the-Shelf |
| ITIL | Information Technology infrastructure Library |
| ITSM | Information Technology Service Management |
| MSP | Managed Service Provider |
| PCA | Physical Configuration Audit |
| RfC | Request for Change |
| SecCM | Secure Configuration Management |
| SCM | Software Configuration Management |
| TRB | Technical Review Board |

2.150.2.1.7
(08-19-2020)

Related Resources

- (1) The following lists the primary sources of references used in the development of Configuration Management.

- IRM 2.5.1 *Systems Development*
- IRM 2.22.1 *Unified Work Request (UWR) Process*
- IRM 2.31.1 *One Solution Delivery Life Cycle (OneSDLC) Guidance*
- IRM 2.125.2 *Change Management Process*
- IRM 2.127.1 *IT Test Policy*
- IRM 2.127.2 *IT Testing Process and Procedures*
- IRM 10.8.1.4.5 *CM-01 Configuration Management Policy and Procedures*
- IEEE-828-2012 Standard for Software Configuration Management in Systems and Software Engineering
- Software Engineering Body of Knowledge (SWEBOK)
- ITIL Service Transition 2011: Service Asset and Configuration Management
- ISO/IEC 20000-1 Information Technology - Service Management - Part 1: Service Management System Requirements
- ISO/IEC 20000-2 Information Technology - Service Management - Part 2: Guidance on the Application of Service Management Systems
- NIST SP 800-128 Guide for Security-Focused Configuration Management

2.150.2.1.8
(04-18-2025)

Training

- (1) Process training involves training all stakeholders about key processes that are crucial for an organization to deliver business objectives. Training provides clarity to employees on a set of procedures that needs to be carried out as part of the process and the best possible way to do them. Listed below are the training resources available in the IRS Integrated Talent Management and SkillSoft Percipio online training platforms:

- Configuration Management Overview (CBT) (Course 23279)
- Change Management Process Overview (CBT) (Course 43161)
- Software Configuration Management Handbook, Third Edition
- LLPKG - Overview of the ITIL Service Lifecycle
- ITIL(r) 4 Foundation: Introduction
- ITIL(r) 4 Foundation: Service Management Practices (Part 1)
- ITIL(r) 4 Foundation: Service Management Practices (Part 2)
- ITIL Service Desk, IT Asset, Service Configuration, and Change Control Management

Note: Technical training for Configuration Management is also available in the IRS Integrated Talent Management and SkillSoft by querying the phrase “Configuration Management”.

2.150.2.2
(08-19-2020)

Configuration Management Process

- (1) The following sections below describes the overall Configuration Management process that includes its process flow, inputs, outputs, activities, and guidelines.

2.150.2.2.1
(08-19-2020)

Main Process Diagram

- (1) The Main Process Diagram illustrates the key process activities and process interfaces for Configuration Management.

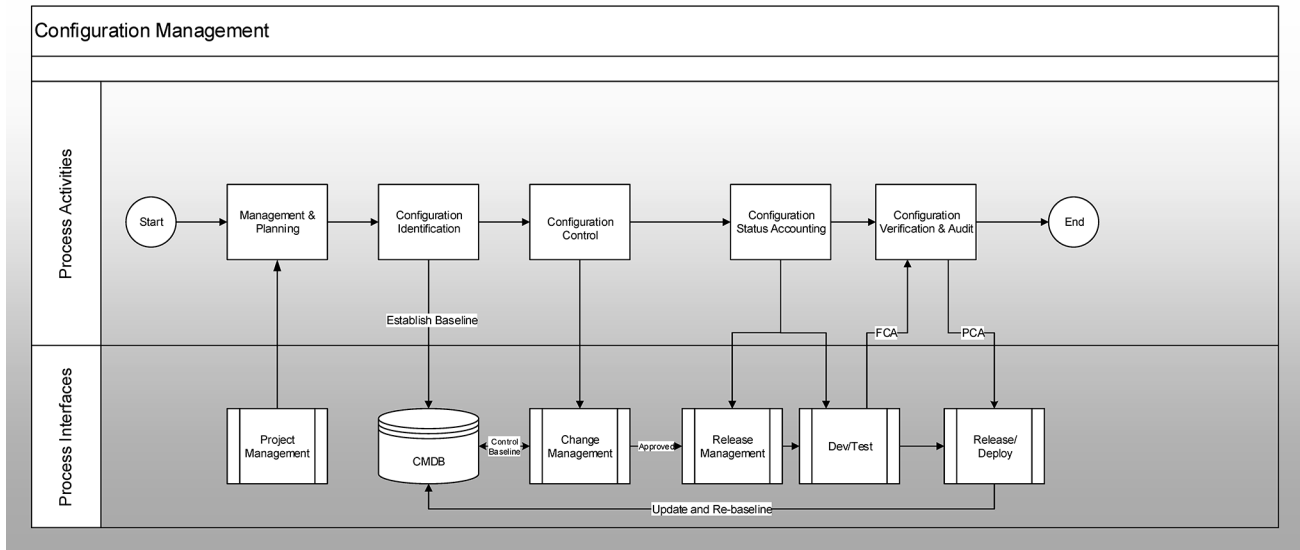


Figure 2.150.2-2

2.150.2.2.2 (04-18-2025) Inputs

- (1) Process inputs are used as triggers to initiate the process and to produce the desired outputs. Users, stakeholders or other processes provide inputs. The following is a list of inputs for this process:

| Name | Description | Supplier |
|--|--|---|
| Change Request (CR) or Work Request (WR) | A CR or WR is the record of a change proposal for changes to a configuration item baseline and that is worked through the Change Management or Unified Work Request process. | Change Analyst, Requestor |
| IT Asset | Tangible and physical IT components that the organization owns, such as computers, servers, network, applications, software, including digital data. | Asset Management, Authoritative Data Source |

2.150.2.2.3 (04-18-2025) Outputs

- (1) Each process produces tangible outputs. These outputs can take the form of products or data and can be delivered to a user or stakeholder or they can be used as inputs to other processes. Outputs are measurable in terms of quantity and quality.

| Name | Description | Recipient |
|------|-------------|-----------|
|------|-------------|-----------|

| | | |
|------------------------------------|---|--|
| Audit Results | <p>Findings from the Functional Configuration Audit (FCA) that is used to verify the actual performance of the configuration item meets the requirements stated in its performance specification; and for systems, the actual performance of the system meets the requirements stated in the system performance specification.</p> <p>Findings from the Physical Configuration Audit (PCA) that is used to examine the actual configuration of the configuration item that is representative of the product configuration to verify that the related design documentation matches the design of the deliverable's configuration item. This includes validation of many of the supporting processes that are used in the production of the configuration item as well as any elements of the configuration item that were redesigned after the completion of the FCA also meet the requirements of the configuration item's performance specification.</p> <p>Findings summarizing the results from a CMDB audit revealing the differences between the CMDB records and the installed configuration items.</p> | <p>Project Manager, Project Stakeholder, Software Quality Assurance, Software Developer, Software Tester</p> <p>CI Owner, CM Process Manager</p> |
| Configuration Baseline | A baseline for a configuration item that has been formally reviewed and agreed on, that thereafter serves as the basis for further development or delivery, and that can be changed only through change control procedures. | CI Owner, Configuration Manager, Project Manager, CCB, and TRB. |
| CI Information | Information on a configuration item or multiple configuration items. | CI Owner and Configuration Manager |
| CI Structure Chart | A required process artifact under the OneSDLC that documents and describes the relationships between and the affected configuration items for a project. | Configuration Manager, Project Manager, and CM Process Manager |
| Configuration Item | A collection and combination of hardware, software, and documentation that is used to deliver a product or service. | Project Manager, Configuration Manager, CI Owner |
| Configuration Management Plan | A formal document and plan that describes the roles and responsibilities, resources, tools, change authority, and the appropriate process to manage the configuration item throughout its lifecycle. | Configuration Manager, Project Manager, and CM Process Manager |
| Configuration Management Worksheet | A configuration item register that list all related documentation, hardware, and software that is required to develop, test, deliver, monitor, and control the configuration items for the project. | Configuration Manager and Project Manager |

| | | |
|---------|---|--|
| Reports | Reports on information contained about the status of a configuration item. For example, changes, baseline information, etc. | Project Manager, Configuration Manager, Software Quality Assurance, CI Owner |
|---------|---|--|

2.150.2.2.4
(04-18-2025)
Activities

- (1) An activity is a major unit of work to be completed in achieving the objectives of the process. A process consists of a sequence of related activities that transforms inputs into outputs and performed by the roles defined in the process. Activities are measurable in terms of efficiency and effectiveness. Configuration Management consist of the following 5 processes:
 - a. **Management & Planning.** This activity establishes the CM Plan that describes the appropriate Configuration Management process that will be performed during the product lifecycle. It details the Configuration Management activities, the assigned roles and responsibilities, change authority, tools, and appropriate process to manage the configuration items within scope of the CM Plan.
 - b. **Configuration Identification.** This activity identifies the configuration items that will be controlled, establishes standard naming convention and schemes, version control, structural relationships between configuration items, assignment of ownership, identification and selection of key attributes, and establishes configuration baselines.
 - c. **Configuration Control.** This activity is responsible for managing changes and release implementation of changes to configuration items and baseline configuration documentation by leveraging on existing formal change and release processes. It also ensures that only authorized and identifiable configuration items are in the infrastructure and that there is a corresponding, accurate, and complete configuration item record representing its actual or physical configuration item.
 - d. **Configuration Status Accounting.** This activity establishes the process for recording, managing, and reporting on the status of approved configuration documentation and information, including traceability of all changes.
 - e. **Configuration Verification & Audit.** This activity establishes the integrity in the configuration documentation and information used as the basis for configuration control and support of the product throughout the software development lifecycle by maintaining the integrity of the configuration baselines. It confirms that the resulting configuration baselines and documentation conform to the standards or requirements, and also confirms the integrity of a system or product prior to delivery.

2.150.2.2.5
(08-19-2020)
Guidelines for Configuration Management

- (1) In accordance with IRM 2.150.1 *Configuration Management Policy*, IT organizations must follow the Configuration Management process and implement it for their programs, projects, and systems throughout the lifecycle. Configuration Management includes identifying and baselining configuration items, controlling and recording changes, reporting on the status and implementation of the changes, and conducting review and evaluation to ensure that the build conforms to the requirements and all items identified as part of the configuration are present in the product baseline. The following sections describes further details for each of the Configuration Management process.

2.150.2.2.5.1
(04-18-2025)

Management & Planning

- (1) IT organizations, programs, and projects must develop a plan and strategy to define the scope, objectives, resources, and appropriate Configuration Management process in a CM Plan. This also applies to any organizations providing IT services to the IRS, such as IRS Business and Functional Units and Managed Service Providers. Once the CM Plan is approved through the OneSDLC Readiness Exit Review, it is valid for 3-years and can be used for subsequent releases.
- (2) IT organizations may establish an organizational or Associate Chief Information Office (ACIO)-level CM Plan that defines their current resources, including CCBs, and tailored Configuration Management process to manage their existing configuration items. IT Projects that are owned by their ACIO may leverage on their ACIO-CM Plan for their product configuration items and used for their OneSDLC Readiness Exit Review by using the Addendum to the ACIO CM Plan (Addendum) to document the agreement and any deviations against their ACIO CM Plan. If there are too many deviations against the ACIO CM Plan, it is recommended that the project develops a Project CM Plan. The ACIO CM Plan is valid for 3-years and must be revised or re-certified and approved by the IT CM Process Owner to maintain currency and to be used by their IT Projects to leverage from. Exclusions to using the ACIO CM Plan are:
 - Non-IT organization. IRS Business Units and Functional Units must develop a Project CM Plan.
 - Managed Service. Systems owned and managed by the Managed Service Provider must develop a Project CM Plan or functional equivalent. A functional equivalent must consist of the four CM major activities that defines how the configuration items will be managed and controlled. Functional equivalent must be reviewed and approved by IT CM PMO for OneSDLC Readiness Exit Review.
 - Program Management Office (PMO). PMOs developing a CM Program Plan or equivalent must develop a Project CM Plan. A CM Program Plan does not define the specific scope, schedule, and configuration items that will be delivered for each project associated with each OneSDLC Readiness Exit Review.
- (3) Planning for Configuration Management is essential to the success of managing and delivering the product or IT solution into deployment through sustaining operations since Configuration Management extends throughout the system's life cycle. Planning involves the "what" and "how" activities are to be performed and performing these activities according to the CM Plan. The most significant activity is identifying the appropriate Configuration Management process to manage the configuration items based on the software methodology used. In configuration planning, the following items must be considered into the CM Plan:
 - a. Scope and Objective. The project scope and objective must be clearly defined.
 - b. Roles and Responsibilities. The appropriate Configuration Management roles and description of its responsibilities, including other related process roles must be defined and assigned.
 - c. Configuration Management Tools. Configuration Management and related tools must be identified that will be used to support Configuration Management and related activities.
 - d. Configuration Management Training. Personnel assigned to support Configuration Management must be trained in the process and/or tools.

- e. Contractor/Vendor Control. IT projects that acquire or make use of commercial tools and/or services, must define type of service provided to IRS, where the service is hosted, interfaces with IRS systems, and level of customization.
 - f. Software Development Life Cycle (SDLC) Model. The type of SDLC approach (e.g., COTS, GOTS, Custom COTS, Managed Service, or Planned Maintenance) that will be used must be defined to establish the appropriate Configuration Management process to develop or change the configuration item.
- (4) A standard CM Plan Data Item Description (DID), including the Addendum DID has been established by the IT CM PMO. Organizations, programs, and projects may differ in scope and complexity and a single format may not always be applicable. The standard CM Plan DID provides the minimum format for plans and maximum amount for flexibility. If the section for the format is not applicable, the sentence "Not Applicable" for this section must be inserted to indicate that this section has not been overlooked.

Note: *ACIO CM Plan updates and recertifications must be submitted to the IT CM PMO for review and approval by the IT CM Process Owner.*

Note: *In reference to Secure Configuration Management, IRM 10.8.1.4.5.8 CM-09 Configuration Management Plan is what establishes the requirement to document, define configuration item, and develop a CM Plan.*

2.150.2.2.5.2
(04-18-2025)
**Configuration
Identification**

- (1) IT organizations, systems, and projects must identify candidate software configuration items that will be placed under configuration control and management. This involves understanding the software configuration within the context of system configuration and selecting software configuration items. Software configuration is the functional and physical characteristics of the software defined in the technical documentation or achieved in the product. The following are examples of software items that are candidate software configuration items:
- plans
 - specifications and design documents
 - testing materials
 - software tools
 - source and executable codes
 - code libraries
 - data and data dictionaries
 - documentation for installation, maintenance, operations and software use
- (2) Identify configuration items. Configuration Identification is responsible for the planning and preparation for identifying configuration items. This includes modeling the IT infrastructure to determine what configuration items will look like and how they are related to each other. It also includes the development of a taxonomy, standard naming conventions, configuration item selection and classification, and assignment of CI Ownership for defining configuration item attributes. The following items below describe the steps for identifying configuration items.

- a. **Build Logical Model.** IT infrastructure components must have a representation in the models and standard methods for modeling configuration items and their relationships must be defined and applied consistently across the IT organization. Standard methods for modeling services, applications, and components must be established and then creating conceptual logical model of the IT infrastructure. These standard methods and conceptual models will become requirements for modeling the services, application, and IT infrastructure in the system architecture and CMDB. The logical model defines the scope of the Configuration Management activities, that is the level of detail that the organization would want to trace the relationships. Initially beginning with a high-level approach by defining several key services and tracing them back to the application to servers and other service components.
- b. **Establish Taxonomy and Standard Naming Convention.** Services and applications can be referred to by multiple names. Standardization of terms for service and application components is required to consistently model the service and all of its related components. Standard terms and definitions must be defined and published. This includes establishing a standard naming convention with required data elements as part of the control. Accordingly, using the logical model, define what parts of an application will be designated as a service, what parts will be designated as an application, and what are the parts that are running on the platform.
- c. **Configuration Identification Index or Configuration Identification ID (CII).** Establish a standard CII methodology to assign a unique identifier for each configuration item so that it can be distinguished from all other configuration items and their associated product configuration information. See IRM 2.150.2.3.1 *Configuration Identification Index (CII) Guidelines* including IRM 2.150.2.3.2 *Document Versioning Guidelines* for configuration documentation. For IT infrastructure configuration items using automated tools such as a CMDB, a unique identifier assignment is usually part of its automated functionality to automatically assigned CII.
- d. **Define CI Types and CI Subtypes.** Establishing configuration item classification based on CI Type and CI Subtypes enables management and organization of configuration items. Define CI Types based on the logical model for IT infrastructure components. This entails categorizing any and all configuration items that will be managed (e.g., services, applications, servers, database). This will also establish guidelines for the selection of configuration items and define a set of CI Types and its CI Subtypes. The selection of configuration items and the level to which they are defined are very important parameters in the design of the CMDB. That is, if the level is defined too granular, the CMDB can become cumbersome, bloated and difficult to manage. If the level is too high, the CMDB may not meet the operational requirements of the supported processes. An example of a CI Type and CI Subtype is listed in Figure 2.150.2-1. For configuration documentation, see IRM 2.150.2.4 *Configuration Documentation Classification Guide*. The table below provides a general example of a CI Type and CI Subtype.

| CI Type | CI Subtype |
|---------|------------|
|---------|------------|

| | |
|----------------------------|--|
| Data | <ul style="list-style-type: none"> • metadata • data attribute • data relationship • database • database schema |
| Documentation | <ul style="list-style-type: none"> • plan • specification • design document • requirement • testing material • source code • data dictionary • documentation for installation, maintenance, operations, and software use |
| Facility | <ul style="list-style-type: none"> • data center • computer or server room |
| File | <ul style="list-style-type: none"> • configuration file • executable code or executable file |
| Computer | <ul style="list-style-type: none"> • mainframe • server • appliance |
| Network Device and Related | <ul style="list-style-type: none"> • router • switch • load balancer • IP address |
| Process | <ul style="list-style-type: none"> • business process • IT process |
| Service | <ul style="list-style-type: none"> • business service • application service • technical service |
| Software and Application | <ul style="list-style-type: none"> • business application • commercial-off-the-shelf • system software • programming software • firmware and driver • middleware • database management system |

- e. **Assign CI Owner.** Identify and assign an owner for each CI Type including CI Subtypes. The CI Owner will be responsible for identifying the information about the CI Type, such as what attributes are needed and valid values and/or parameters for each attribute, and the source for gathering the information. CI Ownership may be the business owner and/or technical owner, specifically the group or individual that will be responsible for approving or making the change. CI Ownership may be assigned at the organization, group, or individual level. For organization or group, such as using an organizational symbol or assignment group, it is recommended to also identify an individual point of contact.
- f. **Define Configuration Item Attributes.** The CI Owner defines and selects the attributes for each CI Type that they own. An attribute is a piece of

information about the configuration item such as name, location, version number, etc. This includes any valid values and/or parameters for each attribute. Configuration item attributes are used to identify the characteristics of the configuration item and also used for controls and baselines. Attributes may be discoverable or non-discoverable. Discoverable attributes are configuration item attributes gathered through auto discovery tools in hardware assets and added in the CMDB. Non-discoverable attributes are configuration item attributes that are manually defined, updated, and collected from sources used and added in the CMDB.

- g. **Identify Sources of Information for Configuration Items.** Identify the sources where the information can be found for each configuration item, such as application, software products, and hardware repositories. Auto discovery tools can also be used to identify and gather the information and related components on the network. For configuration documentation, the source may be in a specific repository such as SharePoint.
- h. **Establish Relationships or Interfaces.** Relationships or interfaces between the service (Business Service and Application Service) and IT infrastructure configuration items must be determined to identify impact of a change or incident against other configuration items. Relationships, and the ability to map those relationships to configuration items, are the defining characteristic of Configuration Management.

Documenting relationships may be done by any of the following:

- short description between configuration items
- table
- block diagram
- engineering drawing

For an automated CMDB, mapping of relationships between configuration item components may be done manually or automatically. Relationships are what convert an inventory of the IT infrastructure in the CMDB that can be used for creating service maps, determining impact, analyzing the risk of a change, and building a model of the IT infrastructure.

Note: *The IT CM PMO provides a CI Structure (CIS) Chart as a resource to manually document configuration item relationships.*

- i. **Register Configuration Items or Populate the CMDB.** Identify and select configuration items that will be managed and controlled. Configuration item records are what is managed by Configuration Management in order to manage its actual or physical configuration item. Configuration item records may be registered manually or through an automated tool, such as a CMDB. For a CMDB, begin populating the CMDB with CI Types and associated attributes. Use incremental approach to build out one CI Type at a time and verify with the CI Owner before proceeding to the next. For example, identify the application associated with each server and then identify all the attributes and owners associated with each one, initiate mapping and populate the CMDB with the CI Type and configuration details, and verify the relationship mapping between the application, hardware, and other components is correct with the CI Owner.

Note: *The IT CM PMO provides a CM Worksheet template as a resource to manually register configuration items. It is the organization, program, and*

project' responsibility to maintain their configuration items and system inventory to manage, control, and report on their documentation, hardware, and software configuration items.

- (3) Baseline Configuration Items. Software baseline is a formally approved version of software configuration items that is formally agreed upon and establishes the basis for a formal change process for future changes. Baselines must be established as the software configuration item progresses through the software development lifecycle. There are 3 types of software or system baselines:
- a. **Functional baseline** defines the functional requirements of the system or system specifications (system level architecture and design) and its interface characteristics containing the system's capability, functionality, and overall performance. It is established upon completion of system requirements review under the Domain Architecture Phase. This consist of approved configuration documentation describing a system or top level configuration item's performance, such as the Vision, Scope and Architecture.
 - b. **Allocated baseline** defines the configuration items that compose the system and how it is distributed or allocated across lower-level configuration items. It is established upon completion of software requirements and software interface requirements review under the Logical and Physical Design Phase. This consist of approved configuration documentation describing the functional and interface characteristics allocated from the system for each configuration item, such as Interface Control Document and Simplified Design Specification Report.
 - c. **Product baseline** defines the release contents or configuration items of the project for production. It is established upon completion of all technical documentation and software for product testing and acceptance that culminates in the Configuration Audits under the Systems Development Phase. This consist of the source code and approved technical documentation describing the configuration of a configuration item for production and operational support, such as Computer Operator Handbook.

The table below list examples of Configuration Management baselines for traditional software development projects.

| Functional | Allocated | Product |
|---|---|--|
| Business System Report (BS) Business Impact Analysis (BIA) Cloud Design and Requirements Package (CDRP) Configuration Management Plan (CMP) Engineering Plan Government Equipment List (GEL) Test Strategy & Plan Vision, State and Architecture (VSA) | CI Structure (CIS) Chart Configuration Management Worksheet (CMW) Interface Control Document (ICD) Simplified Design Specification Report (SDSR) | Computer Operator's Handbook (COH) Source Code User Documents and Training Materials |

Note: Source is from the OneSDLC Compliance List and Technology Insertion Artifact List.

The table below list examples of Configuration Management baselines for Agile software development projects.

| Functional | Product |
|---|---|
| Business System Report (BS) Business Impact Analysis (BIA) Cloud Design and Requirements Package (CDRP) Configuration Management Plan (CMP) Engineering Plan Government Equipment List (GEL) Test Strategy & Plan Vision, State and Architecture (VSA) | CI Structure (CIS) Chart Configuration Management Worksheet (CMW) Interface Control Document (ICD) Simplified Design Specification Report (SDSR) Source Code User Documents and Training Materials |

Note: Source is from the OneSDLC Compliance List and Technology Insertion Artifact List.

For software that is acquired, its origin and initial integrity must be established. Following the acquisition of a software, changes to the item must be formally approved and baseline according to the appropriate procedure.

Note: In reference to Secure Configuration Management, IRM 10.8.1.4.5.1 CM-02 Baseline Configuration requirements for baseline configurations for systems and system components include connectivity, operational, and communications aspects of systems. Baseline configurations are documented, formally reviewed, and agreed-upon specifications for systems or configuration items within those systems. Baseline configurations serve as a basis for future builds, releases, or changes to systems and include security and privacy control implementations, operational procedures, information about system components, network topology, and logical placement of components in the system architecture. Maintaining baseline configurations requires creating new baselines as organizational systems change over time. Baseline configurations of systems reflect the current enterprise architecture.

- (4) Establish Software Libraries. A software library is a controlled collection of software and related documentation for software development. It is also used for software release management and deployment activities. Software libraries must be established for source code management and control, development, and testing. Tools used to support the software libraries must support Configuration Management controls such as version and access control. Software libraries must also be established for configuration item related documentation to maintain current baselines and previous versions. These should include plans, designs, requirements, and any technical and operational documentation that support the software.

2.150.2.2.5.3
(04-18-2025)

Configuration Control

- (1) IT organizations, programs, and projects must implement a controlled change process and provide tailored methods and standard operating procedures for effectively planning, recording, controlling, and validating product requirements and data that contain the requirements. Tailoring will depend on the organization and the level of control or complexity needed. The change process is the cornerstone activity for Configuration Management including tracking changes to ensure that the configuration of the product is accurately known at any given time. This is accomplished by identifying each baseline and tracking all subsequent changes made to that baseline. Accordingly, Configuration Control is about managing and controlling the configuration item baseline. All changes must be associated with the configuration item, documented, and submitted through a change request. The authority for making decisions on proposed changes is under the CCB. For smaller projects, the authority is delegated to a lower-level change authority, SCCB, based on specific criteria defined for its scope and control. Formal change procedures are defined in IRM 2.22.1 *Unified Work Request (UWR) Process* for business application and IRM 2.125.2 *Change Management Process* for infrastructure.
- (2) Approved change requests are implemented using defined software procedures based on type of software development methodology. Additionally, since there are a number of approved change requests that might be implemented simultaneously, it is critical to track which change request are incorporated for each software versions and baselines. Formal system development procedures are defined in IRM 2.5 *Systems Development*.

Note: *In reference to Secure Configuration Management, IRM 10.8.1.4.17.9 SA-10 Developer Configuration Management sets the requirements for developers of systems, system components, system service etc., to document, manage, and control integrity change for configuration item under Configuration Management.*

- (3) Approved change requests are also received by software quality assurance to plan, prepare, and implement test activities, such as developing test plans, test cases and test scripts and documenting and reporting on the test results to Software Developers for correcting software defects. Individual software applications are integrated with other software products for the final system integration testing. Successful completion of integration testing prepares the product for software release management. Formal testing procedures are defined in IRM 2.127.2 *IT Testing Process and Procedures*.
- (4) Configuration records are updated throughout the software development process and prepared for Configuration Status Accounting & Reporting.
- (5) Software release management is the identification, packaging, and delivery of all software configuration items, such as executable program, documentation (e.g., Version Description Documentation), release notes, and configuration data. This includes delivery of the installation instructions and other data concerning the use of the new system, and defining the environment on which the software will run.
- (6) As part of the change process closure, completed changes may undergo configuration audits and/or software quality assurance to ensure that only approved changes have been made.
- (7) Final updates to configuration records are made and a new baseline is established once the system is deployed into production.

Note: In reference to Secure Configuration Management, IRM 10.8.1.4.5.2 CM-03 Configuration Change Control requirements includes changes to baseline configurations, configuration items of systems, operational procedures, configuration settings for system components, remediate vulnerabilities, and unscheduled or unauthorized changes. For changes that impact privacy risk, the senior agency official for privacy updates privacy impact assessments and system of records notices. For new systems or major upgrades, organizations consider including representatives from the development organizations on the Configuration Control Boards or Change Advisory Boards. Monitoring of changes includes activities before and after changes are made to systems and the auditing activities required to implement such changes.

- (8) Changes to configuration item records in the CMDB must be associated with a change request and submitted by the CI Owner to change the baseline and maintain process compliance. For incorrect configuration item records, an incident must be reported to correct the record and restore the configuration item baseline.

2.150.2.2.5.4
(08-19-2020)
**Configuration Status
Accounting**

- (1) IT organizations, programs, and projects must identify, record, and maintain software configuration status information for software configuration items that were approved and baseline and report its activities for management, software engineering, and other related needs. This includes establishing measurements and tracking change request status, deviations, and waivers as it progresses through the software development process. For example:
- Record and report on all approved configuration documentation including changes
 - Record and report the status of all change requests associated with each configuration item
 - Record and report on the history of change approvals
 - Record and report the status on the verification and validation activities
- (2) Identify the type of report(s) (formal or informal), frequency of reporting, and the audience to report on the configuration status information. For example:
- What kind of report to produce for each software configuration item
 - What data to report and source of the data
 - What is the frequency for reporting on the status for each software configuration item
 - Who are the stakeholders that will receive the reports

2.150.2.2.5.5
(04-18-2025)
**Configuration
Verification & Audit**

- (1) IT organizations, programs, and projects, at discretion, may perform Configuration Audits to ensure that Configuration Management processes are followed, configuration items are built based on requirements, and the integrity of the configuration baselines are maintained. Specifically, Configuration Audits ensure that (1) baselines are complete, correct and consistent in relation to functional and physical specifications; (2) approved changes were correctly implemented and verified; (3) no unauthorized changes have occurred; and (4) software products are ready for release. Therefore Configuration Audit increases software visibility and establishes traceability of changes throughout the development lifecycle and reveals whether the product requirements are being satisfied and whether the preceding baseline has been fulfilled. The audits enable project management to evaluate the integrity of the software

product being developed, resolve issues that may have been raised by the audit, and correct defects in the development process. Configuration Audits are conducted prior to software deployment to ensure that the software product has been built and tested according to specified requirements and all items that are designated as part of the configuration are installed as defined by its requirements.

- (2) There are two formal types of Configuration Audits that are performed prior to release of the software to production: Functional Configuration Audit (FCA) and Physical Configuration Audit (PCA). FCA ensures that each item of the software product has been tested to determine that it satisfies the functions defined in the specifications. PCA determines whether all items identified as being part of the configuration are present in the product baseline. The FCA/PCA are formal audit processes in the traditional software development methodology. FCA/PCA Audits are normally performed by Software Quality Assurance. Further details of the FCA/PCA are described below:
 - a. **Functional Configuration Audit.** FCA is conducted once the software product has been developed and tested. The audit intends to confirm that the software product is verified and tested relative to its allocated requirements in relation to its high-level requirements. FCA ensures that the functional and performance attributes of a configuration item (baseline components) are achieved (done right thing). The FCA should occur at least once for new development but may be held more frequently as determined by the Project/Product Manager.
 - b. **Physical Configuration Audit.** PCA is conducted once the software product has been fully completed and all corrective actions have been closed. PCA ensures that each configuration item, as-built, conforms to the technical documentation that defines it, that is: (1) all items identified as being part of the configuration are present in the product baseline; (2) the correct version and revision of each part are included in the product baseline; and (3) each item corresponds to information contain in the baseline's configuration status report. In other words, the configuration item (baseline components) is installed as defined by the requirements in its detailed design documentation (done thing right). PCA should be accomplished on the product baseline for each release to verify its authenticity before the full deployment decision to verify compliance with the stated requirements and to ensure that all life-cycle documentation supports the added functionality.

Note: FCA/PCA activities are defined through a checklist. Since each product have different scope and complexity, there is no standard template for the FCA/PCA checklist. However, standard requirements described above, specifically identification of configuration baseline items and status of changes to configuration baseline items, should define FCA/PCA requirements to establish the checklist.

- (3) Configuration Audits for Agile are informal and may be performed at the end of each sprint. Unlike the traditional method (FCA/PCA), the approach is flexible, iterative, and focuses on continuous communication and collaboration between the audit team and stakeholders, and completed in short time frames. An audit backlog is created that defines the scope of the items for the audit which are reviewed and prioritized based on risk and value, and updated based on the needs of the organization, such as emerging issues experienced by stakeholders. The audit team collaborates with the stakeholders on how an item in the

backlog will be tested and examined, including the expected value from the test and audit requirements. The audit is conducted and completed within sprint intervals by dividing the audit into small increments and grouping user stories into sprints. At Sprint Review Meetings, the backlog items are reviewed, and the completed work is demonstrated to the Product Owner to determine whether the items satisfy the requirements. A Point of View report is developed that summarizes all relevant insights gained from the observations and stories.

- (4) Configuration Verification & Audit for the CMDB must be performed to ensure data quality and integrity. Configuration Verification activities are performed by CI Owners and supported by various CMDB roles that is responsible for managing configuration item records in the CMDB. The following Configuration Verification activities are:
 - a. **Data Integrity Verification.** Review for (1) unregistered components and (2) missing components (previously registered components).
 - b. **Data Quality Verification.** Review for (1) completeness (i.e., required fields and recommended fields are complete); (2) correctness (i.e., duplicate configuration items, orphan configuration item relationships, and staleness/old configuration item(s); and (3) compliance (i.e., desired state based on standard configuration for each configuration item).

A verification report is generated to document the overall status and for any discrepancies and corrective actions required to reconcile the discrepancies. Changes to configuration item and its record must be submitted through Change Management to manage and control the baseline. Corrections to configuration item and its record must be submitted through Incident Management to restore the baseline.

Configuration Audit on the CMDB must also be performed by a Configuration Auditor that supports the CM Process Manager. The objective is to evaluate whether the Configuration Verification process is being performed including measuring the accuracy and quality of data in the CMDB. Configuration Audits may also trigger corrective actions required to reconcile the discrepancies and distributed to CI Owners for resolved.

2.150.2.3 (04-18-2025) Configuration Management in the Cloud

- (1) Configuration Management in the cloud environment (off-premises) is still a requirement as it is in the IRS environment (on-premises). The differences are the configuration items and roles and responsibilities between the IRS and the Cloud Service Provider (CSP). In the cloud environment, the following cloud computing models are:
 - a. **Infrastructure-as-a-Service (IaaS).** Computing resources, such as servers, storage, and networking capabilities over the internet are provided to the customer allowing its users to deploy and manage their applications without ownership and maintenance of the hardware.
 - b. **Software-as-a-Service (SaaS).** Software applications are delivered over the internet on a subscription basis allowing the customer and its users to access and use the software through a web browser without ownership and maintenance of the software on cloud servers.
 - c. **Platform-as-a-Service (PaaS).** A complete platform with hardware, software, and development tools are delivered over the internet, allowing the customer and its users to build, deploy, and manage applications without ownership of the entire IT infrastructure (hardware and software).

- (2) In a cloud environment, the CSP provides the cloud-based resources (IT infrastructure and technical services) and the Managed Service Provider (MSP) manages and supports the customer's cloud services from the CSP. Since these IT infrastructure assets are owned and managed by the CSP, the customer needs to identify the configuration items in the cloud that they will own and/or manage to comply with Configuration Management. Accordingly, the customer is responsible for their data, endpoints, account, and access management. The CSP is responsible for its cloud infrastructure ensuring its reliability, availability, and security. Examples of customer configuration items in different cloud services (e.g., Amazon Web Services, Microsoft Azure, Google Cloud Platform) can include the following:

- instance type selection
- operating system choice
- security policies
- network topology and settings
- storage size and type
- database schema design
- application configurations
- load balancing configurations
- custom domain names
- access control policies

The CSP provides specific configuration settings or attributes that the customer can customize to tailor the service to their needs. The following are key points to these customer configuration items:

- a. Customization. Select attributes that allows the customer to configure the cloud service based on customer requirements.
- b. Managed in the Cloud Console. Allows the customer to access and modify the configuration item attributes through the CSP console to configure define/update the set values and parameters.
- c. Tracking and Auditing. Changes to customer configuration items are tracked and audited by the MSP.

- (3) The following are the customer role in cloud Configuration Management:

- Defining desired configurations. Specifying the settings and parameters for their cloud instances, applications, databases, and other services.
- Implementing security configurations. Setting appropriate security policies, user access controls, and encryption mechanisms to protect their data within the cloud.
- Managing access controls. Determining access rights to different cloud resources including permission levels.
- Using configuration management tools. Employing cloud provider-specific tools or third party Configuration Management platforms to automate and manage configuration across their cloud environment.
- Monitoring and auditing configurations. Reviewing and auditing their cloud configurations to identify security vulnerabilities and compliance issues.

2.150.2.4
(08-19-2020)
**Configuration
Identification Index (CII)
and Document
Versioning Guidelines**

- (1) The following sections below provides the guidance for assigning Configuration Identification Index and Document Versioning for documentation configuration items.

2.150.2.4.1
(04-18-2025)
CII Guidelines

- (1) The CII guidance applies to manually record and register configuration items, usually for configuration documentation. The CII is composed of 5 fields: Config-, ID-, Index-, Ver-, and Date. The fields are separated by a hyphen and when joined, make up the CII. For example, the figure below illustrates how the CII is decomposed for **IT:EO:DMPG-PLN-IT_CM_Plan-V1.0-01012025**.

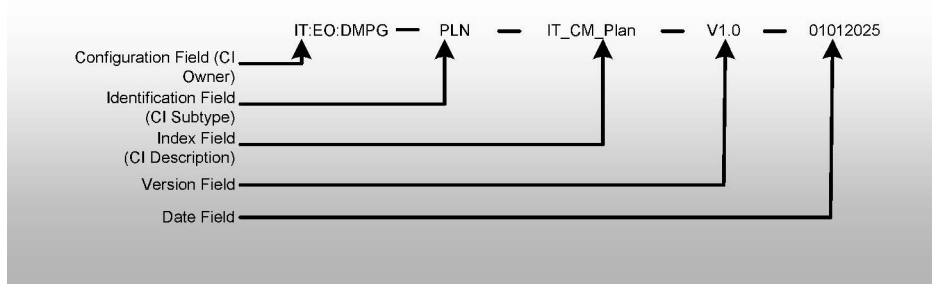


Figure 2.150.2-3

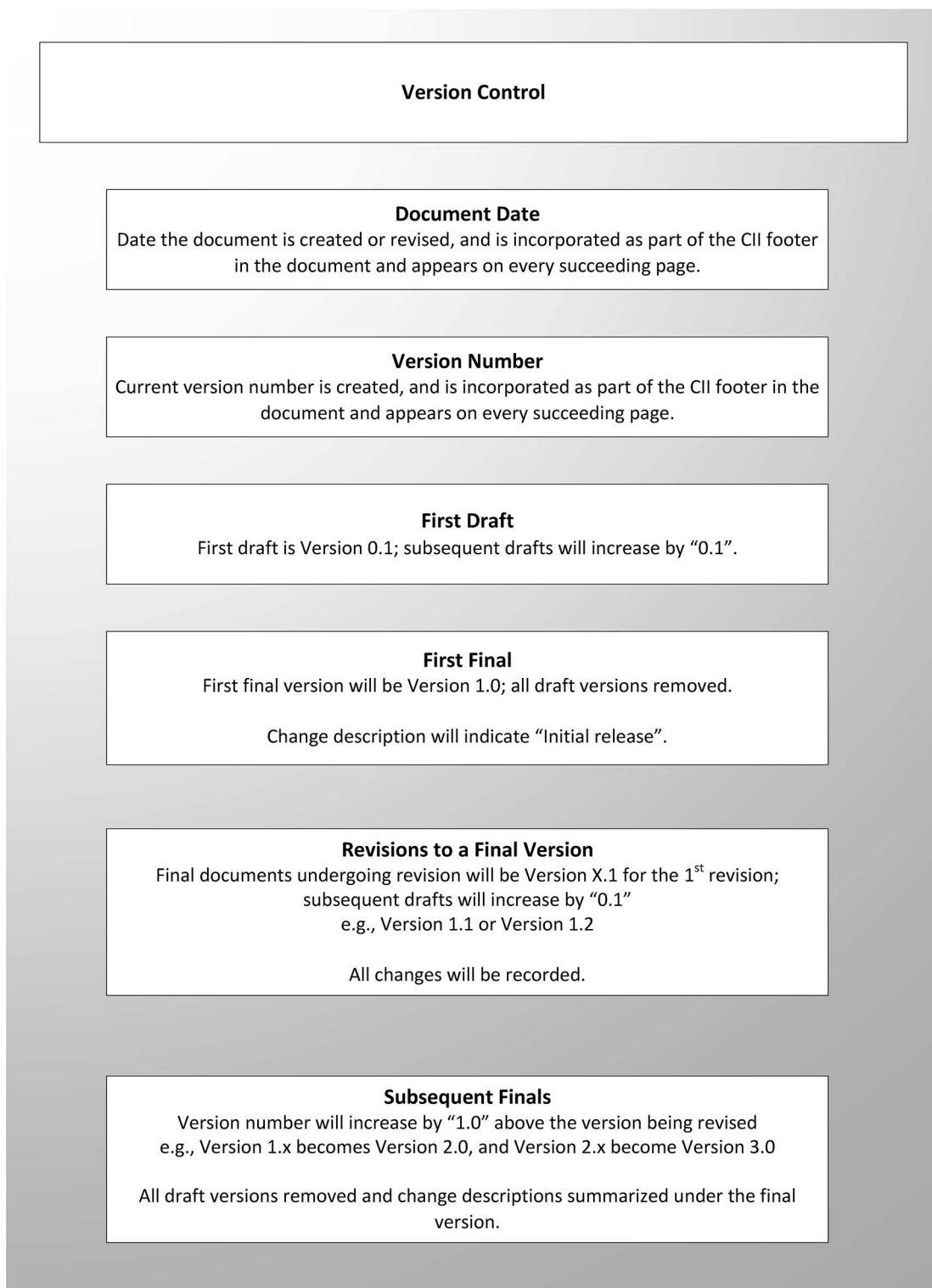
1. The **“Configuration (Config)”** field is defined as the CI Owner and will contain the IRS organizational symbols or external Service Provider as input. A CI Owner is the one that owns and is responsible for the specific configuration item and the primary validation point for information related to that configuration item. The field size is limited to 15-characters in length. If the organizational symbol exceeds 15-characters, it will need to be **truncated starting from the left** to meet the 15-character limit. For example, for OS:IT:EO:DMPG:GRMB would be truncated to **IT:EO:DMPG- or EO:DMPG:GRMB**.
2. The **“Identification (ID)”** field is defined as the CI Classification. The CI Classification contains the CI Type and specific class of a configuration item (or CI Subtypes). For configuration documentation, the various CI Subtypes may be requirements, designs, or plans. This field is limited to 4-characters in length. For example, IT:EO-DMPG-**PLN-**. For the CI Classification, see IRM 2.150.2.4 *Configuration Document Classification Guide* for reference.
3. The **“Index”** field is defined as the configuration item reference containing the abbreviated title of the configuration item. This field is limited to 15-characters in length. For example, IT:EO:DMPG-PLN-**IT_CM_Plan-**.
4. The **“Version (Ver)”** field is defined as the version of the configuration item. For configuration documentation, IRM 2.150.2.3.2 *Document Versioning Guidelines*, provides the instructions for versioning guidelines. In addition, when finalizing the document version, all draft versions listed in the “Change History” or “Record of Changes” table will need to be removed and its nature of changes summarized under the final version. The version must start with a “V” in the CII, such as V1.0, V2.0, etc. This field is limited to 4-characters in length. For example, IT:EO:DMPG-PLN-IT_CM_Plan-**V1.0-**.

5. The “**Date**” field is defined as the date of creation or update of the configuration item and must be in “MMDDYYYY” format. For documentation configuration items, IRM 2.150.2.3.2 *Document Versioning Guidelines*, also provides the instructions for dates as part of versioning guidelines. The date must be 8-characters long and must be in MMDDYYYY format. For example, IT:EO:DMPG-PLN-IT_CM_Plan-V1.0-**01012025**.
- (2) For documentation configuration items, the CII should be inserted as a footer in the document. Any changes to the document, the Ver- and Date- fields will need to be updated on the CII.

Note: The CII may also be used as a Document Identification Number.

2.150.2.4.2
(04-18-2025)
**Document Versioning
Guidelines**

- (1) Document versioning refers to the use and management of multiple versions of a document. The fundamental aspect of document versioning is tracking changes and tracking the creation of multiple document versions such as by numbering document versions in successions. Below are the document version control guidelines, and as illustrated in the figure below.

**Figure 2.150.2-4**

1. **Document Date.** Date the document is created or revised, and is incorporated as part of the CII footer in the document and appears on every succeeding page.

2. **Version Number.** Current version number is created, and is incorporated as part of the CII footer in the document and appears on every succeeding page.
3. **First Draft.** Registering the first and subsequent drafts of the document.

The following steps are listed below:

1. The first draft of a document will be V0.1.
2. Subsequent drafts will increase by "0.1" in the version number (e.g., V0.2, V0.3, V0.4).
3. Date will change with each draft version.
4. All changes will be documented in the change history table.
4. **Final Version.** Registering the first final version of the document.

The following steps are listed below:

1. Document will be deemed as final after all reviewers have provided final comments and dispositioned.
2. The first final version of a document will be V1.0.
3. Change to the latest date when the document becomes final.
4. Remove draft version(s) and only final version remain with "Initial release" language in the change history table.
5. **Revisions to a Final Version.** Registering updates to the first final version of the document.

The following steps are listed below:

1. Final documents undergoing revisions will be V0.1 for the first draft of the previous final revision.
2. While the document is under review, subsequent draft versions will increment by "0.1", that is V0.1, V0.2, V0.3, etc.
3. Date will change with each draft version.
4. All changes will be documented in the change history table.
6. **Subsequent Finals.** Registering updates to subsequent final versions of the document.

The following steps are listed below:

1. When the revised document is deemed final, the version will increase by one over the version being revised, that is draft V1.3 will become final V2.0.
2. Change to the latest date when the document becomes final.
3. Remove draft version(s) and only final version remain with summary of changes from previous draft(s) in the change history table.
4. Subsequent final documents will increment by one from the previous final version, that is from V1.0 to V2.0, or V2.0 to V3.0, etc.

Note: If the changes are minor, increment the tenths place after the decimal for the next final version, that is from V1.0 to V1.1. However, if the changes are substantial, increment by one for the next final version, that is from V1.0 to V2.0.

2.150.2.5
(04-18-2025)

(1) The following table below defines the configuration documentation CI Subtypes for IRS IT:

**Configuration
Documentation
Classification Guide**

| Types of Documents | Description | Examples |
|-------------------------------------|---|---|
| Agenda (AGN) | An outline or list of topics that will be discussed in a meeting. | Meeting Agenda |
| Agreement (AGT) | A commitment between two parties (e.g., service provider and a client, or between two organizations) describing the outlines of an understanding that two or more parties have agreed to. | Service Agreement (Master Service Level Agreement (MSLA) Service Level Agreement (SLA) and Operational Level Agreement (OLA)), Memorandum of Understanding (MOU), Memorandum of Agreement (MOA). |
| Briefing (BFG) | A summary of a situation or presentation of information. | Briefing Papers, Presentation (PRS), Kick off Meeting (KOM), Meeting (MTG) |
| Charter (CHTR) | A formal document that defines what an organization, team, or project is intended to be and accomplish. | Configuration Control Board (CCB) Charter, Technical Review Board (TRB Charter), Governance Board (GB) Charter, Advisory Board (AB) Charter, Project Charter, Project Decommissioning Charter |
| Internal Management Documents (IMD) | Official communications that designate authorities and/or provide instructions to staff for IRS officials and employees. | Internal Revenue Manual (IRM), Delegation Order (DO), Policy Statement, Interim Guidance (IG), Directive (DIR), Process or Process Description (PD), Procedure (PROC), Standard Operating Procedure (SOP) |
| Meeting Minutes (MM) | Official record of the discussions, motions, proposed or voted on, activities and decisions made during formal or informal meeting, briefing, or presentation. | Minutes, Notes |
| Planning (PLN) | A document that defines the course of actions based on the organization or project objectives in order to attain the specific goals before the work begins. | CM Plan (CMP), Addendum (ADD) to the ACIO CM Plan, Contingency Management Plan, Project Management Plan (PMP), System Deployment Plan (SDP), System Test Plan (STP), Engineering Plan, Project Decommissioning Plan (PDP), Risk Management Plan (RMP), Work Breakdown Structure (WBS), Integrated Master Plan (IMP), Integrated Master Schedule (IMS), Enterprise Organizational Readiness (EOR) Workbook |

| | | |
|--|---|--|
| Privacy & Civil Liberties Impact Assessment (PCLA) | A document that provides the analysis of how information in an identifiable form is collected, stored, protected, shared, and managed and provides a means to assure compliance with all applicable laws and regulations governing taxpayer and employee privacy. | Privacy Impact Assessment Management System (PIAMS) System Description, Qualifying Questionnaire, Major Change Determination, Share-Point PIA, Social Media PCLIA, Survey PCLIA |
| Reference (REF) | A document that provides pertinent details about a subject. | Dictionary (Data Dictionary), Schema (Database Schema) |
| Report (RPT) | A document which summarizes a finding, assessment, observation, or evaluation for a particular purpose and audience. | Lessons Learned Report (LLR) |
| Requirements and Design (REQD) | A document that defines the need and solution for development of the product. | Business Case, (BC), Concept of Operations (ConOps), Vision, State, and Architecture (VSA), Business Systems Report (BSR), REPO Standard Requirements Repository, Government Equipment List (GEL), CI Structure (CIS) Chart, CM Worksheet (CMW), Simplified Design Specification Report (SDSR), Interface Control Document (ICD), System Architecture or Diagram, Model (Service, Application, Infrastructure), Use Case (UC), User Story |
| Section 508 (508) | A document that describes the approach and test to be used to ensure that the solution being developed or implemented will be accessible to users with disabilities and demonstrates compliance via actual test results. | Accessibility Compliance Approach, Accessibility Risk Information, Applicable Provisions and Testing, 508 ELC Project Initiation Questionnaire |
| Security (SEC) | A document that defines the security plan and requirements for information systems. | Security Categorization Worksheet, Information System Contingency Plan (ISCP), System Security Plan (SSP), Digital Identify Risk Assessment (DIRA), ESAT Application Audit Checklist, ESAT Audit Control Response (ACR) Worksheet, Security Risk Assessment (SRA), SRA Mitigation Plan, Interconnection Security Agreement (ISA), Authorization Boundary Memo (ABM), Security Control Assessment (SCA) Plan, Information System Security Officer (ISSO) Concurrence, Vulnerability Scans, Source Code Security Analysis, Cyber Exit Concurrence Memo, Security Assessment Memo |

| | | |
|------------------|--|--|
| Standard (STAN) | A document that defines the requirements, specifications, guidelines or characteristics that can be used consistently to ensure that materials, products, processes, and services are fit for purpose. | ABA Application Naming Standard, EOps Server Naming Standard, Enterprise Standard Stack Configuration for Red Hat Enterprise Linux 7.6 for System Z |
| System (SYS) | A document that provides detailed technical information about development, operations, and maintenance of the product. | Software Design Document (Programmer Requirements Package (PRP), Functional Specification Package (FSP), Programmer Instruction (Computer Program Book), Source Code, Testing Document (End of Test Completion Report (EoTCR), Operations Document (Computer Operations Handbook (COH), Version Description Document (VDD), Transmittal (XMTL) |
| Templates (TMPL) | A master version of a digital document for completing a form or artifact. | Data Item Description (DID), Form (FRM) |
| Training (TRNG) | A document that defines the course or materials to establish the knowledge and skills for a particular process, function, or technology. | Training materials |
| User (USER) | A document that defines how to use a product or service to end users. | Guide (GUIDE), User Guide (UG), User Manual (UM) |