ISACA®
Trust in, and value from, information systems

COBIT® 5
AN ISACA® FRAMEWORK

CONFIGURATION MANAGEMENT

Using COBIT® 5

ISACA®
With more than 110,000 constituents in 180 countries, ISACA (*www.isaca.org*) helps business and IT leaders maximize value and manage risk related to information and technology. Founded in 1969, the nonprofit, independent ISACA is an advocate for professionals involved in information security, assurance, risk management and governance. These professionals rely on ISACA as the trusted source for information and technology knowledge, community, standards and certification. The association, which has 200 chapters worldwide, advances and validates business-critical skills and knowledge through the globally respected Certified Information Systems Auditor® (CISA®), Certified Information Security Manager® (CISM®), Certified in the Governance of Enterprise IT® (CGEIT®) and Certified in Risk and Information Systems Control™ (CRISC™) credentials. ISACA also developed and continually updates COBIT®, a business framework that helps enterprises in all industries and geographies govern and manage their information and technology.

Disclaimer
ISACA has designed and created *Configuration Management: Using COBIT® 5* (the "Work") primarily as an educational resource for security, governance and assurance professionals. ISACA makes no claim that use of any of the Work will assure a successful outcome. The Work should not be considered inclusive of all proper information, procedures and tests or exclusive of other information, procedures and tests that are reasonably directed to obtaining the same results. In determining the propriety of any specific information, procedure or test, security governance and assurance professionals should apply their own professional judgment to the specific circumstances presented by the particular systems or information technology environment.

ISACA
3701 Algonquin Road, Suite 1010
Rolling Meadows, IL 60008 USA
Phone: +1.847.253.1545
Fax: +1.847.253.1443
Email: *info@isaca.org*
Web site: *www.isaca.org*

Provide Feedback: *www.isaca.org/configuration-management*
Participate in the ISACA Knowledge Center: *www.isaca.org/knowledge-center*
Follow ISACA on Twitter: *https://twitter.com/ISACANews*
Join ISACA on LinkedIn: ISACA (Official), *http://linkd.in/ISACAOfficial*
Like ISACA on Facebook: *www.facebook.com/ISACAHQ*

ISBN 978-1-60420-346-2

Acknowledgments

ISACA Wishes to Recognize:

Development Team
Rajesh Nagendra, ITIL, PwC, SDC Bangalore
Bart Peeters, CISA, PwC, Belgium
Sven Van Hoorebeeck, CISA, PwC, Belgium

Work Group
Jose Manuel Ballester Fernandez, CISA, CISM, CGEIT, Temanova, Spain
Charles Betz, AT&T, USA
Sandeep Godbole, CISA, CISM, CGEIT, CEH, CISSP, Syntel, India
Jimmy Heschl, CISA, CISM, CGEIT, ITIL Expert, bwin.party digital entertainment plc, Austria
Larry Marks, CISA, CGEIT, CRISC, CFE, CISSP, PMP, IBM, USA

Expert Reviewers
Gerardo H. Arancibia Vidal, CISM, CRISC, NeoSecure S.A., Chile
Todd D. Atteberry, NTT Data, USA
Goutama Bachtiar, Advisor, Global Innovations and Technology Platform, Singapore
Dinesh O. Bareja, CISA, CISM, ITIL Cyber Defense Research Centre (Jharkhand Police), India
Rafael Fabius, CISA, CRISC, Republica AFAP S.A., Uruguay
Dan Haley, CISA, CRISC, CGEIT, Johnson & Johnson, USA
Tomas Hellum, LinkGRC, Denmark
Ken Hendrie, CRISC, GCIH, ITIL, PRINCE2, BAE Systems Detica, Australia
Giridhar Laveti, CISA, CRISC, CGEIT, Independent Consultant, India
Romulo Lomparte, CISA, CISM, CGEIT, CRISC, CRMA, LAQMS, Grupo Epensa, Peru
Shankar Natarajan, CISA, Pricewaterhouse Coopers, India
Yossi Joe Nadivi, CISA, CGEIT, ITIL, Strategic Business Solutions, Israel
Wil Nixon, CISA, Canada
Emilio A. Samudio D., CISA, CRISC, Banco Atlas S.A., Paraguay
Andreas Schober, CISA, CRISC, CIA, CRMA, A1 Telekom Austria AG, Austria
Mike Thompson, CISA, ISACA Sydney Chapter, Australia

ISACA Board of Directors
Tony Hayes, CGEIT, AFCHSE, CHE, FACS, FCPA, FIIA, Queensland Government, Australia, International President
Allan Boardman, CISA, CISM, CGEIT, CRISC, ACA, CA (SA), CISSP, Morgan Stanley, UK, Vice President
Juan Luis Carselle, CISA, CGEIT, CRISC, RadioShack, Mexico, Vice President
Ramses Gallego, CISM, CGEIT, CCSK, CISSP, SCPM, Six Sigma Black Belt, Dell, Spain, Vice President
Theresa Grafenstine, CISA, CGEIT, CRISC, CGAP, CGMA, CIA, CPA, US House of Representatives, USA, Vice President
Vittal Raj, CISA, CISM, CGEIT, CFE. CIA, CISSP, FCA, Kumar & Raj, India, Vice President
Jeff Spivey, CRISC, CPP, PSP, Security Risk Management Inc., USA, Vice President
Marc Vael, Ph.D., CISA, CISM, CGEIT, CRISC, CISSP, Valuendo, Belgium, Vice President
Gregory T. Grocholski, CISA, The Dow Chemical Co., USA, Past International President
Kenneth L. Vander Wal, CISA, CPA, Ernst & Young LLP (retired), USA, Past International President
Christos K. Dimitriadis, Ph.D., CISA, CISM, CRISC, INTRALOT S.A., Greece, Director
Krysten McCabe, CISA, The Home Depot, USA, Director
Jo Stewart-Rattray, CISA, CISM, CGEIT, CRISC, CSEPS, BRM Holdich, Australia, Director

Knowledge Board
Christos K. Dimitriadis, Ph.D., CISA, CISM, CRISC, INTRALOT S.A., Greece, Chairman
Rosemary M. Amato, CISA, CMA, CPA, Deloitte Touche Tohmatsu Ltd., The Netherlands
Steven A. Babb, CGEIT, CRISC, Betfair, UK
Thomas E. Borton, CISA, CISM, CRISC, CISSP, Cost Plus, USA
Phil J. Lageschulte, CGEIT, CPA, KPMG LLP, USA
Anthony P. Noble, CISA, Viacom, USA
Jamie Pasfield, CGEIT, ITIL V3, MSP, PRINCE2, Pfizer, UK

ACKNOWLEDGMENTS *(CONT.)*

Guidance and Practices Committee
Phil J. Lageschulte, CGEIT, CPA, KPMG LLP, USA, Chairman
John Jasinski, CISA, CGEIT, ISO20K, ITIL Exp, SSBB, ITSMBP, USA
Yves Marcel Le Roux, CISM, CISSP, CA Technologies, France
Aureo Monteiro Tavares Da Silva, CISM, CGEIT, Brazil
Jotham Nyamari, CISA, Deloitte, USA
James Seaman, CISM, CRISC, RandomStorm, UK
Gurvinder Singh, CISA, CISM, CRISC, Australia
Siang Jun Julia Yeo, CISA, CPA (Australia), MasterCard Asia/Pacific Pte. Ltd., Singapore
Nikolaos Zacharopoulos, CISA, CISSP, DeutschePost–DHL, Germany

Table of Contents

List of Figures

1. Introduction

Background

Enterprises continuously experience changes; some are forced by changes in the external environment, and others are driven by internal forces. Independent of the nature of changes, enterprises are not monolithic structures; rather, they are living organisms that must be tuned to the environment so that they can adapt, to not only survive, but thrive. When changes occur in one part of the enterprise without proper communication and coordination, signs of malfunction are likely to manifest as business disruptions, inefficiencies and potential financial losses. Configuration management (CM) reduces the risk of these malfunctions as part of a strategy to manage internal enterprise changes and minimize unforeseen impacts. As enterprises and technology become larger and more complex, enterprises have a more imminent need for formal processes to manage changes to the landscape configuration so that they can assess, in advance, the impact of these changes on business functions. The goals for every enterprise CM implementation are to maximize IT investment and enhance the likelihood of achieving enterprise objectives.

Purpose of This Publication

The purpose of this publication is to help enterprises create a homogenous view of CM and implement a sustainable process. Practice shows that enterprise stakeholders have varied ideas about the meaning of the term “configuration management” and what it entails, causing misalignment in the implementation of CM and the possibility of unmanaged expectations. CM is a strategic capability that supports many other activities within an enterprise, rather than a standalone process with simple objectives. Therefore, different stakeholders develop their own idea of what CM is based on their needs to interact with CM.

A number of challenges are linked to CM. This publication describes the most important challenges and formulates mitigating actions that are supported by COBIT 5 practices to manage configuration successfully. These practices are available in *COBIT® 5: Enabling Processes*, in process BAI10 *Manage configuration*. *Configuration Management: Using COBIT® 5* provides additional and more detailed practical guidance for both IT and business professionals relating to good practices in CM. Therefore, *Configuration Management: Using COBIT® 5* is complementary to COBIT 5 and its related products.

Who Should Use This Publication

CM has many stakeholders. In some enterprises, a dedicated group of CM professionals is responsible for CM activities. These CM professionals interact with all other stakeholders involved with CM because CM is a functional capability that supports other processes in an enterprise. Systems administrators and software developers have a major role in CM, but CM also supports the people who work in other functions such as project management, change management, help desk, incident management and capacity management, allowing them to gauge the impact of their activities on the overall environment. Audit and assurance can use CM as guidance or a source of information during the audit planning phase. IT service managers can use CM to estimate the impact that a change will have on business users. IT finance may rely on CM for cost accounting and recovery. This guide is intended to provide practical guidance for all stakeholders involved directly with or impacted by the CM process—CM professionals and all of the people involved in the capabilities that CM supports.

Scope and Approach

This publication begins by defining CM and providing a brief description of the process, its activities and its impact on other capabilities. One chapter is devoted to the significance of and the process to implement and maintain a configuration management database (CMDB) or a configuration management system (CMS). Risk and threats for CM and the CMDB are discussed in chapter 4. Chapter 5 presents good practices and mitigating actions related to the identified risk and threats. The last part of the publication provides guidance on how to conduct a COBIT 5 Process Capability Assessment to determine the level of capability of an existing CM process.

This guide is structured as follows:
- Chapter 1—Introduction
- Chapter 2—Configuration management: evolution of CM, goals and benefits, enablers overview, and the relationship of CM to other disciplines
- Chapter 3—Configuration management database (CMDB): definition, benefits, common data elements and best practices to build a CMDB; the differences between a CMDB and a CMS
- Chapter 4—Risk and threats related to CM
- Chapter 5—Mitigating actions: best practices intended to address identified risk
- Chapter 6—Process Capability Assessment for CM to support continuous improvement
- Chapter 7—High-level mapping of COBIT 5 and ITIL V3 for CM
- Chapter 8—Glossary
- Chapter 9—References
- Appendices—Practical guidance in the form of data design models, implementation project plan examples and audit checklist examples

2. Configuration Management

This chapter describes how CM evolved and provides insight into its goals and benefits, the process enablers, and links to other disciplines that interact with CM.

Evolution of CM

CM in the IT world originated from the need to control program source code. CM was meant to be a version control system that tracked and managed changes to program source code and documentation. When source code versions were created or modified, the source code control system allowed programmers to compare, restore or integrate program source code in a safe way. Later, the benefits of CM extended to product development and IT landscape management. For the remainder of this publication, the term "configuration management" means the process for recording and updating information that is related to the IT infrastructure.

COBIT 5[1] defines the purpose of CM as the process to provide sufficient information about service assets to enable the service to be effectively managed, assess the impact of changes and deal with service incidents.

Within IT service management (ITSM), the purpose of CM is to identify, record, control, report, audit and verify service assets and configuration items (CIs), including baselines, versions, constituent components, their attributes, and relationships to efficiently and effectively support other processes by providing accurate configuration information.[2]

Due to the complexity and interdependence of the multiple functions supporting the IT infrastructure, CM has evolved into a key process of accurately recording information about the IT environment and thus providing input to manage changes, incidents, problems and new projects. The IT environment can encompass software, hardware, network appliances, data structures, process definitions, documentation, credentials, personnel and any other element that is part of the IT infrastructure. The methods to implement CM can vary, but the goal should always remain the same: to have a repository where CIs and their relationships to other CIs are recognized and documented through their life cycle. Even small changes have the capacity to impact related elements. Therefore, for any proposed modifications to CIs, the impact on other CIs should be assessed with the related risk before any change is made. Approved changes and their impact on other processes and potential business users should be documented to maintain up-to-date configuration information.

[1] ISACA, *COBIT 5: Enabling Processes*, USA, 2012, p. 167, BAI10 Configuration Management
[2] "Configuration Management," Munich Institute for IT Service Management (mITSM), *http://www.mitsm.de/itil-wiki/process-descriptions-english/configuration-management*

An important distinction is the difference between enterprise CM and element CM:[3]
- Enterprise CM comprises the higher-level CM process, as explained later in this chapter, and the inventories of, and dependencies between, CIs. Enterprise CM can include questions such as:
  – What software, databases, servers, hardware, etc., are used by this service?
  – Which business processes will be impacted when we make a change to a particular application?
- Element CM involves the internal state of CIs. This includes questions such as:
  – When did attribute Y change for this database?
  – What is the patch level of this operating system?

This publication focuses most on enterprise CM, but aspects of element CM are sometimes mentioned because the distinction between the two is not always clear. The main reference for this publication is COBIT 5, which deals with the governance and management aspect of enterprise IT.

Goals and Benefits

CM goals are derived from stakeholder needs, which can be related to a specific set of COBIT 5 enterprise goals.[4] CM contributes particularly to the achievement of the following COBIT 5 enterprise goals:[5]
- **Managed business risk (safeguarding of assets)**—By recording and updating the information about CIs and their interdependencies, the enterprise can manage business risk that is associated with changes to the IT landscape.
- **Compliance with external laws and regulations**—CM includes verification of compliance of the enterprise landscape with an established baseline of regulatory requirements, to show compliance with external laws and regulations.
- **Business service continuity and availability**—A mature element CM process and supporting systems aids in resolving incidents faster (e.g., fast retrieval of current configurations, ability to perform fast comparisons between current and previous configurations to determine possible causes and simple rollback to previous stable configurations) and identifying ways to prevent reoccurrence, thus contributing to an increasing availability of business services.
- **Information-based strategic decision making**—CM enables management staff to make decisions based on documented information about the IT landscape and its critical components.
- **Optimisation of service delivery costs**—CM provides the ability to map the entire service delivery organization and interdependencies within the enterprise. The mapping contributes to detecting and eliminating redundant steps in the service delivery process and therefore reduces service delivery costs.
- **Optimisation of business process costs**—CM also helps map supporting IT hardware and software components to the service delivery business processes. In performing this mapping, CM contributes to optimizing cost by identifying inefficiencies and solving issues/problems much faster in the enterprise IT environment.

[3] Betz, Charles T.; *Architecture and Patterns for IT*, Morgan Kaufmann, USA, 2006
[4] See *COBIT 5: Enabling Processes*, pp. 13–15, for more information.
[5] Only the most applicable goals for this publication are mentioned.

- **Compliance with internal policies**—CM includes verification of enterprise environment compliance with a baseline of policy requirements, to show compliance with internal policies.

As described in the COBIT 5 goals cascade,[6] achieving enterprise goals requires a number of IT-related goals. CM contributes particularly to the following primary COBIT 5 IT-related goals:[7]

- **IT compliance and support for business compliance with external laws and regulations**—CIs contain baseline information that is aligned with external laws and regulations. Performing regular checks on actual settings supports IT and business compliance.
- **Optimisation of IT assets, resources and capabilities**—Periodic inventories of CIs provide the details that are needed to build an accurate map of significant IT assets present in the enterprise. This process identifies redundant IT assets and the relevance of, and need for, new IT assets when planning investments.
- **Availability of reliable and useful information for decision making**—A CMS may contain several attributes for each CI in an enterprise and logged in the CMDB. Management staff can run reports or statistics on these data to create reliable and useful information for decision making. CM reports and statistics can enable other processes that further contribute to the achievement of IT-related goals. For example, availability and resource utilization planning and portfolio management are some of the processes that can use input from CM to enable decision making that can positively impact the enterprise bottom line.

The operational benefits of CM can be categorized into key financial, compliance/legal, information security and operational benefits. Following are common CM benefits for each category:

- **Financial benefits:**
  - Capital:
    - Clearly defined service asset cost model
    - Efficiencies gained by reducing redundant purchases
  - Operational:
    - Clearly defined service cost models based on configuration dependencies
    - Lower support costs due to a decrease in reactive support issues
    - Lower network costs due to device, circuit, and user tracking tools and processes that identify unused network components
    - Prevention of business disruptions due to understanding of all interdependencies and assessment of the full impact of changes within the IT infrastructure
    - Minimized time to recover after a disaster and reduced associated costs due to maintaining an overview of related CIs and processes
- **Compliance/legal benefits:**
  - Better targeting of scarce security and governance resources due to CM dependencies that help governance, risk and compliance staff to understand which data and business processes are at risk

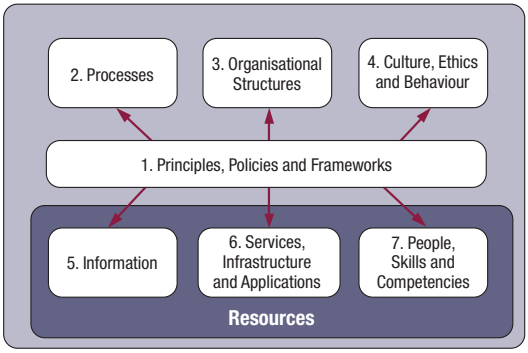
[6] See COBIT 5, pp. 17–18, for more information.
[7] Only the most applicable goals for this publication are mentioned.

  - Evaluation and integration of changes to components with full understanding of the risk and dependencies involved
  - Ability to define and enforce formal policies and procedures that govern asset identification, status monitoring and auditing. Furthermore, a CM model enforces the important aspects of a CI, as required by the business for governance purposes. CM activities further enforce a cohesive approach to controlled activities for a practical implementation of policies and procedures.
  - Baselines and enhanced traceability of changes to the systems, which add to the establishment of an environment that is more efficiently and effectively auditable and thus provides enhanced assurance
  - At the element level, standardized configuration settings, detailed guidance for secure and compliant systems operations, and baselines for traceability and control of changes to systems in scope
- **Information security benefits:**
  - Verification of the impact on related items and assessment of the risk of a proposed change to a system, which is a good starting point to maintain confidence in the completeness and integrity of information systems
  - An overview of the different lines of defense protecting an enterprise network and how they interact with each other, which allows for the discovery of "openings" in the lines of defense
  - Regular checking or tracking of CIs against the approved secure configuration baselines, which provides information that is needed to identify unauthorized breaches toward policies and procedures
  - A mature CM process aids in investigations relating to potentially harmful modifications of configurations, which is especially helpful after a security breach or operations disruption when management needs to understand what changes created the vulnerability
  - Version control and production authorization of hardware and software components, which assists in preventing vulnerable systems from being released into production
- **Operational benefits:**
  - Ability to quickly determine business user impact from system/network changes or outages
  - Ability to quickly determine the owners of a CI in case of questions, issues, problems or changes
  - Regular checking of CIs against the baseline to provide information that is needed to identify and plan quality enhancements
  - Business continuity benefits, including the ability to revert to previous versions of a CI or retrieve current versions of configurations

CM Enablers Overview

COBIT 5 defines seven enablers, as shown in **figure 1**. These enablers are broadly defined as anything that can help achieve the objectives of the enterprise. In this section, the key elements of the each enabler that is related to CM are presented and their contributions to the achievement of the aforementioned goals and benefits are described.

Figure 1—COBIT 5 Enterprise Enablers



Source: COBIT 5, figure 12

1. Enabler: Principles, Policies and Frameworks
Policies govern process execution, provide a high-level framework for process definition and implementation, and ensure consistency with the business and technology strategy of the enterprise.

Specific **CM policies** set the guidelines for all CM activities within the enterprise. These policies should be part of the overall enterprise policies framework and link closely to the operational and IT security policies. The following list shows good practice examples of statements that can be added to CM policies. Other, more specific statements that can be included in the policies depend on the specific circumstances and requirements of the enterprise:

- Only authorized personnel should have access to CM repositories.
- Access requests are approved and controlled through the proper IT management group.
- The CMS is the consolidated repository for all configuration information.
- Modifications to the established CM model go through the established change management process and require the approval of stakeholders who are responsible for providing oversight. In some enterprises, a dedicated group named the Configuration Change Board (CCB) may be established to provide oversight over the CM process and supporting systems.
- All CIs have defined owners who are responsible for maintenance, accuracy and review of the baseline settings.
- Modifications to the definitions of CIs and their relationships are reviewed and approved by stakeholders prior to implementation.
- The status of the CIs in the CMS is accurately maintained and authorized by stakeholders.
- A baseline is determined and approved for every new CI and before any major change to an existing CI.
- A standard naming convention is used in defining CIs and their relationships. The naming convention is followed according to procedures.
- All CIs have a defined criticality level (critical vs. noncritical and criticality duration) as identified through the business continuity processes and inferred by their dependencies to the services supported by them.

Subsequently, it is also good practice to determine a **CM model**[8] to guide all CM activities. The model details applicable policies and good practices and provides step-by-step guidance for the stakeholders who are involved in the implementation and maintenance of this process. For effective adoption, the model should be properly communicated to all stakeholders after its creation and after each update.

The elements of a CM model differ across enterprises, based on their needs, size, goals, etc. However, the following list contains some of the most important elements of a CM model that should be considered:

- **CM process or capability purpose**—The model defines the enterprise's specific purpose for CM and, most importantly, the scope of this process. Depending on the size of the enterprise, it may be too difficult and costly to keep track of every single element in the IT infrastructure; therefore, the first step in implementing CM is to define the scope that will be managed. This section addresses more specifically the following:
  - Which service assets and CIs, including versions, baselines and constituent components (attributes and relationships), will be identified, controlled, recorded, reported, classified, audited and verified
  - How the integrity of service assets and CIs will be protected through the service life cycle by establishing and maintaining an accurate and complete CM system over the entire IT infrastructure and having better control of the services. (See **figure 3**. BAI10.05 provides detailed guidance on how to verify and review the integrity.)
- **Scope of CM**—Determine the infrastructure components or CIs at a high level. This includes:
  - Data center hardware (computing, network, racks and power)
  - Data center software/applications
  - End-user computing hardware
  - End-user computing software/applications
  - Hardware and software configurations
  - Documentation suites (e.g., test plans, designs, organizational chart)
  - Business processes

  Similarly, all assets or application configurations (list of applications) that are specifically out of scope are determined and documented, including the reason for being out of scope.
- **Goals and capabilities of CM**—Specify enterprise goals for CM. These can include:
  - Support the business and customer objectives and requirements.
  - Support efficient and effective service management processes by providing accurate configuration information to enable people to make decisions in a timely manner.
  - Minimize the number of quality and compliance issues caused by improper configuration of services and assets.
  - Optimize service-related asset utilization and life cycle management.
  - Account for service costs by associating hardware and software with business applications and services.

[8] As described in *COBIT 5: Enabling Processes*, p. 168

  - Reduce service outages through better understanding of the technical state of systems components.
  - Reduce the need to continually reinventory CIs and their dependencies.
- **CM standards**—CM professionals work with other stakeholders to agree on standards and procedures. Standards involve enterprise decisions that represent preferences on components used in designing the IT landscape. These standard components can involve:
  - People (e.g., skill types, experience levels, internal/external sourcing)
  - Processes (e.g., system development life cycle [SDLC], change management, patch management)
  - Frameworks for effective IT service management (e.g., ITIL, TOGAF)
  - Technologies (e.g., operating systems, database technologies, programming languages, hardware platforms).

  These standards affect decisions that are made while developing the blueprint and design plans that support the desired operating model. CM standards are also agreed on to establish a consistent method of representing design objects (e.g., organization charts, process models, data models, application models and technology models). Each design object has a standard set of properties, also known as attributes and standard relationship properties to other design objects. These relationships assist in design analysis and planning.
- **CM terminology**—CM professionals work with the enterprise to establish standard business and IT terminology, or a common enterprise language. This common language is critical to relating different parts of the business. Terminology can include:
  - Business semantics (e.g., customer, service, product and vendor), the most important terms
  - People (e.g., functions, roles, responsibilities and capabilities)
  - Processes (e.g., order to cash, supply chain and financial reporting)
  - Technologies (e.g., data, applications and technology types and properties)

  The establishment of common terminology provides a common enterprise language and provides for a consistent "as-is" model data gathering process to design a "to-be" future state. While there is no common language that transcends enterprises and industries, good practice is to look for industry standards, as available, and use tools to capture and manage these terminologies. Although the CM team is responsible for managing the enterprise terminology repository, it is good practice to work with the industry, business and IT subject matter experts and the enterprise key stakeholders to establish this common language. It is critical that the organization subject matter experts not only participate in the establishment of enterprise terminology, but also continue to manage, update and steward these terminologies for continuity, adoption, refinement and long-term sustainability. A governance process for enterprise terminology management is recommended.

These design object properties, relationships and terminology provide consistency in:
  − Developing the design baseline
  − Conducting data gathering
  − Communicating the design plan for the future state
  − Assessing baseline compliance
- **Define CIs and related attributes**—The model should incorporate a list of all of the CIs and related CI attributes that are aligned to the services in scope. This definition brings clarity to planned and commissioned services.
- **Define services with CIs and their relationships**—Produce a list of all of the services in the enterprise at a high level, e.g., email system, financial system, Enterprise Resource Planning (ERP) system, communication services, network services. From this list, identify which services will be in scope for CM. Note: Depending on the size of the enterprise, controlling every process or service may not be a realistic proposition.
- **Establish criticality**—The criticality level of CIs should be derived from the services they support. Service Level Agreements (SLAs), Business Continuity Plans (BCPs) and Disaster Recovery Plans (DRPs) are sources of information to establish an accurate and realistic criticality level. **Figure 2** provides examples of how to define and document criticality and criticality time frame for CIs.

Figure 2—CI Example Documentation

Services	CI	Criticality	Criticality Time Frame
Email services	Exchange server	Critical	Work days–9:00 AM to 9:00 PM
Financial services	SAP systems	Critical	Dates–1, 2, 3, 4, 5, 26, 27, 28, 29, 30 and 31
Network services	Routers	Critical	24/7
Network services	Wireless routers	Noncritical	Not applicable
Communication services	IP telephony	Critical	24/5
Business continuity	Backup servers	Critical	24/7

- **Define configuration data design models.**—A data design model is developed for the services provided. The developed models do not imply architecture of the services and systems, but a logical relationship between CIs. A model template is created to establish the relationships between CIs. The template is a reference point for updating CIs in the future and to define relationships. See chapter 3 Configuration Management Database (CMDB) for an example of a configuration data model.
- **Determine process data requirements.**—The model describes data requirements, including classification, retention, archiving and destruction policies as defined by the enterprise records management strategy for all data types that are needed for each CI record and to update relationships to other CIs, as necessary. These requirements can be defined by using the different process life cycle phases and then describing the data required under each phase. Document whether the data requirements are mandatory or optional.

- **Identify process roles and responsibilities.**—Include role definitions, responsibilities and skills needed for each role. Roles and responsibilities, along with identified process owners, must meet business requirements. Multiple roles may be played by one person in the enterprise. Roles and responsibilities for the main stakeholders in CM are further discussed later in this chapter in the organizational structures section.
- **Assign process responsibilities.**—Individuals within the enterprise are assigned to each role based on their experience, skill set, knowledge of CM and professional certifications.
- **Determine procedures.**—Document the following in the model:
  - High-level process flow diagram
  - Procedure flow diagrams
  - Narrative in detail
  - Subprocess goals, scope, inputs and outputs

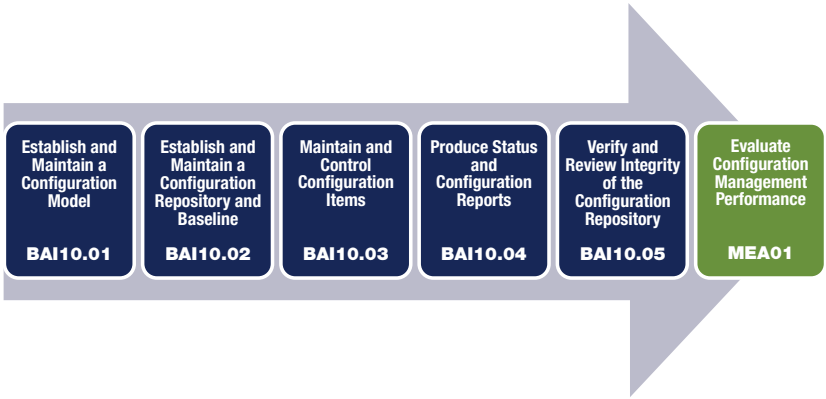
  Special attention should be given to improvement activities, access procedures and configuration control procedures to maintain the integrity of systems, services and CIs. Good practice is to also include configuration auditing procedures to specify CI variances—recorded, corrected and reported—and CM procedures to identify, track and control CI versions.
- **Detail configuration process interfaces.**—Define process interfaces, which entails detailing inputs and outputs for each process from data and activities perspectives. Interdependencies between processes are also established, so that changes can be managed to minimize breakdowns in operations.
- **Define measurements and controls.**—Define measurements and controls that are required to attain process objectives and goals. Define the reports that are required to monitor the health of the process and demonstrate attainment of objectives during periodic performance evaluations.

2. Enabler: Processes

COBIT 5 process BAI10 *Manage configuration* consists of five management practices. This section provides detailed activities for the five management practices, as described in COBIT 5 BAI10. **Figure 3** represents the CM process and its link to the relevant COBIT 5 BAI10 management practices. The sixth element is added to include the customized activities to perform periodic evaluations of CM performance, as recommended in the COBIT 5 process MEA01 *Monitor, evaluate and assess performance and conformance*.[9]

[9] See *COBIT 5: Enabling Processes*, pp. 203–206, for more information.

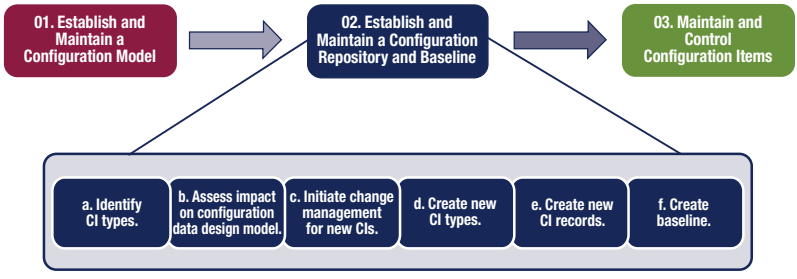
Figure 3—CM Process (BAI10 and MEA01)



BAI10.01 Establish and maintain a configuration model.
A configuration model is established and maintained in accordance with the enterprise's business and IT models and strategies and general IT guidelines and practices, to establish guidance for the CM stakeholders. Involvement of all stakeholders and all resources throughout the establishment of the CM model is critical to its success. The configuration model can be considered a logical model of the services, assets and infrastructure and specifies how to record CIs and the relationships among them. The model includes the CIs that are considered necessary to manage services effectively and provides a single reliable description of the assets in a service.

A more detailed description of the CM model can be found in the previous section of this chapter: Principles, Policies and Frameworks.

Figure 4—Establish and Maintain a Configuration Repository and Baseline



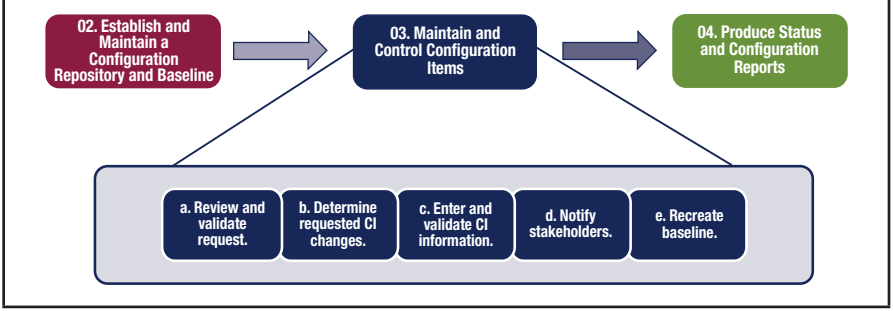
BAI10.02 Establish and maintain a configuration repository and baseline.
The repository and baseline activity cells shown in **figure 4** are described as follows:
a. **Identify CI types.**—Scan the IT infrastructure manually or through automation to determine the types of CIs. This represents the necessary CIs in the CMS.

b. **Assess impact on configuration data design model.**—Determine the impact of CI types on the CMDB data model. The result may be that a new CI type needs to be created and a change to the data model is required. It may not always be necessary to change the existing data model if the CI types can be represented in the existing model.
c. **Initiate change management for new CIs.**—If the impact analysis indicates the creation of a new CI type and/or a modification to the data model, a change request is created and approval is required from the CCB.
d. **Create new CI types.**—After required approval of a change request, a new CI type is created and/or a change to the data model is executed.
e. **Create new CI records.**—A CI record is created and categorized in the correct CI type in the CMS, with the necessary parameters and relationships.
f. **Create baseline.**—A CI is baselined when installed for the first time. This initial CI baseline will be the reference point for future changes. A configuration baseline should include:
   - Release records (current, past and planned)
   - Other change records (current, past and planned)
   - The status of the CI and its documentation when a change is approved and implemented
   - The status of the CI and its documentation when a package release is applied
   - Standard specifications on the hardware and software involved
   - Affected business processes and process owners

CI baselines should be recreated, at a minimum, every quarter or before a major change/release/project implementation. Each baseline should have a unique name, based on the following template:
Request Number_Descriptive Name_Date:
- **Request number**—The traceable number for the CI request
- **Descriptive name**—A short descriptive name of the baseline, with the preferred link to the appropriate project or release effort
- **Date**—The date of implementation, in the format of YYYY.MM.DD

An example of a baseline name is R999_SAP Release 5.6_2009.11.08.

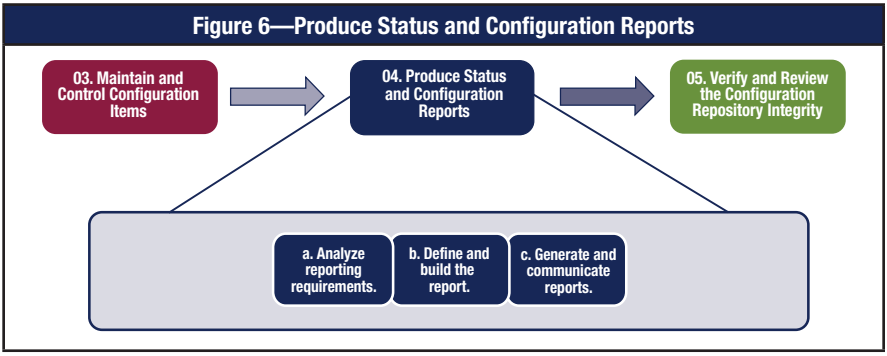
Figure 5—Maintain and Control Configuration Items



BAI10.03 Maintain and control configuration items.
When a request is received to create or change CI records, as shown in **figure 5**, the following steps are good practice to follow:

a. **Review and validate request.**—Review the request for accuracy of the provided information about the CIs and their parameters. This information is referenced to the CM model that is already designed and reviewed against its baseline to ensure completeness and accuracy. The request is reviewed to verify that its origin is from authorized sources. The request should also contain the necessary justification according to the defined policy. With the necessary information available, the request is approved or rejected.
b. **Determine requested CI changes.**—From the request, extract all the information about the CI that is necessary to add or update the record. Determine specific changes to CI.
c. **Enter and validate CI information.**—Make changes in the CMS based on the determined specific changes. These changes may involve creating new CIs, superseding CIs, updating CI information, adding CI parameters and superseding CI parameters. An initial validation can be considered by contacting the CI owner or physically checking the CI. If there are discrepancies, the update is rejected and the requestor is informed.
d. **Notify stakeholders.**—To increase awareness, changes to the CMS should be communicated to CI owners and other stakeholders. A distribution list should be defined and maintained.
e. **Recreate baseline.**—Determine the requirement for a new baseline creation. If required, follow the baseline guidelines as explained in BAI10.02, point f. For initial onboarding of data to CMS, the following activities are good practice:
   - **Scanning of CIs**—Perform a manual or automated scan to determine all of the CIs and their configuration parameters in the infrastructure.
   - **Identify CI errors**—Scanned data may require sanitization. Determine the errors based on nomenclature and policies defined by the CM model.
   - **Resolve errors**—Resolve the CI errors identified. To produce accurate data, investigate for correct information by performing initial auditing.

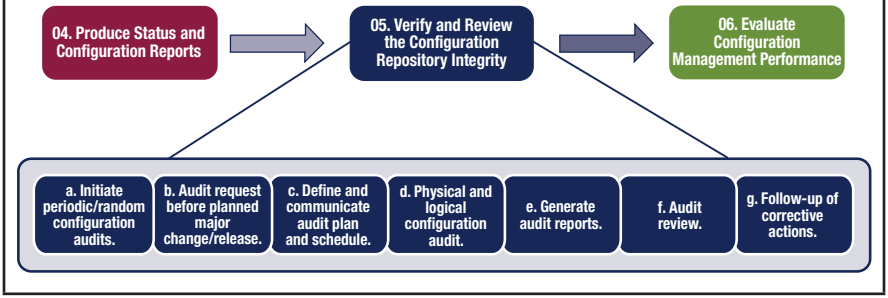
Figure 6—Produce Status and Configuration Reports



BAI10.04 Produce status and configuration reports.
CM is a data-centric process that can help generate many types of reports, e.g., simple inventories, exception reports, reports broken down by business units and dependency reports. Most of these reports should be defined and configured as part of the day-to-day reporting requirements; however, management may need new reports to make decisions. The steps shown in **figure 6** are good practice to follow to satisfy any reporting requests:

a. **Analyze reporting requirements.**—Determine whether the reporting requirements, including content, frequency and media, are compatible with the CM model. If this is not the case, a customization to the existing reporting structure is required and step b will be necessary.
b. **Define and build the report.**—If the report does not exist in the CM model, the new report has to be defined and built according to the requirements. This entails a change in the CM model and requires the approval of the CCB because the effort spent on customization of the report will be significant to the business.
c. **Generate and communicate reports.**—Produce the report and format the data according to the identified requirements and communicate and disseminate the report to all stakeholders. Following are examples of specific reporting requirements:
   - Identify status changes of CIs and report against the baseline.
   - Match all configuration changes with approved requests for change, to identify any unauthorized changes. Report the unauthorized changes to the change management authority.
   - Establish a procedure to add newly identified stakeholders to the distribution list of standard reports.

Figure 7—Verify and Review the Configuration Repository Integrity

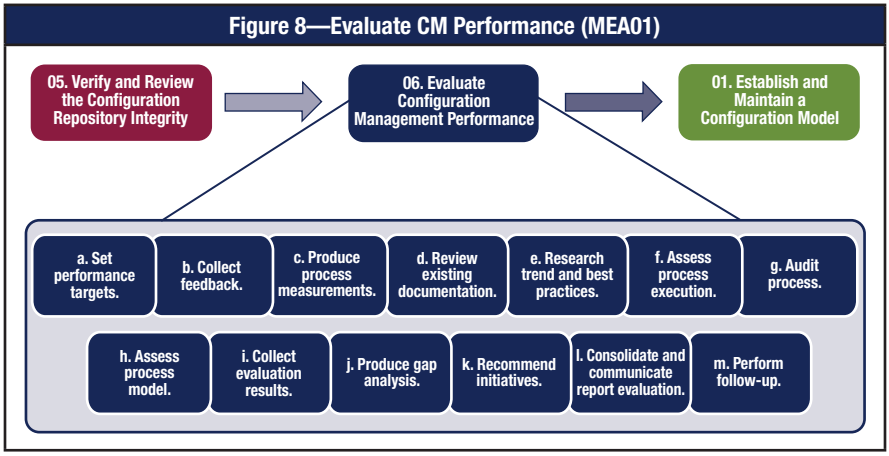


BAI10.05 Verify and review integrity of the configuration repository.
The configuration repository should be reviewed periodically and verified for completeness and accuracy against baseline settings. The descriptions for the cells in **figure 7** are:

a. **Initiate periodic/random configuration audits.**—An enterprise's audit activities are triggered by periodic or random needs for assurance on live CI settings. Periodic self-assessments should be scheduled quarterly or every six months. Random audits are through management request or as a requirement from a stakeholder organization to gain a perspective on critical activities.

b. **Audit request before planned major change/release**—Verification and auditing activities are performed prior to a major change or release to reduce the impact and enhance the success of a change/release. A service manager should make this request based on the insights related to this role.
c. **Define and communicate audit plan and schedule.**—The audit project plan should define a sizeable sample and the CI types in the scope. The schedule and resources required during the audit phase should be determined in the audit project plan. The audit project plan is communicated to management and stakeholders.
d. **Physical and logical configuration audits**—CI owners and auditors should be present to physically locate audit CIs in the scope. A verification of the actual logical configuration of the components against configuration records stored in the CMS is executed, and all deviations are noted in the final report. Audits on physical and logical configuration relationships are conducted for holistic results and efficiency of the process. A template is provided in appendix B.
e. **Generate audit reports.**—An audit report is generated for management, detailing the results and recommendations of the audit. Various other reports are created from the audit reports, as required for the enterprise, based on business needs. Examples are a discrepancy report and a compliancy report.
f. **Audit review**—An audit review is conducted with stakeholders across the enterprise. This review evaluates the final audit and discrepancy reports and identifies opportunities for improving the quality of the CMS or to remove any obsolete or inaccurate assets. This review provides a good opportunity to establish preventive methods to reduce future discrepancies.
g. **Follow-up of corrective actions**—Report and review all deviations that were noted for approved corrections or actions to remove obsolete or inaccurate assets.

Figure 8—Evaluate CM Performance (MEA01)



MEA01 Evaluate CM performance.
Figure 8 shows the activities associated with the MEA01 process. These activities are needed to perform a thorough assessment and evaluation of the CM performance, as described in the COBIT 5 enabling process MEA01 *Monitor, Evaluate and Assess Performance and Conformance*. The following activities are the minimum that should be executed to evaluate CM and determine whether the process is performing as intended.

a. **Set performance targets.**—Define specific key performance indicators (KPIs) and metrics for CM, and set minimum target performance levels for the KPIs and metrics. Examples of such metrics are provided in chapter 6 in the section on example KPIs and metrics for performance assessment.
b. **Collect feedback.**—Collect effectiveness and efficiency feedback from process reports, management, customers and users. Gather information from other processes that may point to this process as an area needing improvement.
c. **Produce process measurements.**—Collect process measurements to determine whether the objectives are being met. Evaluate effectiveness and efficiency measurements related to the process.
d. **Review existing documentation.**—Review existing process documentation including process descriptions, procedures, training materials, SLAs, operation level agreements (OLAs), underpinning contracts (UCs), etc.
e. **Research trends and best practices.**—Research trends and good practices from consulting enterprises, industry experts and IT service management enterprises to benchmark the process measurements and existing documentation.
f. **Assess process execution.**—Assess process execution and performance specifically against SLAs and impact on business. Assess all aspects affecting process execution, including tools, data, procedures, interfaces, communication, costs, meetings, training, skills, reporting, staff levels and organizational structure.
g. **Audit process**—Make use of the results and recommendations of internal or external audits reports.
h. **Assess process model**—Assess the current CM model maturity. Evaluate the process costs and value produced. Evaluate the ability of the model to support the process execution.
i. **Collect evaluation results.**—Collect all of the various evaluation results for further analysis.
j. **Produce gap analysis.**—Analyze information assessed and collected. Rate the current process capability level and determine whether a higher level of capability is desired or required. Consider IT plans and planned changes and determine the gap between current state and desired state. For more guidance on how to rate the current process capability, see chapter 6 of this publication.
k. **Recommend initiatives.**—Produce tactical and strategic recommendations with cost, benefits, risk exposure, priority and implementation approach details to allow for informed decision making.
l. **Consolidate and communicate report evaluation.**—Consolidate evaluation and communicate the evaluation to management.
m. **Perform follow-up.**—Follow up on the evaluation and recommendations made and ensure that corrective actions are being implemented.

3. Enabler: Organisational Structures

It has been established that multiple stakeholders are involved in the CM process. Furthermore, as described in this chapter in the section on CM in support of other disciplines, CM supports many other disciplines, thus increasing the number of stakeholders directly or indirectly involved. The level of involvement of each stakeholder depends on the size of the enterprise and the strategic goals CM is helping to achieve. This section describes in detail the involvement of key stakeholders.

Figure 9—Configuration Management RACI Chart											
	CM Structures				Related Structures to the CM Process						
Key Management Practice	Service/Configuration Manager	Configuration Analyst	Configuration Administrator/Librarian	CCB	Chief Information Officer (CIO)	Head IT Operations	Head IT Administration	Head Architect	Head Development	Business Process Owners	Audit/Assurance
Establish and maintain a configuration model	R				C	A	R	C	I	C	C
Define CIs, attributes, business model and relationships	A	R		C		C	C				
Approve request for new CIs and changes to CIs	R	R		R	A	I	C				
Register new CIs	A	R	R				C	C			
Establish and maintain a configuration repository and baseline	R				A		R	C	R		
Approve changes to CIs and CMS	R		R	R	A	C		C			
Produce status and configuration reports	I				I	A	R	C	C	I	I
Verify and review integrity of the configuration repository	R					A		R	R	I	C
Conduct configuration audits and report the nonconformance	A		R	I	I	C		C			R

The RACI (Responsible, Accountable, Consulted and Informed) chart in **figure 9**[10] is a high-level representation of different stakeholders in the CM process. A brief description of each group's roles and responsibilities in the CM process is also included.

- **Service/Configuration Manager:**
  - Accountable for the design and development of the CM process
  - Gathers and scrutinizes all business and functional requirements for CM tools, proposes changes to the scope and level of detail of CM and obtains necessary sign-offs
  - Ensures that the CM process is communicated effectively throughout the enterprise
  - Develops the identification system and naming convention
  - Agrees on uniquely identifying CIs in accordance with the naming conventions; ensures that staff complies with identification standards for object types, environments, processes, life cycles, documentation, versions, formats, baselines, releases and templates
  - Develops and maintains the interfaces to other process
  - Plans and implements the population of the CMS
  - Evaluates the existing CMS and its design, implementation, and management of new or improved systems for efficiency and effectiveness
  - Creates reports on the effectiveness, conformance and value of CM and the CMS

[10] The RACI chart is based on the RACI chart in *COBIT 5: Enabling Processes* for the BAI10 process. For presentation purposes, specific CM roles have been added and others have been removed.

- Plans and manages the CMS and central libraries, and ensures regular housekeeping of the CMS
- Organizes configuration audits
- Provides reports, including management reports, impact analysis reports and configuration status reports
- Follows up on identified gaps in the reports and initiates and drives improvement programs
- Provides training on the CM process to other stakeholders

• **Configuration Analyst:**
- Proposes the scope of the service asset and CM processes and functions
- Creates service assets and CM processes and functions, which includes CI registration procedures
- Supports the creation of the service assets and CM standards, plans and procedures, and their implementation
- Ensures that the correct roles and responsibilities are defined in the asset and CM plans and procedures
- Provides training on asset and CM to stakeholders of supported disciplines (see the section on CM in support of other disciplines in this chapter)
- Coordinates with the configuration administrator or librarian on population of the asset and CMS; manages assets, the CMS, central libraries, common codes, and data; ensures regular housekeeping of the assets and the CMS
- Ensures that developers and CMS users comply with identification standards for object types, environments, processes, life cycles, documentation, versions, formats, baselines, releases and templates
- Facilitates impact assessment for request for changes (RFCs) and ensures that implemented changes are authorized
- Creates change records, configuration baselines and package release records to specify the effect on CIs
- Ensures that the assets and CMS are updated when a change is implemented
- Uses the assets and the CMS to help identify other CIs affected by an error that is affecting a CI.
- Performs periodic and *ad-hoc* configuration audits
- Creates and populates project libraries and CMS; checks items and groups of items in the change management tools
- Accepts baseline products from third parties and distributes products
- Builds system baselines for promotion and release
- Monitors problems and maintains database for collection and reporting of metrics

• **Configuration Administrator or Librarian:**
- Controls the receipt, identification, storage and withdrawal of all supported CIs
- Provides information on the status of CIs
- Numbers, records, stores and distributes asset and CM issues
- Assists asset and CM with preparing the asset and CM plan
- Creates an identification scheme for CM libraries and the definitive media library (DML)
- Creates an identification scheme for assets and the definitive spares (DS)
- Creates libraries or other storage areas to hold CIs
- Assists in the identification of products and CIs

  - Maintains the current status of information on CIs
  - Accepts and records the receipt of new or revised configurations into the appropriate library
  - Archives superseded CI copies
  - Holds the master copies
  - Reports on change requests and track the progress through to completion
  - Issues copies of products for review, change, correction or information when authorized to do so
  - Maintains a record of all copies issued and notifies holders of any changes in their copies
  - Produces configuration status accounting reports and assists in conducting configuration audits
- **CCB:**
  - Oversees and directs changes to the CM model and all CM activities
  - Makes decisions on all strategic and operational initiatives, such as changing the scope of the CMS, providing restricted access, major modifications to the CMS, etc.
  - Manages the CM efforts to improve the effectiveness and responsiveness of the IT infrastructure, to better enable the business efforts to be in alignment with the corporate goals
  - Verifies that all new IT initiatives are in alignment with the CM model
  - Manages deviations and waivers to the CM model
  - Reviews, approves and prioritizes changes to the CM model for effective deployment of resources
  - Promotes and encourages interactions with other relevant disciplines, such as IT architecture, security, project management, business continuity planning/disaster recovery, internal audits, etc., to improve communication, coordination and cooperation
  - Escalates issues to the configuration manager
  - Enforces CM good practices and guiding principles

An important aspect to clarify, related to the CCB, is its relationship with the Change Advisory Board (CAB). While the CCB is a requesting CM authority, i.e., to make changes to the CM model, the CAB is an authorizing authority.

The CCB has access to information about the IT infrastructure, including a complete understanding of services, SLAs and contracts, policies and procedures, interdependencies of CIs, risk, relationships between CIs and services, ownership of CIs, knowledge of service design requirements, etc. With this knowledge and understanding, it is able to make decisions for initiating CM model changes when needed. After a request for change is raised by the CCB, the change management process is initiated, wherein the CAB is involved to authorize requested changes.

Although in practice both CCB and CAB seem to have common members, this is not considered good practice. An example of this is a person raising a change to the CM model from the CCB who is the same person in the CAB approving the change.

Ensuring segregation of duties is critical in setting up these two organizational structures. Good practice shows that the CCB should consist of the following stakeholders:

- Configuration managers
- Configuration analysts
- Configuration librarians
- Tools administrators
- For specific topics, when needed: business process owners and enterprise architects

Related structures participating in the CM process are:

- **CIO:**
  - Responsible for the support of the CM capability, including appropriate staffing and assignment of accountability for its performance and continuous improvement, and periodic review of evidence that it is performing and improving as expected
- **Head of IT Operations:**
  - Accountable for the entire change management process and the proper execution of all of the related tasks
- **Head of IT Administration:**
  - Responsible for establishing and maintaining a logical model of the services, assets and infrastructure and how to record CIs and the relationships among them. Included are the CIs that are considered necessary to manage services effectively and provide a single reliable description of the assets in a service.
  - Defines and agrees on the scope and level of detail for CM (i.e., which services, assets and infrastructure CIs to include) with the configuration manager and CIO
  - Establishes and maintains a CM repository and creates controlled configuration baselines
- **Head Architect:**
  - Consulted, when required, by the configuration manager, analyst or librarian in the daily execution of their tasks on the relationship aspects of the CIs
  - When participating in an audit, the head architect is responsible for verifying correctness of these relationship aspects of the CIs.
- **Head of Development:**
  - Responsible for identifying the CI types needed, and initiating a change request if a new CI type is needed
  - Responsible for providing input on specific values of CI attributes when creating a baseline
  - When participating in an audit, the head of development is responsible for verifying the correctness and accuracy of the CI information in the CMS
- **Business process owners:**
  - Main responsibility (of the supported disciplines by CM) is to define the business requirements and SLAs and support the CM professionals on any business-related question they might have, to ensure successful execution of the CM process. For more information see the section in this chapter on CM in support of other disciplines.

- **Audit:**
  - Conducts periodic CMS audits to check the accuracy, completeness, compliance and security of records against the baseline and also conducts audits on the CM process to ensure compliance to the CM model
  - Consolidates the observations and nonconformances
  - Provides reports to the CIO, configuration manager and other relevant stakeholders

4. Enabler: Culture, Ethics and Behaviour
Principles and policies are important mechanisms to communicate the expected culture and behavior to enable the enterprise to achieve its goals. However, policies must be enforced and compliance must be promoted through incentives and deterrents that help to create the desired behaviors.

Good practice for creating, encouraging and maintaining desired behaviors includes the following:
- Communication throughout the enterprise of desired behaviors and the underlying corporate values
- Awareness of desired behavior, strengthened by the example behavior exercised by senior management and other champions
- Incentives to encourage and deterrents to enforce desired behavior are visible
- Rules and norms, which provide more guidance on desired organizational behavior; these link very clearly to the principles and policies that an enterprise puts in place.

To get the maximum value from the CM process and the information in a CMS, a culture of communication, collaboration and discipline should be fostered, as follows:
- Communication—Open and overt so that facts are not omitted, misrepresented or understated. All parties involved recognize the value of communicating any decisions to all stakeholders in a timely and accurate way.
- Collaboration—Stakeholders and CM professionals are encouraged to collaborate. All individuals are respected as professionals and experts in their areas of responsibility.
- Discipline—Stakeholders and CM professionals follow policies and procedures to ensure that CM related goals are achieved.

Behaviors that can stimulate a culture where the employees are motivated to maintain and improve data quality in the CMS and management can feel confident when using CMS outputs to make decisions, thus maximizing the value of investments to build a CM process, include the following:
- Senior management sets direction and demonstrates support for CM practices.
- Senior management commitment is demonstrated by assigning the necessary resources and empowering employees to make decisions.
- Defined policies are documented, communicated and enforced.
- Business accepts ownership of risk. Responsibility and accountability are clearly defined and accepted. IT-related risk is not viewed solely as the responsibility of the IT organization.

5. Enabler: Information
The most important information items for CM are the CI records. Any relevant information about a CI is stored in its record in the official repository. For more information on the CIs, see chapter 3 on CMDB, where they are described in detail.

Other key information items that are part of the CM process include the different reports that can be created using CI data. There are two types of reports:
- Business reports—Designed for management to report on the status of CM
- Operations reports—More detailed and provide insight into the status of CIs, related changes, incidents and problems

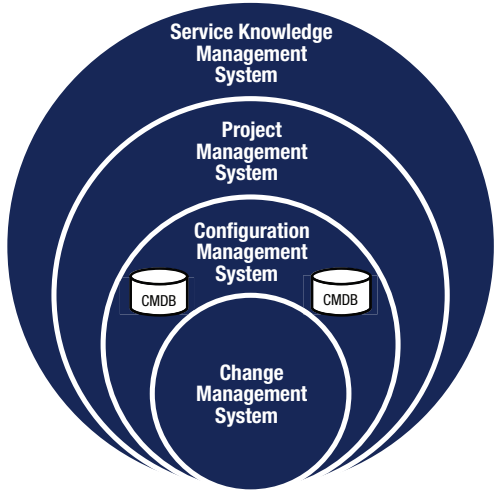
The CMS should have sufficient standard reports pre-built to handle most of the day-to-day business and operations reports.

Examples of the different reports that can be prepared using CI data are:
- **Business reports:**
  - CM "database variance"
  - Completeness and accuracy of the business process mapping for supporting CIs
  - Regulatory reports for in/out of scope systems
  - Business continuity support report
  - Audit reports, in a predetermined format
  - Auditors "action taken" reports
- **Operations reports**—Based on business requirements, can detail a variety of information (one of the most common information items included in these reports is total number of CIs):
  - Percentage of unregistered CIs
  - Total number of changes
  - Percentage of unauthorized changes in the CMS, level of accuracy of status of CIs
  - Status account of CI records in the CMS
  - Number of changes related to a specific CI (can be a separate periodic report)
  - Number of incidents related to a specific CI (can be a separate periodic report)
  - Number of problems related to a specific CI (can be a separate periodic report)

6. Enabler: Services, Infrastructure and Applications
Three key infrastructure elements are related to CM:
- Element CMDBs—Repositories used to store baseline configurations throughout their life cycle
- Enterprise CMDB—Repository used to store CI relationships and dependencies
- CMS—Contains one or more element CMDBs and one enterprise CMDB to manage CI attributes and their relationships across the enterprise. A CMS is an architecturally integrated, federated suite of systems providing information for the management of IT. See chapter 3 for more detailed information on the CMDB.

The service knowledge management system (SKMS) is a set of tools and databases used to manage knowledge and information. It is also part of the infrastructure supporting the CM process. The SKMS includes the CM system as well as other tools and databases used to manage IT-related services. The SKMS stores, manages, updates and presents all information that an IT service provider needs to manage the full life cycle of IT services. **Figure 10** depicts the overlapping relationship between the CM and SKM process/systems.

Figure 10—SKMS Overlapping Relationship With CMS



7. Enabler: People, Skills and Competencies

To build and sustain an effective and efficient CM process, CM professionals should have specific skill sets and competencies as shown in **figure 11**.

Figure 11—Skill Set Descriptions

Skill	Description
Enterprise expertise	General understanding of the enterprise and business environment and knowledge relevant to areas of responsibility. A basic understanding of the enterprise objectives is important to be able to place assurance work in context to the organizational goals.
Data management and data quality	Proficiency in managing large volumes of information and organizing data in a structured way
Problem solving	Ability to efficiently solve problems in a structured, analytical way. CM is a complex undertaking, with many dependencies. CM professionals should have a structured approach to solve problems and to deal with the complexity.
System development life cycle (SDLC)	Ability to understand and assess technology development life cycles and solid understanding of how development teams perform their functions

Figure 11—Skill Set Descriptions *(cont.)*	
Skill	**Description**
Advanced IT concepts	Ability to understand and maintain skills to ensure an advanced level of technical expertise relevant to the areas of responsibility (e.g., security, database administration, report writing)
Communication	The CIs in a CMS have many dependencies and relationships with many other CIs. To fully understand dependencies and solve problems in these areas, it is important to have good communication skills to be able to talk to the different stakeholders and understand these dependencies. Communication also refers to the ability of IT-minded professionals to express themselves in business terms and transform business needs into IT requirements.
Team coordination	Ability to maintain productive relationships and well-coordinated tasks within the CM team. Because the CM environment is complex and has many dependencies with other disciplines, it is important to coordinate and share all the information that is available in an efficient way.

Professional certifications may be necessary depending on the level of complexity of the process and supporting systems. Widely recognized training such as the COBIT 5 Foundation certificate (ISACA), CMDB certification[11] (APMG International) or ITIL certifications are recommended for CM professionals, because this training is designed to provide a comprehensive insight into the concepts underlying the service assets and CM process as well as related concepts in other processes across the IT service life cycle.

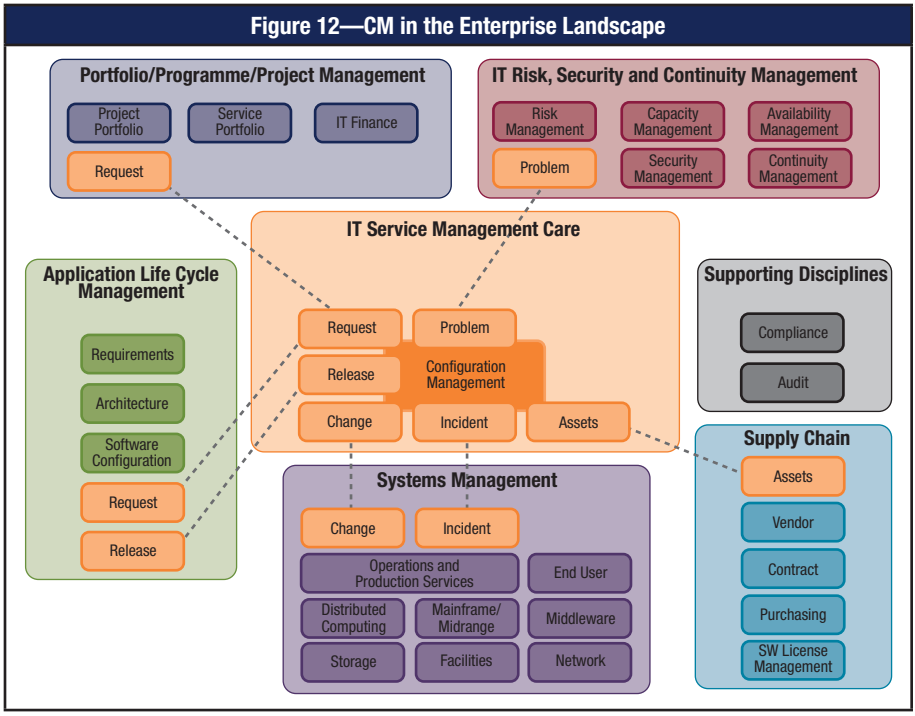
CM in Support of Other Disciplines

IT Service Management (ITSM) is the core of the IT landscape. It manages the:
- Change requests and changes to the IT environment
- Resolution of incidents and problems
- Releases to the production environment

CM is at the core of the ITSM discipline and thus supports many other disciplines in the enterprise. CM is in no case a standalone discipline, but has many links with other disciplines in the enterprise.

Figure 12[12] provides an overview of the key links that CM has with other disciplines. The remainder of this section discusses each of these links and explains how this process can support the different disciplines in the enterprise landscape.

[11] APMG-International CMDB Certification
[12] Based on Betz, Charles T.; "The High Value CMDB," EMA, 14 June 2012, *www.brighttalk.com*



ITSM Core
CM is at the core of ITSM in an enterprise. CM supports the many functions that ITSM performs within an enterprise.

CIs and their relationships contain information that is needed for efficient and effective incident identification and resolution and for subsequent problem analysis and solution identification. Resolutions for configuration-related incidents and problems associated with CIs are subsequently documented in the CM system. During the cycle from request to the release of a change, the information stored in the CIs can be used to perform a change impact assessment; perform a risk analysis categorization; set up a release plan; and perform the building, testing and actual implementation of the change. Consequently, this cycle leads to verified, updated and audited CIs.

Next to the more technical aspects of service management, CI information can also serve as the basis for establishing and monitoring SLAs.

IT Risk, Security and Continuity Management
COBIT 5 Processes: APO12 *Manage risk*, APO13 *Manage security*, BAI04 *Manage availability and capacity*, DSS03 *Manage problems,* DSS04 *Manage continuity*

The dependencies/relationships maintained in the CMS are able to translate raw event data into business impacts and raw performance data into business views on IT consumption, supporting, respectively, the disciplines of availability and performance

management. This translation into, and understanding of, real business impact is critical for providing enhanced availability and performance services to the customers (especially internal) of these disciplines. Business users find it far more convenient to have capacity broken down to a list of 40 to 50 applications, with the top three identified that they should be worried about, based on an understanding of the business needs and value, compared to having a list of servers with their capacity limits.

Imagine an exploit in progress and many alarm bells ringing for device PRD56984. These data do not provide enough information for effective risk and security management. More information is needed, such as what is the business purpose of this device/application, to assign the appropriate level of importance and urgency to the exploit. The appropriate actions to take will differ if the application is the financial system of the enterprise or if it is a database containing only contact details. The CMS contributes by establishing this link to the business purpose.

CM aids continuity management in the way that the CI relationship information and status determine the resources affected and required to be restored during service degradation. Continuity management actions and requirements (e.g., recovery time objective [RTO], recovery point objective [RPO], and risk scores) are documented in the CMS.

Portfolio/Program/Project Management
COBIT 5 Processes: APO05 *Manage portfolio*, APO06 *Manage budget and cost*, BAI01 *Manage programmes and projects*

CM can play an important role in IT finance because the CI attributes can serve as an information source for determining the cost of the enterprise assets. Service cost models can be set up and those data can be used as a basis for chargeback. Practice shows that once the CMS is effectively used for cost recovery and related finance matters, it has been accepted by the enterprise. The CIs are labeled with the current value of depreciated assets as output from this process.

Application Life Cycle Management
COBIT 5 Processes: APO03 *Manage enterprise architecture*, BAI02 *Manage requirements definition,* BAI03 *Manage solutions identification and build*, BAI07 *Manage change acceptance and transitioning*

CM plays an active role in supporting enterprise architecture. The agreed-on design modeling standards and enterprise terminology are stored in the CMS and are used by the architects to create an enterprise-compliant architecture.

During the software development and configuration process, specific CM tools aid in software source-code control. This originated in the first use of the term ”CM” and continues to be a part of CM practices today.

The CMS also provides a clear overview of who and how many are running which version of a program, allowing for effective decisions regarding software choices, as explained in the section on supply chain.

Systems Management
COBIT 5 Processes: BAI06 *Manage changes*, DSS01 *Manage operations*, DSS02 *Manage service requests and incidents*

CM aids the systems management discipline indirectly by supporting the efficient and effective handling of change and incidents to the systems, as explained previously. Configuration and systems management are also directly linked because the CMS can contain (depending on size and complexity) the complete configuration of the enterprise system landscape and individual components in the landscape. All details about these individual components are found in the CMS, providing an extensive repository of information for appropriate management of the systems in an enterprise.

Supply Chain
COBIT 5 Processes: BAI09 *Manage assets*, APO09 *Manage service agreements,* APO10 *Manage suppliers*

A CMS aligned with the IT asset management capability can have extensive data on IT assets. These data, in turn, can be aligned with the enterprise fixed asset system to enable the supply chain group to manage the inventory of assets in an efficient way and to prepare a buying strategy that takes into account the current inventory to avoid purchasing redundant assets. This alignment can also provide the enterprise with plenty of information to have a good bargaining position in future negations with vendors.

The software packages that the enterprise has acquired are included in the assets. The CMS can provide valuable information for software license management because it can record the item, along with license type and number of seats. Software license noncompliance is a dangerous situation for any enterprise due to the hefty penalties that can be imposed by software vendors and the legal liabilities. Including such information in the CMS can help manage licensing requirements effectively, thus providing an additional layer of protection for the enterprise.

Other Supported Disciplines
Using a CMS creates an auditable trace of changes. A baseline is defined in the CMS. Regularly verifying real-life settings with the baseline allows for enhanced verification of compliance with baseline and early detection of deviations. External and internal audit can benefit from a CMS by using the baselines and audit traces in the CMS to conduct a more efficient and effective audit.

3. CMDB

The discipline of CM requires managing a large number of information items, with multiple interdependencies, across the entire enterprise. For large enterprises with complex or dispersed facilities, this could be impossible to handle without the support of specialized tools that are designed to capture the necessary information before the CMDB can be populated. (In less complex environments, CI data can be captured using spreadsheets to normalize data prior to CMDB population.) This chapter includes:
• A definition of CMDB
• Details about the specific business and technical benefits of using a CMDB
• The benefits of using a data design model
• A list of good practices to build a robust and sustainable CMDB

Definition

Element CMDBs are repositories used to store baseline configurations throughout their life cycles. Enterprise CMDBs are repositories used to store CI relationships, dependencies and the history of any changes. A CMS is an architecturally integrated, federated, suite of systems providing information for the management of IT. CMS can contain one or more element CMDB and one enterprise CMDB to manage CI attributes and their relationships across the enterprise. A configuration record consists of CIs and their relationships with other CIs. All CIs have configurable attributes.

Centralized vs. Federated CMBD

A centralized CMDB contains all CIs and their relationships.
Pros:
• Less duplication of records
• “One source of record”
• Appropriate for small environments

Cons:
• Difficult to build and maintain due to potential size
• Performance degradation due to potential size

A federated CMDB model consists of one central CMDB to store the source of record for CIs containing limited information and the relationships with other CIs. The central CMDB is linked to other CMDBs containing the rest of the CI information and requests CI details on as-needed basis.

Pros:
• Appropriate for large and complex environments
• Easier to build and maintained
• May use legacy data repositories

Cons:
- Additional investment needed to procure multiple data repositories
- Additional overhead to maintain interfaces between the central and dispersed CMDBs
- Redundant data may exist
- Requires data normalization for interfaces to work and to eliminate duplicate records

Benefits

There are various business and technical benefits in using a CMDB. The most relevant benefits are:

- **Business benefits**
  - **Structured control over technology assets through improved tracking and visibility**—A single source of information with CIs and their configuration parameters defined. These relationships with other CIs establishes visibility through the infrastructure. The CMDB contains status updates of all of the activities of the CM process, in a controlled manner, emphasizing tracking and control as required by the business. Improved tracking is also a key aspect for audits and contributes to more efficient and effective audits.
  - **The enterprise's system reliability is enhanced through more rapid detection and correction of improper configurations that can negatively impact performance.**—CM process activities, such as authorization of CIs through the CCB, streamline the commissioning of any CI. CM also reduces improper configuration documentation. The auditing aspect of CM processes emphasizes the detection of erroneous updates to the CMDB and their correction. A rigorous control enhances reliability of systems with accurate data of CIs, parameters and status updates.
  - **Improved asset maintenance through the ability to better utilize proactive, preventive, and predictive measures**—Documentation of business needs in the CMDB provides access to the most recent information, which enhances decision making capability for the enterprise. Proactive and preventive measures help reduce the impact on financial loss due to service unavailability and unwarranted IT asset deployment. These measures also help optimize capacity planning and improve availability, thus providing a way to have a more predictable IT infrastructure.
  - **Greater agility through more accurate analysis of the impact of potential changes to hardware, software, firmware, documentation, testing procedures, etc.**—In a predictive IT infrastructure, a CMDB can help management anticipate actions to reduce the impact (risk) of changes, thus establishing greater agility.
  - **Enhanced reconciliation and management of complex systems and infrastructure**—Streamlined products, CIs types and classes, and design of configuration parameters enhance reconciliation and reduce the complexity of the infrastructure.
- **Technical benefits**
  - **Faster problem resolution, thus giving better quality of service**—A common source of problems is the incompatibility among different CIs. Having to detect these errors manually for a large group of components can consume considerable time and be less effective than doing so using a CMDB report to identify and solve a problem.

- **More efficient change management**—Accurate design of CI relationships in the CMDB has fewer chances of introducing incompatibilities and problems to the infrastructure.
- **Control of licenses**—It is possible to identify licensed and unlicensed copies of software, which can pose financial and security risk. Noncompliance with legal requirements leads to a negative impact on the enterprise (financial and reputational).
- **Greater levels of security**—An up-to-date CMDB allows the detection of infrastructure vulnerabilities and security threats when assessing CIs against recommended security settings.

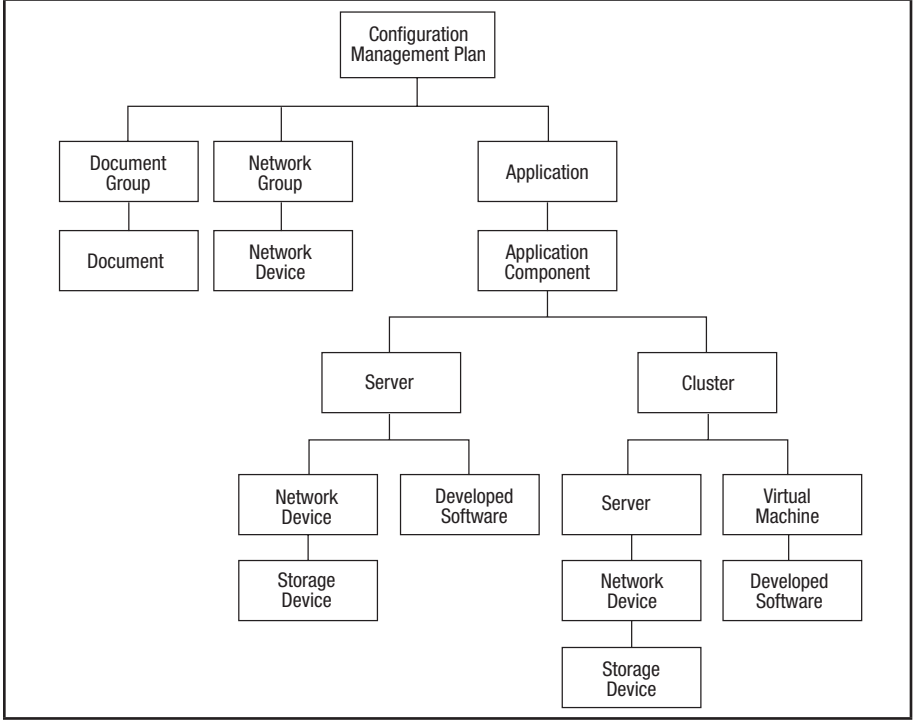
Data Design Model and Common Data Elements

The first step to building a CMDB is to define what is in the scope. The second step is to define a data design model to represent CIs and their hierarchical relationships to other CIs. A template to build a data design model in a structured way is shown in appendix C Data Design Template for a Data Design Model.

Figure 13 provides an example of a CI hierarchical relationship data design model:

- **Document group**—This CI class can be used to logically group documents (e.g., process documents, implementation plans, back-out plan, disaster recovery plans, run books, test plan, design documents):

Figure 13—Data Design Model Example



  - **Document**—Critical documents defined as CIs are put here and those CIs may be related as a child CI to any other CI in the model.
- **Network group—**This CI class can be used to logically group network components:
  - **Network device—**Critical network components can be stored as CIs to document their parameters and relationships.
- **Application**—Can be modeled where the major application is the parent of any subsystem:
  - **Application component**–An application may or may not have application components; this is an optional level that can be used to design the data model:
    - **Server**—Generically used for all hardware computing platforms:
      - **Developed software**—Represents application development efforts (possibly outsourced) where change management should be managing the impact of application changes on the environment and *vice versa*
      - **Network device**—Is used here to identify specific network devices that are dedicated to provide connectivity between a server and a storage device:
        - **Storage device**—Is used to model all external/standalone storage devices
    - **Cluster**–Is used to model any and all logical groupings of computing resources, such as virtualized environments, clustering software installations, or logical partitions (LPAR). A VMware® "farm," an installation of Windows clustering software, or LPARs on a mainframe are all examples of CIs that can be instantiated in this CI class. In virtual environments, a virtual machine (VM) can run on any hardware computing platform within the cluster, at any given moment in time. This model eliminates changing relationships when such moves take place. In mainframe environments, there is usually only one VM running on a particular hardware partition, but this model will accommodate partitions running in virtualized environments as well.
      - **Server**—Generically used for all hardware computing platforms
      - **Network device**—Is used here to identify specific network devices that are dedicated to provide connectivity between a server and a storage device:
        - **Storage device**—Is used to model all external/standalone storage devices
      - **Virtual machine—**Is used to model all VMs and their parameters
        - **Developed software—**Represents application development efforts (possibly outsourced) where change management should be managing the impact of application changes on the environment and *vice versa*

Example applications:
- Email (the parent) may have application components (children) of:
  - Outlook®
  - Lotus Notes®
  - BlackBerry®
- Accounting (the parent) may have application components (children) of:
  - Accounts payable
  - Accounts receivable

Definition and Example CIs

Any component that needs to be managed to deliver an IT service is a CI. Information about each CI is recorded in a configuration record in the CMDB, which is hosted in the CMS and is maintained throughout its life cycle by the CM process. CIs typically include IT services, hardware, software, network, buildings, people and formal documentation such as process documentation and SLAs.[13] **Figure 14** contains some of the most common examples of CIs

Figure 14—CI Examples

CI Name	Information
Network access point	• Communication endpoint • Internet Protocol (IP) endpoint • Local area network (LAN) endpoint • Protocol endpoint
System	• Application • Application infrastructure • Application system • Cluster • Computer system • Inventory location • Mainframe • NT domain • Printer • Software server
System components	• BIOS • Card • CD ROM • Database storage • Disk Drive • Disk partition • File system • Floppy drive • Hardware package • Hardware system component • Keyboard • Local file system • Logical system component • Media • Memory • Monitor • Network port • Operating system • Package • Patch • Pointing device • Processor • Product • Rack • Remote file system • Resource pool • Shared drives • System resources • System software • Tape drive • UPS • USB ports • Virtual system enabler

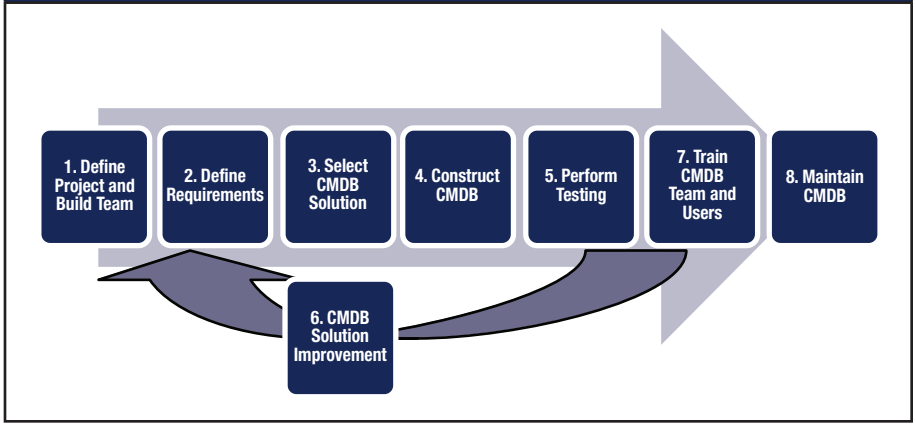
Some system CIs can be a composition of individual system component CIs. For example, a desktop computer consists of many different CIs, such as the applications to be installed, hard drive, monitor, keyboard, mouse and memory.

[13] ITIL V3

Good Practices to Build a CMDB

To build a sustainable solution, follow a standard approach for project management, as discussed in the COBIT 5 process BAI01 *Manage Programmes and Projects*.[14] Building a CMDB also requires specific details within the standard approach. This section provides an overview of the specific good practices that are required to build and maintain a CMDB, as shown in **figure 15**.

Figure 15—CMDB Life Cycle



1. Define project and build team.
As described in the COBIT 5 key management practices BAI01.02 through BAI01.04, the project definition involves people, technology, scope, quality and costs. Following are good practices specific to CM for these management practices:

- **Recognize resources for the project team that can fulfill the roles defined.**—The critical resources for a successful CMDB implementation are:
  - Executive sponsor
  - Project manager
  - CMDB owner
  - Other resources recognized in the CM model (See chapter 2, section on the CM enabler overview, point 1.)
- **Organize a project kickoff and finalize project plan.**—After the key resources and roles are identified, initiate a project kickoff. The project team determines the critical activities and develops a high-level project plan. The team assigns project activities to individual resources and develops and communicates a final project plan. The plan incorporates a project management framework.
- **Conduct training on CM and CMDB tool sets.**—Provide extensive training to all resources and roles that are on the project team. This training can lead to CMDB certifications (described in chapter 2, section on the CM enabler overview, item 7). Training can be online or classroom-based. Consider using some of the many publications on CM to enhance knowledge.

[14] See *COBIT 5: Enabling Processes*, pp. 119–127, for more information.

- **Create CMDB goals and mission statement.**—The executive sponsor and key stakeholders help the team to create the CMDB goals and mission statement. The statement and goals are considered to be a business direction and CM high-level requirements.
- **Agree on CMDB goals and mission statement.**—Communicate the CMDB goals and mission statement to the project team. A session to understand this critical statement can be helpful for all the resources by establishing a common understanding and reducing conflict of interest. Ensuring that the project team understands the mission statement is important for the success of the project and to be able to move the project in the direction that achieves the goals. The project team should formally agree to the proposed statement and goals. Later, the project team will deduce the statement and goals into a solution involving people, processes and technology (tools).
- **Define benefits of the CMDB.**—Define the specific benefits that the enterprise expects from the CMDB. (See the benefits section in this chapter.)
- **Build a business case for CMDB implementations.**—Include elements such as:
  - Executive summary
  - Problem statement
  - Solution summary
  - Goals and benefits
  - Cost-benefit analysis
  - Return on investment (ROI)
  - Metrics and measurements
  - Quality standards

The business case is reviewed by the executive sponsor and key stakeholders. Approval of the business case gives consent to start the next phase.

2. Define requirements.

The COBIT 5 process BAI02 *Manage requirements definition* provides guidance for defining the requirements. Following are good practices specific to CM for the BAI02 activities:

- **Gather governance requirements.**—Define an IT governance model encompassing a governance team, the regulatory compliance requirements, the IT service management standards, the quality standards and the business governance requirements.
- **Select supporting best practices.**—Frameworks such as ITIL and COBIT and standards such as International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 20000-1:2011 are service management industry best practices. ITIL applies to service process management, and COBIT provides CM activities to the business. Standards are the minimum activities to be established to improve quality and efficiency.
- **Identify potential problems and risk.**—The abundant project-team industry experience can be harbored in a brainstorming session to document all potential problems and risk to be addressed. Identifying these items provides visibility and speeds up mitigation actions to improve the probability of a successful CMDB implementation. The mitigation actions are tracked and monitored for completion throughout the project.

- **Define inventory requirements.**—The project team defines and agrees on inventory procedures and scope requirements, to establish the level of effort required. The requirements determine the CMDB implementation. The inventory can be created by manual or automated sources. A manual source requires a physical check to detect CIs. When an application scans the network to detect the CIs, the inventory is populated by an automated source. Depending on the inventory sources selected, time frame, resources, cost and schedules can be affected.
- **Define service catalog requirements.**—These requirements define the business services, system services and component services and how each of the CIs is related to the defined services.
- **Define audit requirements.**—Consider specific audit requirements for the CIs and related reports, which are required for audit purposes.
- **Define CMDB requirements to support other disciplines.**—CM interfaces with many other disciplines, such as incident management, problem management, change management, service-level management, availability and capacity management, asset management, and service continuity management. (See chapter 2, the section on configuration management in support of other disciplines.) A requirement to interface the CMDB with all of these processes should be defined for effective functioning. This holistic approach enhances and builds a credible CMDB.
- **Define CI level and IT service model.**—The definition of CI levels and a service model can be derived from an example given in the section on the data design model and the common data elements section in this chapter. This definition helps to build a CMDB to a level required for the business.
- **Define CI attributes.**—Define the CI attributes that are to be maintained in the CMDB. These are the required parameters for the business to be able to manage CM. Business determines the factors and the level of attributes.

3. Select CMDB solution.

The COBIT 5 key management practices BAI03.01 through BAI03.04[15] provide guidance for detailing solution components and selecting them. Good practices specific to CM for these management practices are:

- **Select a CMDB solution.**—Many solutions are available in the market. Select a solution based on the business requirements, functional requirements, level of automation required, etc. Rigorous assessment of the solution is necessary for successful alignment of the CMDB implementation with the business and IT landscape. While evaluating solutions, emphasis must be more on the CM process activities than on a tool or application.
- **Plan CMDB population.**—Draw a road map to populate the CMDB. Data can be populated through manual or automated means. Data population is a tedious, yet critical task. The quality of data reflects the success of its implementation.

[15] BAI03 *Manage Solutions Identification and Build*; see *COBIT 5: Enabling Processes*, pp. 133–139, for more information.

4. Construct CMDB.
The COBIT 5 key management practices BAI03.05, BAI03.06 and BAI03.10 provide guidance with building solutions and maintaining them. Following are good practices specific to CM for these management practices:
- **Construct CMDB.**—Install the related hardware and software. Establish integration policies as required.
- **Create CI life cycle management processes.**—As required by the business
- **Build supporting processes.**—Document CM processes and procedures. See chapter 2,the section on the CM enabler overview, point 2.
- **Populate CMDB.**—Populate the CMDB with data as stated in the plan and data model.

5. Perform testing.
The COBIT 5 key management practices BAI03.07 and BAI03.08 provide guidance with testing solutions. Following are good practices specific to CM for the management practices:
- **Develop testing scripts.**—Develop detailed test scripts to align the business requirements with the functions and configuration of the tool developed. Match each of the requirements with functions of the tool and/or the configuration design of the tool. Testing scripts provide step-by-step instructions for testing and subsequently documenting the test results in the form of a pass or failure. A prerequisite for obtaining a passed test script includes formal approval/sign-off from business users.
- **Develop failed test scripts.**—Document each of the failed test scripts in a bug tracking system, to be corrected and resolved to completion. In consultation with project stakeholders, prioritize each of the bugs detected and assign them to a technical resource for correction. Negotiate a target date so that the final go-live deadline is not affected. When a failed test result is corrected, it will have to be approved by the business users to be considered completely resolved.
- **Review posttest.**—Conduct a review of the CMDB after completing testing and resolving all of the bugs.

6. CMDB solution improvement.
The COBIT 5 process BAI05 *Manage organisational change enablement*[16] provides guidance with managing organizational change. One of the aspects applicable to CM is:
- **Perform iterative CMDB solution improvements.**—Assess the effectiveness of the constructed CMDB solution and take iteratively corrective measures, as appropriate. When corrections are identified to be necessary, define new requirements and repeat the process from step 2, as shown in **figure 15**.

[16] See *COBIT 5: Enabling Processes*, pp. 145–148, for more information.

7. Train CMDB team and users.
Another important element of managing change is training the users on the solution. This is also applicable to CM:

- **Communicate and train CMDB stakeholders.**—Train the project team and users on the CMDB technology and functionalities to ensure a smooth transition to changed or new work processes and operations and minimal frustration when working with the CMDB.

8. Maintain the CMDB.
The detailed good practices, as described in the section on good practices to build a CMDB, allow maintaining the CMDB after it is constructed. Creating CI life cycle management processes and building supporting processes allow for adequate support for maintaining the CMDB after its construction.

4. Risk and Threats Related to CM

Creating awareness within the enterprise about risk and threats related to CM should be part of the strategy to implement and sustain the CM process. This chapter provides a list of common threats that can impact CM and the risk categories related to those threats.

Common Threats in CM

Following are the most common CM threats and related risk (financial, operational and legal/compliance):

T1. Confusion on definition/scope/meaning:

- **Threat description**—Experience shows that many people within one enterprise have different definitions for CM. When talking to each other, they are discussing the topic based on their individual viewpoints and often fail to understand each other. This confusion regarding CM can be related to the fact that CM is a supporting discipline for many other disciplines, as described in chapter 2, and thus fulfills many different tasks for each of these disciplines.

  For example, a CMS built to support the audit function could fulfill other requirements such as change management or software development control. All of these requirements should be part of the initial CMS design to effectively support practices that require CM services.
- **Risk**—Substantial operational risk is attached to this threat because the proper functioning of CM and the CMS can be in danger. Employees may not be motivated to maintain a CMS if it does not support their needs. When people feel that they do not understand each other about the value of CM or the CMS, their problems will not be solved and they will not maintain the CMS accordingly. The data quality can deteriorate quickly, making the CMS useless. Financial risk is also attached to this threat. Large amounts of money can be lost due to misalignment, waste of time, redundant activities and decisions based on poor data.
- **Risk categories**—Operational and financial

T2. Immature guidance on CM:

- **Threat description**—CM is a relatively new idea that has proven to be difficult to implement in large and complex environments. Lack of experience, knowledge, skills and understanding; lack of clear documentation to make implementation practical; and lack of skills to design and build the technology pose threats.
- **Risk**—Unprepared enterprises face operational risk. Unavailability of experienced resources and industry guidance on CMS could increase the cost of implementation and operations. Failure to curtail this risk can also pose a legal and compliance risk to enterprises if decisions are made using the wrong data.
- **Risk categories**—Operational, financial and legal/compliance

T3. Implementation challenges for CM:
- **Threat description**—The implementation of CM creates multiple challenges. A common issue is that enterprises tend to implement a fully automated tool without a CM strategy, policies and processes to support it. In addition to premature automation, often the benefits of CM are not clear. The value of automation and ROI remains a major point of concern for project sponsors and other stakeholders throughout, and after, the implementation. A CMS by itself does not solve all the business and IT problems and related threats.
- **Risk**—Full automation of CM is possible at a very high cost. Optimal automation that takes into account the ROI is a way to improve the capability of the process and reduce human error. The high cost incurred because of a CMS implementation can lead to substantial financial losses, especially when projects fail to deliver the value expected. The CMS may never become fully operational. A process and value definition should be fully defined before making the decision to automate.
- **Risk categories**—Operational and financial

T4. CM resource management:
- **Threat description**—A thorough understanding of IT infrastructure, data center management, and business processes is critical to the CM process. Aligning IT processes to business requirements is a key principle of frameworks such as COBIT 5. People who have a mixed skill set that includes knowledge of COBIT, ITIL, data center and IT infrastructure management, and good practices for CM are rare. Enterprises are taking fewer initiatives to draw a career path along these lines, causing cost to escalate due to unavailability of resources at the right time. People who do have specific knowledge of CM systems and CMDB design require expensive salaries and are scarce.
- **Risk**—Unavailability of people with adequate skills leads to operational issues. A project may fail without the right people at the right time, bringing in the risk of noncompletion. Lack of clarity on investments and underutilized assets is a financial risk that leads to misuse of the enterprise's resources and decreasing financial health.
- **Risk categories**—Operational and financial

T5. Uncontrolled maintenance of CIs:
- **Threat description**—After implementing a CMS, many enterprises fail to provide appropriate maintenance to the CMDB. Processes necessary to ensure that the CMS is maintained and that all database information is up to date are not established, increasing the risk of using stale information for strategic decisions. Lack of proper maintenance can also result in database corruption.
- **Risk**—Uncontrolled maintenance results in random changes/updates to the CIs. This represents a serious threat to the integrity and availability of correct information in the configuration records, resulting in operational issues. No reliance can be placed on the data by supported disciplines such as incident management, problem management, etc. Baseline verification is also impacted because there is no assurance that the latest approved configuration is indeed the current baseline.
- **Risk categories**—Operational, financial, strategic and legal/compliance

5. CM Mitigating Actions

Awareness about the importance of proper CM constitutes the first step to implementing an effective process. This chapter describes mitigating actions, good practices and recommendations to manage the threats and risk defined in chapter 4.

Mitigating Actions Using COBIT 5

This section describes the major mitigating actions for the threats and risk related to CM. With the implementation of these mitigating actions, the impact and probability of a risk event can be greatly reduced. However, residual risk may still exist, and periodic evaluation of the CM process is always advised.

The mitigating actions are:

1. Define a clear strategy for CM in the enterprise.
- Related threats: T1, T2, T3, T4, T5
- Mitigation—Defining a strategy is an action that upper management should take to help the enterprise and its various organizations meet business goals. This strategy defines the directions and plans that should guide business and IT performance to help them meet the business objectives established for them. A vision, mission statements, goals, objectives and benefits of CM, as defined by management, set the strategic direction for this particular process and related activities. These statements and a defined enterprise architecture provide a strong foundation for the implementation and maintenance of the CM process. The next step in establishing a strategy is to transform statements into activities, tasks, roles and responsibilities, project standards, reports, etc., for operational actions.
- Related guidance in COBIT 5:
  – Process: APO02 *Manage strategy*
  – Enabler: Principles, Policies and Frameworks

2. Link CM to the enterprise overall governance model.
- Related threats: T1, T2, T3
- Mitigation—Include CM in the enterprise governance model. The governance model should define the CM stakeholders and their responsibilities, escalation procedures, reporting requirements and frequency, governance bodies, etc.
- Related guidance in COBIT 5:
  – Processes: EDM01 *Ensure governance framework setting and maintenance*, BAI10 *Manage configuration*
  – Enabler: Organisational Structures

3. Define the CM process before automating.
- Related threats: T2, T3, T5
- Mitigation—After establishing a clear strategy and governance model, it is advisable to take some time to determine the level of automation that is required to operate the envisioned CM process properly. Too many enterprises tend to start immediately with a full automation of the process, although full automation is not necessary and creates more problems than solutions. Think about which parts of the CM process need to be automated and which parts are unnecessary to automate.

- Related guidance in COBIT 5:
  - Processes: BAI01 *Manage programmes and projects*, BAI02 *Manage requirements definition*, BAI03 *Manage solutions identification and build*, BAI10 *Manage configuration*
  - Enablers: Principles, Policies and Frameworks; Processes; Information; Services, Infrastructure and Applications; People, Skills and Competencies

4. Establish policies and procedures for CM.

- Related threats: T1, T2, T5
- Mitigation—Standardized procedures and guidelines to build processes and tool sets in which risk is managed and decisions are made on an objective basis by stakeholders. Policies and procedures are the tools to establish this working environment and can support all CM-related activities.
- Related guidance in COBIT 5:
  - Enabler: Principles, Policies and Frameworks

5. Formulate clear requirements.

- Related threats: T1, T3
- Mitigation—The requirements definition forms the basis for establishing the CM model, CI life cycle, deliverables, quality of services, cost, etc. Poorly defined requirements may cause low quality of services and may also lead to financial impact due to scope creep. A prerequisite is that business process owners, IT, legal and other stakeholders must reach a common understanding of the requirements before proceeding to the next step.
- Related guidance in COBIT 5:
  - Processes: APO06 *Manage budget and costs*, APO11 *Manage quality*, APO12 *Manage risk*, BAI02 *Manage requirements definition*
  - Enabler: Information

6. Perform an adequate analysis on products and tools selection

- Related threats: T3, T4
- Mitigation—An intensive analysis to align the business requirements to the tools available in the market is an important step for the successful implementation and a better ROI of CM. The assessment is carried out based on various parameters, such as the level of automation, functional requirements from a business perspective, data alignment at a high level, security requirements of the business, time to deploy, cost, etc. A decision can be taken through consultations with independent organizations providing similar services. Previous success rate through a references approach can also be considered.
- Related guidance in COBIT 5:
  - Processes: APO04 *Manage innovation*, BAI03 *Manage solutions identification and build*
  - Enabler: Services, Infrastructure and Applications

7. Apply the standard project management activities to CM projects.

- Related threats: T2, T3
- Mitigation—Manage CM projects from the investment portfolio in alignment with the enterprise strategy and in the same coordinated way as other projects. Initiate, plan, control and execute CM projects and close with a postimplementation review. Best practices are:
  - Maintain a standard approach for the CM project
  - Manage stakeholder engagement
  - Develop and maintain the CM project plan
  - Launch and execute the project

  - Monitor, control and report on the project outcomes
  - Manage project quality
  - Manage project risk
  - Manage project resources and work packages
  - Close project with review post implementation
- Related guidance in COBIT 5:
  - Processes: APO06 *Manage budget and costs*, APO07 *Manage human resources*, APO11 *Manage quality*, APO12 *Manage risk*, BAI01 *Manage programmes and projects*, BAI07 *Manage change acceptance and transitioning*
- Enablers: All

8. Plan reviews, reviews and more reviews during implementation.

- Related threat: T3
- Mitigation—For implementation, considerable data are required to make decisions on scope, level of defining CIs, scope of CIs, relationships, etc. At any level, a small error in the system can be disastrous. Small details must be reviewed periodically to ensure quality and consistency. Reviewing upfront saves considerable time and effort in a later stage, when searching for and correcting errors can have a greater effect on time and cost.
- Related guidance in COBIT 5:
  - Processes: BAI01 *Manage programmes and projects*, APO11 *Manage quality*
  - Enabler: Information

9. Manage changes to CM in a controlled manner.

- Related threats: T3, T5
- Mitigation—Manage all changes to the CM project and postimplementation operational activities in a controlled manner. This mitigating action pertains to any change relating to business processes, applications and infrastructure. The "controlled manner" implies the following:
  - Using change standards and procedures
  - Performing an impact assessment
  - Prioritization and authorization
  - Tracking
  - Testing
  - Reporting
  - Closure
  - Documentation updates
- Related guidance in COBIT 5:
  - Processes: BAI05 *Manage organisational change enablement*, BAI06 *Manage changes*
  - Enablers: Principles, Policies and Frameworks; Information; Culture Ethics and Behaviour

10. Formalize a process to request emergency changes.

- Related threats: T3, T5
- Mitigation—Business users may have last-minute requirements throughout the life cycle of the implementation, even after the assessment, configuration, testing and sign-off stages are completed. A framework should be in place to determine if and how these business requirements can be incorporated, with formal agreement from the business. Considerations to include in the framework are the effect on existing designs and configuration. Risk assessment and impact analysis should also be considered before approving last-minute requirements or implementations.

- Related guidance in COBIT 5:
  - Processes: BAI05 *Manage organisational change enablement*, BAI06 *Manage changes*
  - Enabler: Information; Principles, Policies and Frameworks

11. Implement segregation of duties.

- Related threats: T3, T4, T5
- Mitigation—Roles and responsibilities are documented, agreed on and communicated to resources involved in the respective activities, in concurrence with management. This ensures that conflicting responsibilities are not assigned to the same person. Documenting the roles and responsibilities decreases the risk of unavailability of resources because a detailed plan can specify which people and skills are required and when they are required.
- Related guidance in COBIT 5:
  - Process: APO07 *Manage human resources*
  - Enablers: Principles, Policies and Frameworks; Organisational Structures; People, Skills and Competencies

12. Ensure traceability.

- Related threat: T3
- Mitigation—The CM process has many business requirements that need to be aligned with IT processes. Developing consolidated templates and standard documents from requirement gathering to implementation results in clear testing activities and easier collaboration across teams. Templates to gather requirements, collecting CIs and building relationships to the level required are necessities to improved quality of implementations.
- Related guidance in COBIT 5:
  - Process: BAI02 *Manage requirements definition*
  - Enabler: Information

13. Evaluate compliance with enterprise policies.

- Related threat: T1
- Mitigation—Evaluation of compliance with all applicable laws and regulations is critical to the existence of the enterprise. Consider all aspects of compliance when making decisions during the determination of business requirements phase. These aspects lead to tangible activities within the CM activities, reporting, and configuration of ITSM applications and tools.
- Related guidance in COBIT 5:
  - Process: MEA01 *Monitor, evaluate and assess performance and conformance*
  - Enabler: Principles, Policies and Frameworks

14. Assign a single point of contact.
- Related threats: T1, T3
- Mitigation—During CM implementation, a large number of people are involved, such as CI owners, process analysts, process managers, technical implementation teams and business analysts. Teaming up for this challenging project is essential, but the possibility that individuals will conflict with each other and cause the project progression to slow is present. Creating a cohesive and collaborative understanding across the project, business and team during the CM implementation is a prerequisite for a smooth progression. Assigning a single point of contact (SPOC) to ensure cohesion and collaboration and who can act as a mediator across the CM implementation project is good practice.
- Related guidance in COBIT 5:
  – Enabler: Organisational Structures

15. Plan to have continuous CM training.
- Related threats: T1, T4, T5
- Mitigation—Ensure that sufficient training is organized for both CM professionals and professionals from supporting disciplines. Given the challenge of hiring adequate CM resources, it is important to ensure that the acquired resources receive proper training for the execution of their respective responsibilities. Training enables the CM professionals to maintain and improve their skill set. Professionals from supporting disciplines should also be given training on the importance of CM for their daily jobs, to streamline the expectations about CM and to ensure that all operational staff members know their responsibilities in the CM process.
- Related guidance in COBIT 5:
  – Processes: APO07 *Manage human resources*, BAI08 *Manage knowledge*
  – Enablers: Information; People, Skills and Competencies

Page intentionally left blank

6. Continuous Improvement to Develop a Capable CM Process and Other Enablers

This chapter provides an overview of the COBIT 5 process capability assessment model that can be used to assess CM and determine the existing capability level. Examples of CM-specific enablers are provided for each of the capability levels to increase understanding of the practical applicability of the capability model.

COBIT 5 Process Capability Assessment Based on ISO/IEC 15504

The COBIT 5 product set[17] includes a process capability model based on the internationally recognized ISO/IEC 15504 Software Engineering—Process Assessment standard. This model achieves the same overall objectives of process assessment and process improvement support, i.e., it provides a means to measure the performance of any of the governance (EDM-based [evaluate, direct, monitor]) processes or management (PBRM-based [plan-build-run-monitor]) processes and allows areas for improvement to be identified.

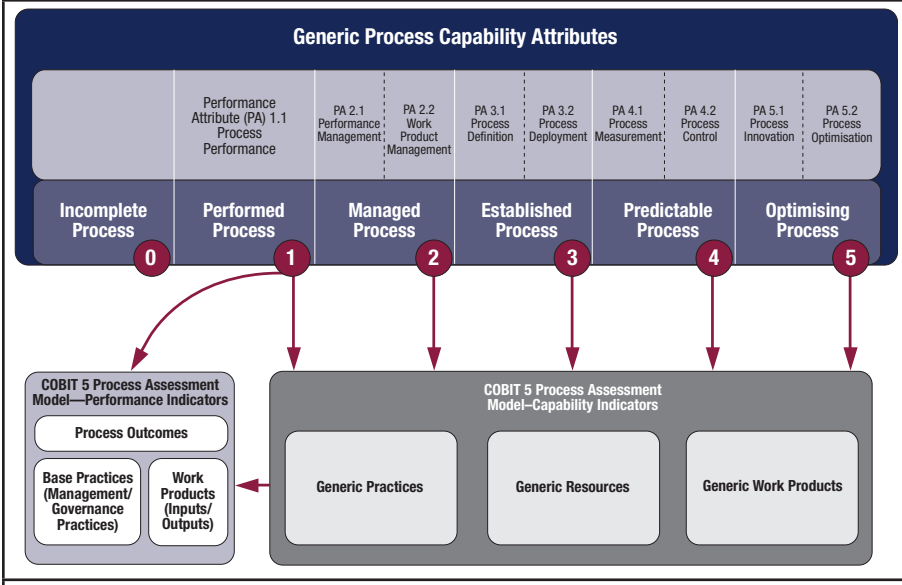
The COBIT 5 ISO/IEC 15504-based assessment approach also facilitates the following objectives:
- Enable the governance body and management to benchmark process capability.
- Enable high-level "as-is" and "to-be" health checks to support the governance body and management investment decision making with regard to the process improvement.
- Provide gap analysis and improvement planning information to support definition of justifiable improvement projects.
- Provide the governance body and management with assessment ratings to measure and monitor current capabilities.

The COBIT 5 process capability model is summarized in **figure 16**.

A process can achieve six different levels of capability, including an "incomplete process" designation if the practice in question does not achieve the intended purpose of the process:
0. **Incomplete process**—The process is not implemented or fails to achieve its process purpose. At this level, there is little or no evidence of any systematic achievement of the process purpose.
1. **Performed process (one attribute)**—The implemented process achieves its process purpose.
2. **Managed process (two attributes)**—The previously described performed process is now implemented in a managed fashion (planned, monitored and adjusted) and its work products are appropriately established, controlled and maintained.
3. **Established process (two attributes)**—The previously described managed process is now implemented using a defined process that is capable of achieving its process outcomes.

[17] See COBIT 5, pp. 42–45.

Figure 16—COBIT 5 Process Capability Model



Source: COBIT 5, figure 19.

4. **Predictable process (two attributes)**—The previously described established process now operates within defined limits to achieve its process outcomes.
5. **Optimising process (two attributes)**—The previously described predictable process is continuously improved to meet relevant current and projected business goals.

Each capability level can be achieved only when the level below has been fully achieved. For example, a process capability level 3 (established process) requires the process definition and process deployment attributes to be largely achieved, on top of full achievement of the attributes for a process capability level 2 (managed process).

A significant distinction exists between process capability level 1 and the higher capability levels. Process capability level 1 achievement requires the process performance attribute to be largely achieved, which means that the process is being successfully performed and the required outcomes are obtained by the enterprise. The higher capability levels then add different attributes to the achieved process performance attribute. In this assessment scheme, achieving a capability level 1, even on a scale to 5, is an important achievement for an enterprise. Note that each individual enterprise chooses (based on cost-benefit and feasibility reasons) its target or desired level, which very seldom is one of the highest levels.

COBIT 5 Process Capability Assessment for CM

Based on the COBIT 5 process capability model, this section provides some examples of how some of the primary elements of the enablers will look at each of

the capability levels of the model. These examples provide practical guidance on assessing the process capability level of the CM process:

1. Performed process—The first level focuses on the process achieving its goals. Although defining target capability levels is up to each enterprise to decide, many enterprises have the ambition to have all their processes achieve capability level 1. Assessing whether the process achieves its goals—in other words, achieves capability level 1—can be done by verifying some basic steps of the CM process:

- Are all CIs created as requested by the business users, with the proper attributes?
- Are the relationships between CIs established?
- Are all requested changes to the CIs assessed, processed and tracked?
- Are baselines being set at regular points in time?
- Are all status reports being produced, including the required details, according to the business request?
- Are compliance and audit checks performed on a regular basis?
- Any other outcomes that the enterprise might be expecting from the CM process

When reviewing these process outcomes, not every outcome will be fully achieved. To have some sort of classification, an ISO/IEC 15504 rating scale can be used to assign a rating to which degree each objective is achieved. This scale consists of the following ratings:

- **N (not achieved)**—There is little or no evidence of achievement of the defined attribute in the assessed process. (0 to 15 percent achievement)
- **P (partially achieved)**—There is some evidence of an approach to, and some achievement of, the defined attribute in the assessed process. Some aspects of achievement of the attribute may be unpredictable. (15 to 50 percent achievement)
- **L (largely achieved)**—There is evidence of a systematic approach to, and significant achievement of, the defined attribute in the assessed process. Some weakness related to this attribute may exist in the assessed process. (50 to 85 percent achievement)
- **F (fully achieved)**—There is evidence of a complete and systematic approach to, and full achievement of, the defined attribute in the assessed process. No significant weaknesses related to this attribute exist in the assessed process. (85 to 100 percent achievement)

In addition, the process (governance or management) practices can be assessed using the same rating scale, expressing the extent to which the base practices are applied. If this first capability level is not achieved, the reasons for not achieving this level are immediately obvious from the approach explained previously, and an improvement plan can be defined:

- If a required process outcome is not consistently achieved, the process does not meet its objective and needs to be improved.

The assessment of the process practices reveals which practices are lacking or failing, enabling implementation and/or improvement of those practices to take place and allowing all process outcomes to be achieved.

2. Managed process—The following examples of the COBIT 5 enablers can be expected at a "managed" capability level, covering the process attributes PA 2.1 Performance Management and PA2.2 Work Product Management:

- **Organisational structures**—A structured organization of different CM professionals is set up, each with their own defined accountabilities and responsibilities, to plan all the CM activities, monitor the performance of the activities and make changes where necessary to meet the plan.
- **Information:**
  - A CM governance model and strategy is defined to coordinate and guide the execution of all the CM activities.
  - The CIs in the CMS have a fixed and uniform structure defined. There is a formal control mechanism set up to ensure that all new input will comply with the structure set forward and there is a regular verification of compliance of the CIs with the structure.
  - Performance management of the process is conducted by using KPIs and metrics to measure the performance. Specific KPIs and metrics for CM are described in the following section on example KPIs and metrics for performance assessment.
- **Services, infrastructure and applications**—The CMS is designed in such a way that the uniform structure of the information CIs is enforced as much as possible.
- **People, skills and competencies**—The structured organization consists of CM professionals, each with their specific skills and competencies aligned to the role and activities they execute.

3. Established process—The following examples of the COBIT 5 enablers can be expected at an "established" capability level, covering the process attributes PA 3.1 Process Definition and PA3.2 Process Deployment:

- **Principles, policies and frameworks**—A defined CM process model is established to enable the execution of the process in a repeatable manner by any CM professional in such a way that it will always achieve its process outcomes.
- **Processes**—The sequence and interaction with other disciplines is determined.
- **Information**—Data are being collected and analyzed as a basis for understanding the behavior of the process and to demonstrate the suitability and effectiveness of the process, and to evaluate where continuous improvement of the process can be made.
- **Services, infrastructure and applications**—The supporting infrastructure and applications are identified, deployed and maintained. A long-term strategy, aligned with the organizational goals, is established to determine the level of automation of the CM process.
- **People, skills and competencies**—The required competencies and roles to perform the defined CM process are identified and they have appropriate education, training and experience.

4. Predictable process—The following examples of the COBIT 5 enablers can be expected at a "predictable" capability level, covering the process attributes PA 4.1 Process Measurement and PA4.2 Process Control:

- **Processes:**
  - Quantitative objectives for the CM process performance, in support of business goals, are established.

- Control limits of variation are established for normal CM process performance.
- Measures and frequency of measures are identified and defined in line with the CM process measurement needs.
- Results of measures are collected, analyzed and reported.
- Corrective action is taken to address variation.
- Control limits are reestablished following corrective action.

• **Information**—CM process information needs are established in support of the business goals. The process measurement needs are derived from the information needs.

5. Optimising process—The following examples of the COBIT 5 enablers can be expected at an "optimising" capability level, covering the process attributes PA 5.1 Process Innovation and PA5.2 Process Optimisation:

• **Processes**—The CM process model is regularly assessed in terms of suitability in accordance to the (changed) business environment and needs.

• **Organisational structures**—The structured organization of different CM professionals is regularly assessed in terms of suitability in accordance with the (changed) business environment and needs.

• **Information**—Continuous improvement is introduced into the CM process in the sense that the CIs and their attributes are regularly assessed on their relevance to the business needs and updated, if required.

Example KPIs and Metrics for Performance Assessment

To assess the performance of a process, as described in the previous section on managed process, the following KPIs and metrics can be used in two categories:

• **CM process controls:**
  - Number of deviations between designated element configuration managers and actual infrastructure
  - Number of discrepancies relating to incomplete or missing configuration information
  - Minimization of redundant collection of data, through the use of established common repositories (CMDBs)
  - Percentage of CIs attribute errors found in the CMS
  - Percentage of number of CIs successfully audited
  - Percentage of accurately configured data determined during audits
  - Percentage of variance determined during audits
  - Number of unauthorized configurations

• **Support, integration and interfacing to all other ITSM processes:**
  - Reduced percentage of change failures as a result of inaccurate configuration data
  - Improved incident resolution time due to the availability of complete and accurate configuration data
  - Number of incidents from failed changes due inaccurate configuration data
  - Number of breached SLAs due to CMS errors or problems
  - Number of IT-related noncompliance issues reported to the board or causing public comment or embarrassment
  - Number of noncompliance issues relating to contractual agreements with IT service providers

- Coverage of compliance assessments
- Frequency of capability maturity and cost optimization assessments
- Trend of assessment results
- Level of business user satisfaction with quality and timeliness (or availability) of management information
- Ratio and extent of erroneous business decisions where erroneous or unavailable information was a key factor

7. High-level Mapping of COBIT 5 and ITIL V3 for CM

Figure 17—High-level Mapping of COBIT 5 and ITIL V3 for CM	
COBIT 5	**ITIL V3**
APO02 Manage Strategy	Service Strategy, 4.1 Strategy Management for IT Services
APO04 Manage Innovation	
APO06 Manage Budget and Costs	Service Strategy, 4.3 Financial Management of IT Services
APO07 Manage Human Resources	
APO09 Manage Service Agreements	• Service Strategy, 4.4 Demand Management • Service Strategy, 4.2 Service Portfolio Management • Service Design, 4.2 Service Catalogue Management • Service Design, 4.3 Service Level Management
APO11 Manage Quality	
APO12 Manage Risk	
BAI01 Manage Programmes and Projects	
BAI02 Manage Requirements Definition	Service Design, 4.1 Design Coordination
BAI03 Manage Solutions Identification and Build	
BAI05 Manage Organisational Change Enablement	
BAI06 Manage Changes	Service Transition, 4.2 Change Management
BAI07 Manage Change Acceptance and Transitioning	• Service Transition, 4.1 Transition Planning and Support • Service Transition, 4.4 Release and Deployment Management • Service Transition, 4.5 Service Validation and Testing • Service Transition, 4.6 Change Evaluation
BAI08 Manage Knowledge	Service Transition, 4.7 Knowledge Management
BAI09 Manage Assets	Service Transition, 4.3 Service Asset and Configuration Management
BAI10 Manage Configuration	**Service Transition, 4.3 Service Asset and Configuration Management**
DSS02 Manage Service Requests and Incidents	Service Operation, 4.2 Incident Management Service Operation, 4.3 Request Fulfillment
DSS03 Manage Problems	Service Operation, 4.4 Problem Management
DSS04 Manage Continuity	Service Design, 4.6 IT Service Continuity Management
DSS05 Manage Security Services	Service Operation, 4.5 Access Management
DSS06 Manage Business Process Controls	
MEA01 Monitor, Evaluate and Assess Performance and Conformance	**Continual Service Improvement, 4.1 The 7-Step Improvement Process**
MEA02 Monitor, Evaluate and Assess the System of Internal Control	
MEA03 Monitor, Evaluate and Assess Compliance With External Requirements	

Page intentionally left blank

8. Glossary

Availability—Ensuring timely and reliable access to and use of information

Baseline architecture—The existing description of the fundamental underlying design of the components of the business system before entering a cycle of architecture review and redesign. Scope Note: COBIT 5 perspective.

Bug tracking system—A software application designed to help keep track of reported software bugs during software development effort

Business process owner—The individual responsible for identifying process requirements, approving process design and managing process performance. Scope Note: Must be at an appropriately high level in the enterprise and have authority to commit resources to process-specific risk management activities

Change—Any planned or unplanned modification to system components

Change management—A holistic and proactive approach to managing the transition from a current to a desired organizational state, focusing specifically on the critical human or "soft" elements of change. Scope Note: Includes activities such as culture change (values, beliefs and attitudes), development of reward systems (measures and appropriate incentives), organizational design, stakeholder management, human resources (HR) policies and procedures, executive coaching, change leadership training, team building and communication planning and execution

Change advisory board (CAB)—Group of stakeholders that delivers support to the change management team by approving requested changes and assisting in the assessment and prioritization of changes. This body is generally made up of IT and business representatives that include: the change manager, user managers and groups, technical experts, possible third parties and customers (if required).

COBIT—A complete, internationally accepted framework for governing and managing enterprise information and technology (IT) that supports enterprise executives and management in their definition and achievement of business goals and related IT goals. COBIT describes five principles and seven enablers that support enterprises in the development, implementation, and continuous improvement and monitoring of good IT-related governance and management practices.

Configuration item (CI)—ITIL V3: Any component that needs to be managed to deliver an IT service. Information about each CI is recorded in a configuration record within the configuration management system and is maintained throughout its life cycle by configuration management. CIs are under the control of change management. CIs typically include IT services, hardware, software, buildings, people and formal documentation such as process documentation and SLAs.

CI class—A class is a technical term indicating one item of a data model. The item can have multiple components that each have a specific state and behavior.

Configuration control board (CCB)—Group of project stakeholders responsible for evaluating and approving or disapproving proposed changes to a system, prioritizing the incorporation of approved changes, scheduling the changes for forthcoming releases. In some projects the CCB may also be responsible for verifying that approved changes are implemented.

Configuration management (CM)—The control of changes to a set of configuration items over a system life cycle

Configuration management database (CMDB)—A database used to store configuration records throughout their life cycle. The configuration management system maintains one or more CMDBs, and each CMDB stores attributes of the configuration record. A configuration record consists of configuration items (CIs) and its relationship with other configuration items. All configuration items have configurable attributes.

Configuration management system (CMS)—ITIL V3: 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.

Continuity management—Process by which plans are put in place and managed to ensure that IT services can recover and continue should a serious incident occur. It is not just about reactive measures, but also about proactive measures—reducing the risk of a disaster in the first instance.

Criticality—The quality, state or degree of being of the highest importance

Data elements—A basic unit of information built on standard structures having a unique meaning and distinct units or values

Data model—Specification describing how a database is structured and used, by providing graphic notations which document entities and their relationships, and the constraints that bind them

Definitive media library (DML)—The secure library in which the definitive authorized versions of all media CIs are stored and protected. It stores master copies of versions that have passed quality assurance checks. This library may consist of one or more software libraries or file-storage areas, separate from development, test or live file-storage areas. It contains the master copies of all controlled software in an organization. The DML should include definitive copies of purchased software (along with license documents or information), as well as software developed on site. Master copies of controlled documentation for a system are also stored in the DML in electronic form.

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

Definitive spares (DS)—Physical storage of all spare IT components and assemblies maintained at the same level as within the live environment. New IT assemblies are stored here until ready for use, and additional components can be used when needed for additional systems or in the recovery from incidents.

Enablers—Factors that, individually and collectively, influence whether something will work

Enterprise architecture—Description of the fundamental underlying design of the components of the business system, or of one element of the business system (e.g., technology), the relationships among them, and the manner in which they support the enterprise's objectives

Gap analysis—Comparison of actual performance with potential performance

HIPAA—Acronym for the Health Insurance Portability and Accountability Act that was passed by the United States Congress in 1996 to ensure the confidentiality and security of protected health information

IEC—International Electrotechnical Commission

Incident—Any event that is not part of the standard operation of a service and that causes, or may cause, an interruption to, or a reduction in, the quality of that service

ISO—International Organization for Standardization

ISO/IEC 15504—Defines the requirements for performing process assessment as a basis for use in process improvement and capability determination

ISO/IEC 20000-1:2011—Service management system (SMS) standard that specifies requirements for the service provider to plan, establish, implement, operate, monitor, review, maintain and improve an SMS. The requirements include the design, transition, delivery and improvement of services to fulfill agreed service requirements.

ITIL—The UK Cabinet Office IT Infrastructure Library. A set of guides on the management and provision of operational IT services.

IT service management (ITSM)—Process-based practice intended to align the delivery of IT services with needs of the enterprise, emphasizing benefits to customers. ITSM involves a paradigm shift from managing IT as stacks of individual components to focusing on the delivery of end-to-end services using best practice process models.

Key performance indicator (KPI)—A measure that determines how well the process is performing in enabling the goal to be reached. Scope Note: A lead indicator of whether a goal will likely be reached and a good indicator of capabilities, practices and skills. It measures an activity goal, which is an action that the process owner must take to achieve effective process performance.

Logical partition (LPAR)—Subset of hardware resources used to create a virtual environment (processor, memory, storage). One physical machine can be divided into multiple logical partitions.

Operating level agreement (OLA)—An internal agreement covering the delivery of services that support the IT organization in its delivery of services

Performance management—In IT, the ability to manage any type of measurement, including employee, team, process, operational or financial measurements. The term connotes closed-loop control and regular monitoring of the measurement.

Payment card industry data security standard (PCI DSS)—Set of requirements designed to ensure that all companies that process, store or transmit credit card information maintain a secure environment

Problem—In IT, the unknown underlying cause of one or more incidents

Process capability—ISO/IEC 15504: A characterization of the ability of a process to meet current or projected business goals

Process capability assessment—Model based on the internationally recognized ISO/IEC 15504 Software Engineering—Process Assessment standard. It provides the means to measure the performance of any of the governance (EDM-based) processes or management (PBRM-based) processes, and allows for areas of improvement to be identified.

Project library—Contains the minutes, reports, current baselined projects, approved documentation, and other configuration management artifact files to keep the project running smoothly

RACI chart—Matrix chart that illustrates who is Responsible, Accountable, Consulted and Informed within an organizational framework

Risk—The combination of the probability of an event and its consequence (ISO/IEC 73)

Return on investment (ROI)—A measure of operating performance and efficiency, computed in its simplest form by dividing net income by the total investment over the period being considered

System development life cycle (SDLC)—The phases deployed in the development or acquisition of a software system. Scope Note: SDLC is an approach used to plan, design, develop, test and implement an application system or a major modification to an application system. Typical phases of SDLC include the feasibility study, requirements study, requirements definition, detailed design, programming, testing, installation and postimplementation review, but not the service delivery or benefits realization activities.

Service knowledge management system (SKMS)—ITIL V3: Is a combination of tools and databases that are used to manage information and knowledge about services

Service level agreement (SLA)—An agreement, preferably documented, between a service provider and the customer(s)/user(s) that defines minimum performance targets for a service and how they will be measured

Single point of contact (SPOC)—An individual representing a group of people or enterprises

SSAE16—Statement on Standards for Attestation Engagements 16, is a regulation created by the Auditing Standards Board of the American Institute of Certified Public Accountants (AICPA) for redefining and updating how service companies report on compliance controls

Stakeholder—Anyone who has a responsibility for, an expectation from or some other interest in the enterprise

Underpinning contract (UC)—A contract between an IT service provider and a third party. The third party provides goods or services that support delivery of an IT service to a customer. The UC defines targets and responsibilities that are required to meet agreed service level targets in an SLA (ITIL V3) virtual environment.

Page intentionally left blank

9. References

Betz, Charles T.; "The High Value CMDB," EMA, 14 June 2012, *www.brighttalk.com*
ISACA, COBIT 5, USA, 2012
ISACA, *COBIT 5: Enabling Processes*, USA, 2012
ISACA, *COBIT 5 for Information Security*, USA, 2012
ISACA, *COBIT 5 for Assurance*, USA, 2013
ITIL V3—Service Strategy, Cabinet Office, UK, 2011
ITIL V3—Service Design, Cabinet Office, UK, 2011
ITIL V3—Service Transition, Cabinet Office, UK, 2011
ITIL V3—Service Operation, Cabinet Office, UK, 2011
ITIL V3—Continual Service Improvement, Cabinet Office, UK, 2011

Page intentionally left blank

Appendix A. Implementation Project Plan Example

Figure 18 contains an example implementation project plan that provides an overview of the most important steps during the implementation project. The example is designed to be used as a tool so that each step can be reviewed and a notification tick mark can be entered to indicate if the step is relevant and applied in the enterprise.

Figure 18—Implementation Project Plan Example

Task Name	Relevant?	Applied?
Plan phase (ITSM CM CMDB)		
Perform workshops.		
Modify enterprise-specific CM process.		
Modify CM templates.		
Publish enterprise-specific CM process.		
Perform assessment data collection—tools.		
Plan phase (CMDB)		
Assess common data design model (CMDB).		
Define enterprise-specific common data design model (CMDB).		
Assess CI location/people/groups, relationships (CMDB).		
Manage configuration coordinator relationships.		
Support group relationships.		
Use people relationships.		
Document enterprise-specific common data design model (CMDB).		
Document gap analysis.		
Review gap analysis (CMDB).		
Document action items to resolve gaps (CMDB).		
Assess integrations (CMDB).		
Assess data infrastructure (CMDB).		
Assess data requirements and repository for CIs (CMDB).		
Assess required CMDB integration point.		
Assess integration tools and data requirements.		
Assess foundation CI and topology discovery integrations.		
Obtain client sign-off on gap analysis (CMDB).		

Figure 18—Implementation Project Plan Example *(cont.)*		
Task Name	**Relevant?**	**Applied?**
Plan phase—CM		
Perform workshops.		
Perform workshops on CM model.		
Modify enterprise-specific CM.		
Create and modify configuration control board process/governance.		
Publish enterprise-specific change management process.		
Assess client approval requirements for CM.		
Assess service life cycle management (SLM) alignment with business.		
Assess enterprise requirements for existing CM processes.		
Assess CMDB/CI requirements.		
Assess roles, responsibilities and ITSM tool permission.		
Assess CCB organization and membership.		
Gather/assess support group requirements.		
Assess reporting requirements.		
Design phase		
Design process alignment.		
Design required process modifications.		
Review and sign-off support groups data.		
Document CCB organization, membership and agenda.		
Design approval mappings.		
Deploy phase		
Deploy tools solution.		
Configure integration requirement points.		
Configure federation integration.		
Configure foundation CIs and topology discovery.		
Test integration configuration.		
Test user permissions.		
CM deploy phase.		
Configure approval mappings.		
Document test cases.		
Conduct process testing.		
Manual CI testing.		
Review test results.		
Sign off.		

Appendix B. Audit Checklist Examples

This section provides two examples, one for a functional configuration audit (FCA) and one for a physical configuration audit (PCA).

Specific audit guidance and audit/assurance programs can be found in *COBIT 5 for Assurance*. This publication contains an extensive and detailed audit/assurance program, allowing the reader to follow a structured approach to define the audit scope and audit all enablers related to the scope at hand.

The examples in **figures 19** and **20** can be used as extra guidance during the execution of the assurance steps of the *COBIT 5 for Assurance* audit/assurance program, to make sure all important aspects of a CM audit are covered.

Figure 19—Functional Configuration Audit Checklist Example

No.	Item	Yes/No/NA
1.	Are all requirements documents available and current?	
2.	Are design documents available and updated with the latest requirements?	
3.	Are the test plan and results available?	
4.	Are all identified requirements allocated to software components?	
5.	Does the configuration described in the design document match the physical configuration of the software?	
6.	Are all identified requirements allocated to test cases?	
7.	Is the test and analysis data available and are the variances to the test results identified and updated?	
8.	In the provided test/analysis data, are all the requirements met?	
9.	If all requirements are not met, are deficiencies identified?	
10.	Is impact analysis performed for all change requests?	
11.	Has change implementation been approved by appropriate authorities?	
12.	During release, do the configuration items used match with the latest version indicated in baseline record?	
13.	Are user sign-off documents available?	

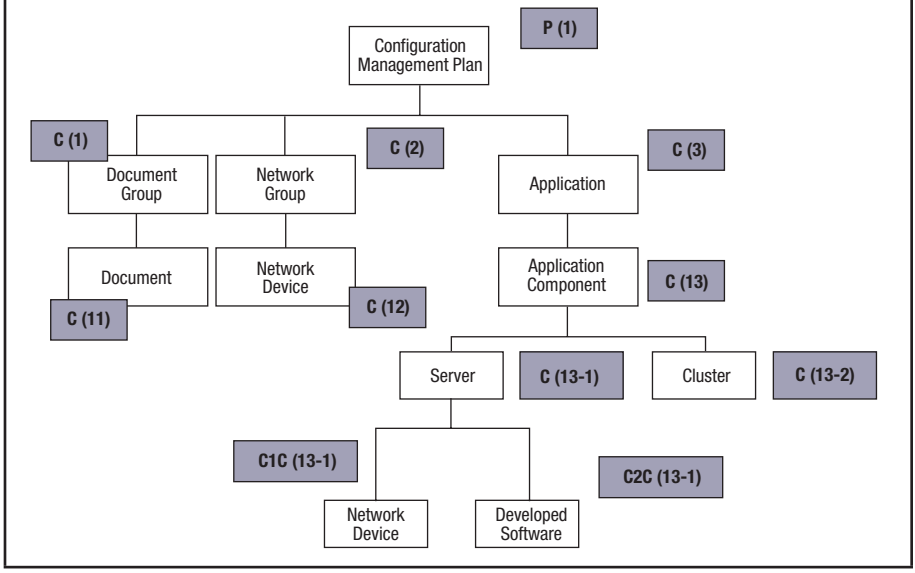
Figure 20—Physical Configuration Audit Checklist Example		
No.	**Item**	**Yes/No/NA**
1.	Are all requirements documents available and current?	
2.	Are design documents available and updated with the latest requirements?	
3.	Do design documents match inventory reports?	
4.	Are the objects baselined in the CM repository?	
5.	Does the source-code inventory report list the products to be delivered to the customer?	
6.	If the answer to 5 is yes, does the list of products on the inventory list match the products on the delivery medium (or installed from the delivery medium)?	
7.	Have the CIs and the exact version or revision that constitutes a particular baseline (documents, source code, problem reports and change documents) been identified?	
8.	If the answer to 7 is yes, do the identified products match the products on (or installed from) the delivery medium?	
9.	If the answer to 7 is yes, do the identified products match the list of products in the inventory report?	

Appendix C. Data Design Template for a Data Design Model

Figure 21 is an example of how to build up the relationship levels of a data design model in a structured way. The example in **figure 21** pertains to **figure 22**.

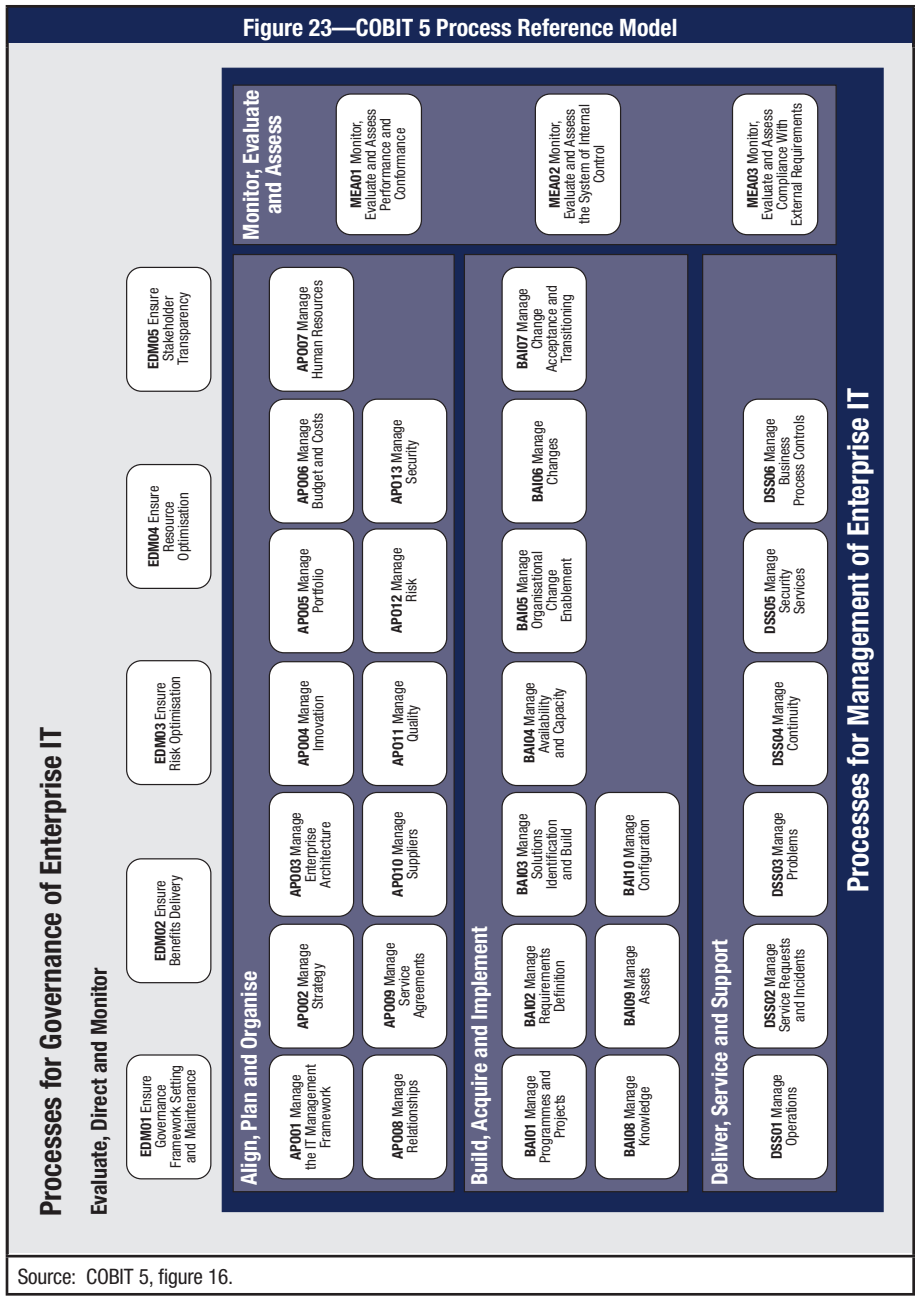
Figure 21—Data Design Model Template					
Relationship Level 1	**Relationship Level 2**	**Relationship Level 3**	**Relationship Level 4**	**CI Class**	**CI Names**
Parent (P1)				CM Plan	
Child (C1)				Document group	
	Child (C11)			Document	
Child (C2)				Network group	
	Child (C12)			Network device	
Child (C3)				Application	
	Child (C13)			Application component	
		Child (C13-1)		Server	
		Peer (C13-2)		Cluster	
			Child (C1C13-1)	Network device	
			Peer (C1C13-1)	Developed software	
			Child (C2C13-1)	Developed software	
			Peer (C2C13-1)	Network device	
		Child (C13-2)		Cluster	
		Peer (C13-1)		Server	

Figure 22—Data Design Model Example



Appendix D. COBIT 5 Processes for Governance of Enterprise IT

Figure 23—COBIT 5 Process Reference Model

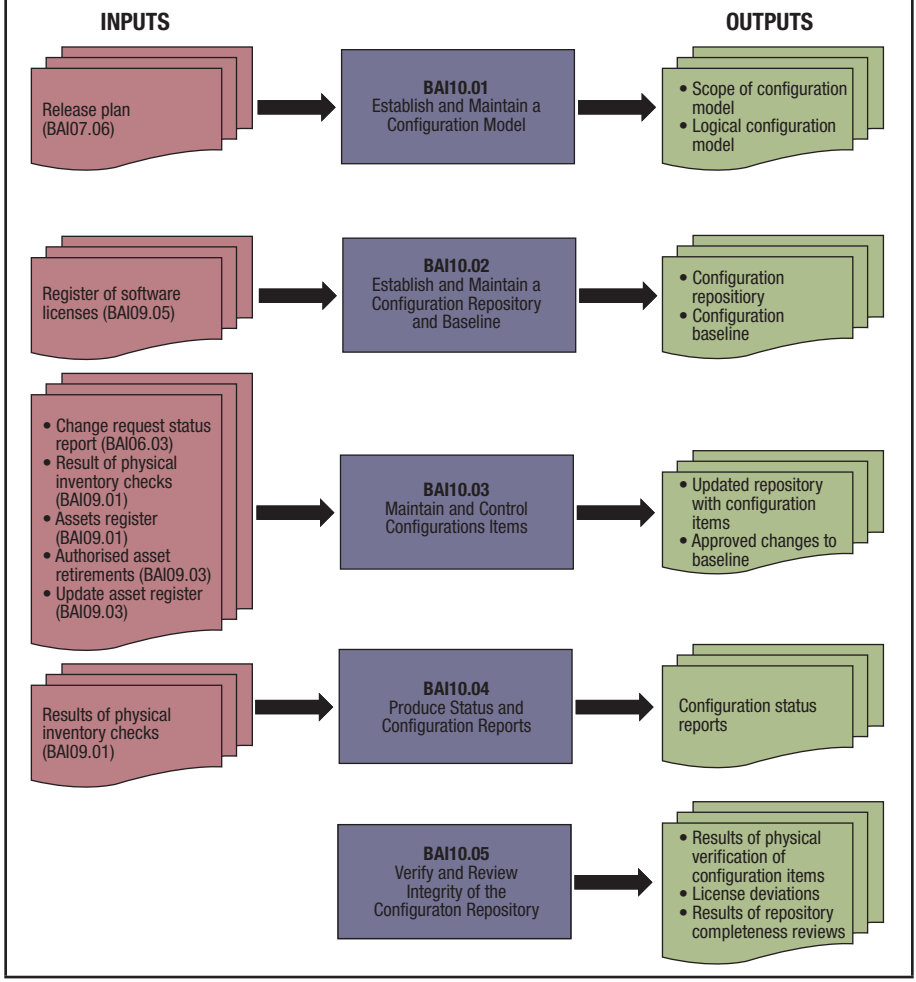


Source: COBIT 5, figure 16.

Page intentionally left blank

Appendix E. BAI10 Manage Configuration Inputs and Outputs

Figure 24—BAI10 Manage Configuration Inputs and Outputs



Page intentionally left blank

Appendix F. Mapping Threats and Mitigating Actions Using COBIT 5

Figure 25—CM Threats and Mitigating Actions Using COBIT 5		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T1. Confusion on definition/scope/meaning **T2.** Immature guidance on CM **T3.** Implementation challenges for CM **T4.** CM resource management **T5.** Uncontrolled maintenance of CIs	**1. Define a clear strategy for CM in the enterprise.** Mitigation—Defining a strategy is an action that upper management should take to help the enterprise and its various organizations meet business goals. This strategy defines the directions and plans that should guide business and IT performance to help them meet the business objectives established for them. A vision, mission statements, goals, objectives and benefits of CM, as defined by management, set the strategic direction for this particular process and related activities. These statements and a defined enterprise architecture provide a strong foundation for the implementation and maintenance of the CM process. The next step in establishing a strategy is to transform statements into activities, tasks, roles and responsibilities, project standards, reports, etc., for operational actions.	• **Process:** APO02 *Manage strategy* • **Enabler:** Principles, Policies and Frameworks

Figure 25—CM Threats and Mitigating Actions Using COBIT 5 *(cont.)*		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T1. Confusion on definition/scope/meaning **T2.** Immature guidance on CM **T3.** Implementation challenges for CM	**2. Link CM to the enterprise overall governance model.** Mitigation—Include CM in the enterprise governance model. The governance model should define the CM stakeholders and their responsibilities, escalation procedures, reporting requirements and frequency, governance bodies, etc.	• **Processes:** EDM01 *Ensure governance framework setting and maintenance*, BAI10 *Manage configuration* • **Enabler:** Organisational Structures
T2. Immature guidance on CM **T3.** Implementation challenges for CM **T5.** Uncontrolled maintenance of CIs	**3. Define the CM process before automating.** Mitigation—After establishing a clear strategy and governance model, it is advisable to take some time to determine the level of automation that is required to operate the envisioned CM process properly. Too many enterprises tend to start immediately with a full automation of the process, although full automation is not necessary and creates more problems than solutions. Think about which parts of the CM process need to be automated and which parts are unnecessary to automate.	• **Processes:** BAI01 *Manage programmes and projects*, BAI02 *Manage requirements definition*, BAI03 *Manage solutions identification and build*, BAI10 *Manage configuration* • **Enablers:** Principles, Policies and Frameworks; Processes; Information; Services, Infrastructure and Applications; People, Skills and Competencies
T1. Confusion on definition/scope/meaning **T2.** Immature guidance on CM **T5.** Uncontrolled maintenance of CIs	**4. Establish policies and procedures for CM.** Mitigation—Standardized procedures and guidelines to build processes and tool sets in which risk is managed and decisions are made on an objective basis by stakeholders. Policies and procedures are the tools to establish this working environment and can support all CM-related activities.	• **Enabler:** Principles, Policies and Frameworks

Figure 25—CM Threats and Mitigating Actions Using COBIT 5 *(cont.)*		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T1. Confusion on definition/scope/meaning **T3.** Implementation challenges for CM	**5. Formulate clear requirements.** Mitigation—The requirements definition forms the basis for establishing the CM model, CI life cycle, deliverables, quality of services, cost, etc. Poorly defined requirements may cause low quality of services and may also lead to financial impact due to scope creep. A prerequisite is that business process owners, IT, legal and other stakeholders must reach a common understanding of the requirements before proceeding to the next step.	• **Processes:** APO06 *Manage budget and costs*, APO11 *Manage quality*, APO12 *Manage risk*, BAI02 *Manage requirements definition* • **Enabler:** Information
T3. Implementation challenges for CM **T4.** CM resource management	**6. Perform an adequate analysis on products and tools selection** Mitigation—An intensive analysis to align the business requirements to the tools available in the market is an important step for the successful implementation and a better ROI of CM. The assessment is carried out based on various parameters, such as the level of automation, functional requirements from a business perspective, data alignment at a high level, security requirements of the business, time to deploy, cost, etc. A decision can be taken through consultations with independent organizations providing similar services. Previous success rate through a references approach can also be considered.	• **Processes:** APO04 *Manage innovation*, BAI03 *Manage solutions identification and build* • **Enabler:** Services, Infrastructure and Applications

Figure 25—CM Threats and Mitigating Actions Using COBIT 5 *(cont.)*		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T2. Immature guidance on CM **T3.** Implementation challenges for CM	**7. Apply the standard project management activities to CM projects.** Mitigation—Manage CM projects from the investment portfolio in alignment with the enterprise strategy and in the same coordinated way as other projects. Initiate, plan, control and execute CM projects and close with a postimplementation review. Best practices are: • Maintain a standard approach for the CM project • Manage stakeholder engagement • Develop and maintain the CM project plan • Launch and execute the project • Monitor, control and report on the project outcomes • Manage project quality • Manage project risk • Manage project resources and work packages • Close project with review postimplementation	• **Processes:** APO06 *Manage budget and costs*, APO07 *Manage human resources*, APO11 *Manage quality, APO12 Manage risk*, BAI01 *Manage programmes and projects*, BAI07 *Manage change acceptance and transitioning* • **Enablers:** All
T3. Implementation challenges for CM	**8. Plan reviews, reviews and more reviews during implementation.** Mitigation—For implementation, considerable data are required to make decisions on scope, level of defining CIs, scope of CIs, relationships, etc. At any level, a small error in the system can be disastrous. Small details must be reviewed periodically to ensure quality and consistency. Reviewing up front saves considerable time and effort in a later stage, when searching for and correcting errors can have a greater effect on time and cost.	• **Processes:** BAI01 *Manage programmes and projects*, APO11 *Manage quality* • **Enabler:** Information

Figure 25—CM Threats and Mitigating Actions Using COBIT 5 *(cont.)*		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T3. Implementation challenges for CM **T5.** Uncontrolled maintenance of CIs	**9. Manage changes to CM in a controlled manner.** Mitigation—Manage all changes to the CM project and postimplementation operational activities in a controlled manner. This mitigating action pertains to any change relating to business processes, applications and infrastructure. The "controlled manner" implies the following: • Using change standards and procedures • Performing an impact assessment • Prioritization and authorization • Tracking • Testing • Reporting • Closure • Documentation updates	• **Processes:** BAI05 *Manage organisational change enablement*, BAI06 *Manage changes* • **Enablers:** Principles, Policies and Frameworks; Information; Culture Ethics and Behaviour
T3. Implementation challenges for CM **T5.** Uncontrolled maintenance of CIs	**10. Formalize a process to request emergency changes.** Mitigation—Business users may have last-minute requirements throughout the life cycle of the implementation, even after the assessment, configuration, testing and sign-off stages are completed. A framework should be in place to determine if and how these business requirements can be incorporated, with formal agreement from the business. Considerations to include in the framework are the effect on existing designs and configuration. Risk assessment and impact analysis should also be considered before approving last-minute requirements or implementations.	• **Processes:** BAI05 *Manage organisational change enablement*, BAI06 *Manage changes* • **Enabler:** Information; Principles, Policies and Frameworks

Figure 25—CM Threats and Mitigating Actions Using COBIT 5 *(cont.)*		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T3. Implementation challenges for CM **T4.** CM resource management **T5.** Uncontrolled maintenance of CIs	**11. Implement segregation of duties.** Mitigation—Roles and responsibilities are documented, agreed on and communicated to resources involved in the respective activities, in concurrence with management. This ensures that conflicting responsibilities are not assigned to the same person. Documenting the roles and responsibilities decreases the risk of unavailability of resources because a detailed plan can specify which people and skills are required and when they are required.	• **Process:** APO07 *Manage human resources* • **Enablers:** Principles, Policies and Frameworks; Organisational Structures; People, Skills and Competencies
T3. Implementation challenges for CM	**12. Ensure traceability.** Mitigation—The CM process has many business requirements that need to be aligned with IT processes. Developing consolidated templates and standard documents from requirement gathering to implementation results in clear testing activities and easier collaboration across teams. Templates to gather requirements, collecting CIs and building relationships to the level required are necessities to improved quality of implementations.	• **Process:** BAI02 *Manage requirements definition* • **Enabler:** Information

Figure 25—CM Threats and Mitigating Actions Using COBIT 5 *(cont.)*		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T1. Confusion on definition/scope/meaning	**13. Evaluate compliance with enterprise policies.** Mitigation—Evaluation of compliance with all applicable laws and regulations is critical to the existence of the enterprise. Consider all aspects of compliance when making decisions during the determination of business requirements phase. These aspects lead to tangible activities within the CM activities, reporting, and configuration of ITSM applications and tools.	• **Process:** MEA01 *Monitor, evaluate and assess performance and conformance* • **Enabler:** Principles, Policies and Frameworks
T1. Confusion on definition/scope/meaning **T3.** Implementation challenges for CM	**14. Assign a single point of contact.** Mitigation—During CM implementation, a large number of people are involved, such as CI owners, process analysts, process managers, technical implementation teams and business analysts. Teaming up for this challenging project is essential, but the possibility that individuals will conflict with each other and cause the project progression to slow is present. Creating a cohesive and collaborative understanding across the project, business and team during the CM implementation is a prerequisite for a smooth progression. Assigning a single point of contact (SPOC) to ensure cohesion and collaboration and who can act as a mediator across the CM implementation project is good practice.	• **Enabler:** Organisational Structures

Figure 25—CM Threats and Mitigating Actions Using COBIT 5 *(cont.)*		
Related Threats	**Mitigating Actions**	**Related Guidance in COBIT 5**
T1. Confusion on definition/scope/meaning **T4.** CM resource management **T5.** Uncontrolled maintenance of CIs	**15. Plan to have continuous CM training.** Mitigation—Ensure that sufficient training is organized for both CM professionals and professionals from supporting disciplines. Given the challenge of hiring adequate CM resources, it is important to ensure that the acquired resources receive proper training for the execution of their respective responsibilities. Training enables the CM professionals to maintain and improve their skill set. Professionals from supporting disciplines should also be given training on the importance of CM for their daily jobs, to streamline the expectations about CM and to ensure that all operational staff members know their responsibilities in the CM process.	• **Processes:** APO07 *Manage human resources*, BAI08 *Manage knowledge* • **Enablers:** Information; People, Skills and Competencies

Appendix G. CM Standards and Certifications

Standards

COBIT 5, Enabling Process BAI10

ITIL Framework—best practice in the provision of IT Service. See *http://www.ogc.gov.uk/index.asp?id=1000364.*

ISO 10007:2003, Quality management systems—Guidelines for configuration management

ISO 20000-1:2005, IT Service Management

GEIA Standard 836-2002, Configuration Management Data Exchange and Interoperability

IEEE 829, Standard for Software Test Documentation

CMMI, CMMI for Development, Version 1.2 Configuration Management

CMII-100E, CMII Standard for Enterprise Configuration Management

ANSI/EIA-649B, Configuration Management

GEIA-HB-649, Configuration Management Handbook

GEIA-859A, Data Management

Certifications

ITIL, *http://www.itil-officialsite.com/*

CMPIC Training & Certification, The Configuration Management Process Improvement Center, *http://cmpic.com/configuration-management-certification.htm*

Configuration Management Training Foundation (Certified International Configuration Manager CICM/Certified International Software Configuration Manager CISCM/Certified Configuration Management Professional CCMP), *http://www.cmtf.com/certification.html*

Institute of Configuration Management, *http://www.icmhq.com/*

APMG, International CMDB Certification, *http://www.apmg-international.com/en/qualifications/cmdb/cmdb.aspx*

Page intentionally left blank