



IEEE Standard for Developing a Software Project Life Cycle Process

IEEE Computer Society

Sponsored by the
Software Engineering Standards Committee

1074TM

IEEE
3 Park Avenue
New York, NY 10016-5997, USA
28 July 2006

IEEE Std 1074TM-2006
(Revision of
IEEE Std 1074-1997)

IEEE Standard for Developing a Software Project Life Cycle Process

Sponsored by the

**Software Engineering Standards Committee
of the
IEEE Computer Society**

Approved 30 March 2006

IEEE-SA Standards Board

Abstract: This standard provides a process for creating a software project life cycle process (SPLCP). It is primarily directed at the process architect for a given software project.

Keywords: software project life cycle, software project life cycle model, software project life cycle process

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2006 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 28 July 2006. Printed in the United States of America.

IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by the Institute of Electrical and Electronics Engineers, Incorporated.

Print: ISBN 0-7381-4956-X SH95532
PDF: ISBN 0-7381-4957-8 SS95532

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied “**AS IS**.”

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, explanation, or interpretation of the IEEE.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854
USA

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

This introduction is not part of IEEE Std 1074-2006, IEEE Standard for Developing a Software Project Life Cycle Process.

Background

IEEE Std 1074 is a standard for the generation of the process governing software development and maintenance for a project. This standard requires selection of a user's software project life cycle model (SPLCM) based on the organization's mission, vision, goals, and resources. It is not intended to define or imply a software project life cycle (SPLC) of its own nor does it presume or suggest any particular SPLCM. This standard describes the individual activities that are to be mapped within the selected model and provides examples of mapping onto typical SPLCMs. However, this standard is not an instructional guide. In addition to providing for the generation of a project process, this standard may also be used to develop organizational processes to support software development and maintenance or to develop special, single-function processes within a project.

This standard applies to the management and support activities that continue throughout the entire project's life cycle as well as all aspects of the software life cycle from concept exploration through retirement.

The activities listed in this standard are not executable processes. They are components of processes and not intended to stand alone. The activities are generic and do not imply a sequential order. They have been administratively grouped for convenience and may be likened to a dictionary where words and meanings are arranged to allow the user to quickly locate a desired activity and its components.

This standard provides activities to be addressed in a software life cycle. It allows the user great flexibility in the manner in which activities are mapped onto the selected model and software project life cycle (SPLC) while preserving a normative standard to which to conform.

Utilization of these activities maximizes the benefits to the user when the use of this standard is initiated early in the software project's life cycle. A project that has proceeded past the initialization phase when this standard is invoked should gradually move into conformance to this standard.

This standard was written for any organization responsible for managing and conducting software projects. It will be useful to project managers, software developers, quality assurance organizations, purchasers, users, and maintainers. It can be used where software is the total system or where software is embedded in a larger system. This standard is also useful for projects that do not span the full software life cycle (i.e., developing a software specification or designing, writing, and verifying software that is based upon specifications developed by another organization, company, or previous project).

This standard allows for continuing harmonization with IEEE/EIA 12207.0 [B3]^a and its successors. The standard may be used to develop the primary and supporting life cycle processes specified in IEEE/EIA 12207. This standard supports the development of organizational standard processes and the selection of a standard, defined project process that is tailored from an organization's set of standard processes (organizational process definition) and integrated project management for each individual project.^b

^a The numbers in brackets correspond to the numbers of the bibliographic references listed in Annex F.

^b Process improvement models that meet these criteria include the SEI CMMI[®] model. This information is given for the convenience of users and does not constitute an endorsement by the IEEE of these models. Equivalent models may be used if they can be shown to lead to the same results.

History

Since this standard's original publication, considerable worldwide attention has been paid to software project life cycle processes (SPLCPs). Use of and comments on IEEE Std 1074-1991/1995/1997, and other quality system and life cycle standards activity, have been carefully considered in preparing this substantive revision of this standard.

The 1995 version was a minor revision to correct specific errors found in the 1991 version.

The 1997 version saw the following changes:

- Activities were rearranged into more logical groupings (called activity groups) such as placing all planning activities into the new Project Planning Activity Group, collecting all project initiation activities, and collecting and expanding all review activities.
- The term *process* as used in earlier versions of this standard was replaced with the term *activity group* to identify collections of activities. Some users of this standard were misinterpreting the collections as actual *processes* and trying to execute them as such. The term *activity group* is intended to eliminate this misconception.
- The importance of risk management led to the addition of a new activity, manage risks.
- The recognition that software can be acquired from other sources, for use in the system being developed, led to the addition of the Software Importation Activity Group.

The following changes are among those included in this current version:

- The focus of the standard was more clearly centered on a single process for a given project.
- The term *compliance* was changed to *conformance* to reflect international standards usage.
- Recognition of the importance of release management led to the addition of release management activities.
- The emerging importance of, and increased attention to, software security led to the addition of two activities: determine security objectives and confirm security accreditation.

Notice to users

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: <http://standards.ieee.org/reading/ieee/updates/errata/index.html>. Users are encouraged to check this URL for errata periodically.

Interpretations

Current interpretations can be accessed at the following URL: <http://standards.ieee.org/reading/ieee/interp/index.html>.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying

patents or patent applications for which a license may be required to implement an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Participants

At the time this standard was completed, the Software Life Cycle Processes Working Group had the following membership:

Keith R. Middleton, *Chair*
Tom E. Starai, *Vice Chair*
John W. Horch, *Editor*

Bar Biszick-Lockwood
Ronald F. Dean, Sr.
Christof Ebert
Carol A. Long

Michael T. McCaffrey
John Napier
Cathie L. Neel

Robert J. Schaaf
Bob Shillato
Jan Van Moll
Charlene C. Walrad

The following individuals also contributed to the development of this standard by attending one meeting or providing comments on one or two drafts:

Dave Bachman
Marisa Burt
Geoffrey Darnton

Peter Hantos
Bernard Homes
Susan K. Land

Denis Meredith
James W. Moore
David J. Schultz

The following members of the balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Edward A. Addy
T. Scott Ankrum
Butch Anton
Ali Al Awazi
Adam J. Bagby
William D. Bartholomew
Edward E. Bartlett
Richard E. Biehl
Bar Biszick-Lockwood
Juris Borzovs
David G. Bowen
Curtis Browne
Dino Butorac
Juan C. Carreon
Norbert N. Carte
Danila Chernetsov
Keith Chow
S. M. Claassen
James P. Conigliaro
Paul R. Croll
Sylvia V. Daiqui
Geoffrey Darnton
Ronald F. Dean, Sr.
Sean Dougherty
Scott P. Duncan
Sourav K. Dutta
Kenneth D. Echeberry
Timothy Ehrler
William Eventoff
Harriet Feldman
Andrew C. Fieldsend
Ronald M. Fluegge
Andre F. Fournier

Michael D. Geipel
Gregg C. Giesler
Colin M. Glanville
John Garth Glynn
Lewis Gray
Randall C. Groves
Scott A. Gudgel
John Harauz
Mark O. Henley
Rutger A. Heunks
Bernard Homes
John W. Horch
Dennis Horwitz
Piotr Karocki
Robert B. Kelsey
Mark J. Knight
Thomas M. Kurihara
George W. Kyle
Susan K. Land
Kenneth Lang, Jr.
J. Dennis Lawrence
David J. Leciston
Solomon Lee
Yeou Song Lee
Daniel G. Levesque
Daniel A. Lindberg
Murray R. Little, Jr.
Carol A. Long
William Lumpkins
Faramarz Maghsoodlou
Richard A. Martin
L. J. Tajerian Martinez
Richard A. McBride

Michael T. McCaffrey
Mark F. McGranaghan
Francisco J. Melendez
Gary L. Michel
Keith R. Middleton
James W. Moore
John Napier
Cathie L. Neel
Michael S. Newman
Ciro A. Noronha
Mark C. Paulk
Miroslav Pavlovic
Howard W. Penrose
Robert A. Peterson
Donivan R. Porterfield
Cam K. Posani
Vikram Punj
Gerald M. Radack
Annette D. Reilly
Pascal Richer
Robert A. Robinson
James E. Sanders, Jr.
Helmut H. Sandmayr
Gary A. Savard
Robert J. Schaaf
Terry A. Schmidt
David J. Schultz
William J. Selph
Lynn J. Simms
Carl A. Singer
David Singleton
James M. Sivak
Mitchell W. Smith

Luca Spotorno
Friedrich Stallinger
Thomas E. Starai
Doug Steele
Gerald J. Stueve
K. S. Subrahmanyam

Douglas H. Thiele
John A. Thywissen
Winfried Tiedge
Thomas A. Tullia
Mark-Rene Uchida
Irene P. Viana

Richard J. Walker
Charlene C. Walrad
John W. Walz
Paul R. Work
Oren Yuen
Janusz Zalewski

When the IEEE-SA Standards Board approved this standard on 30 March 2006, it had the following membership:

Steve M. Mills, *Chair*
Richard H. Hulett, *Vice Chair*
Judith Gorman, *Secretary*

Mark D. Bowman
Dennis B. Brophy
Joseph Bruder
Richard Cox
Bob Davis
Julian Forster*
Joanna N. Guenin
Mark S. Halpin
Raymond Hapeman

William B. Hopf
Lowell G. Johnson
Herman Koch
Joseph L. Koepfinger*
David J. Law
Daleep C. Mohla
Paul Nikolich

T. W. Olsen
Glenn Parsons
Ronald C. Petersen
Gary S. Robinson
Frank Stone
Malcolm V. Thaden
Richard L. Townsend
Joe D. Watson
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Satish K. Aggarwal, *NRC Representative*
Richard DeBlasio, *DOE Representative*
Alan H. Cookson, *NIST Representative*

Jennie Steinhagen
IEEE Standards Project Manager, Document Development

Angela Ortiz
IEEE Standards Program Manager, Technical Program Development

Contents

1. Overview	1
1.1 Scope	1
1.2 Purpose	1
1.3 Product of standard	1
1.4 Intended audiences	1
1.5 Conformance	2
1.6 Relationship to other key standards	2
1.7 Relationship to process improvement	3
1.8 Organization of this document	4
2. Definitions and acronyms	5
2.1 Definitions	5
2.2 Acronyms	6
3. Key concepts	6
3.1 Activities	6
3.2 Elements of the software project life cycle process (SPLCP)	7
3.3 Mapping	8
3.4 Input information and output information	8
4. Implementing the standard	10
4.1 Establish the requirements for a software project life cycle process (SPLCP)	10
4.2 Select software project life cycle model (SPLCM)	10
4.3 Develop software project life cycle (SPLC)	10
4.4 Establish software project life cycle process (SPLCP)	11
4.5 Validate software project life cycle process (SPLCP)	11
Annex A (normative) Software project life cycle (SPLC) activities	12
A.1 Project Management Section of activity groups	12
A.2 Pre-Development Section of activity groups	34
A.3 Development Section of activity groups	43
A.4 Post-Development Section of activity groups	53
A.5 Support Section of activity groups	63
Annex B (informative) Mapping example	79
Annex C (informative) Information mapping template	90
Annex D (informative) Sample software project life cycle models (SPLCMs)	95
D.1 Requirements-defining software project life cycle model (SPLCM)	95
D.2 System retirement software project life cycle model (SPLCM)	97
D.3 Development and delivery software project life cycle model (SPLCM)	99

Annex E (informative) Glossary.....	102
Annex F (informative) Bibliography	103

IEEE Standard for Developing a Software Project Life Cycle Process

1. Overview

1.1 Scope

This standard provides a process for creating a software project life cycle process (SPLCP). It is primarily directed at the process architect for a given software project. It is the function of the process architect to develop the SPLCP.

This methodology begins with the selection of an appropriate software project life cycle model (SPLCM) for use on the specific project. It continues through the definition of the software project life cycle (SPLC), using the selected SPLCM, the activities provided in Annex A, and the portion of the software life cycle that is relevant to the project. The methodology concludes with the augmentation of the software life cycle with organizational process assets (OPAs) to create the SPLCP.

The activities that are provided in Annex A cover the entire life cycle of a software system, from concept exploration through the eventual retirement of the software system. This standard does not address nonsoftware activities, such as contracting, purchasing, or hardware development. It also does not mandate the use of a specific SPLCM, nor does it provide a selection of, or a tutorial on, SPLCMs. This standard presumes that the process architect is already familiar with a variety of SPLCMs, with the criteria for choosing among them and with the criteria for determining the attributes and constraints of the desired end system and the development environment that affects this selection. Finally, this standard does not prescribe how to perform the software activities in Annex A.

1.2 Purpose

This standard defines the process by which an SPLCP is developed. It is useful to any organization that is responsible for managing and performing software projects. It can be used where software is the total system or where software is part of a larger system.

1.3 Product of standard

The product of an application of this standard is the software project life cycle process (SPLCP) required for a specific software project.

1.4 Intended audiences

This standard is written to provide direction and guidance to process architects and other project personnel concerned with the implementation or performance of project processes.

1.4.1 Process architect

The primary audience for this standard is the process architect. The process architect is the person or organizational entity with the responsibility and authority to develop and maintain SPLCPs. This standard presumes that the process architect is already familiar with a variety of software project life cycle models (SPLCMs), with the criteria for choosing among them and with the criteria for determining the attributes and constraints of the desired end product and the project environment that affects this selection.

The process architect is expected to have the following expertise:

- a) An understanding of the activities in Annex A
- b) Knowledge of the OPAs
- c) Knowledge of the standards listed in Annex F
- d) Involvement in any and all software process improvement activities

1.4.2 Other interested parties

This standard may also be of use to the performers of the activities presented in Annex A (i.e., the project team).

1.5 Conformance

This standard can be applied to any SPLCM. Examples of some SPLCMs are contained in the informative Annex D. In order to conform to this standard, the development of the SPLCP shall conform to Clause 4; and the required activities identified in Annex A shall be included in the software project's SPLCP.

Many SPLCs do not address the entire software life cycle, but instead only address a subset of the software life cycle. If a SPLC uses only selected part(s) of the software life cycle, conformance to this standard may be achieved by including all the activities that impact the selected part(s) and the activities identified as *required* in Annex A. For example, if a SPLC is addressing only the designing and coding of a software artifact, all the activities in A.3.2 and A.3.3 and all the required activities shall be included. In this example, the remaining activities are optional and may be included in the SPLC as needed. If specific software development methodologies and techniques are used that combine the functionality of two or more activities, the activities not used, or combined, shall be noted and justified in the software project planned information for the project.

1.6 Relationship to other key standards

No standard lives isolated from its associated standards. This standard is related to ISO 9001 [B29] and IEEE/EIA 12207.0 [B3].

1.6.1 Relationship to ISO 9001 [B29]

The ISO 9001 family of standards recommends organizing a software development project in accordance with a selected life cycle model. It is intended that a conforming application of this standard would satisfy this recommendation; however, it would be the responsibility of the applier to assure that the developed SPLCPs satisfy specific requirements of applicable clauses. Application of an SPLCP to ISO 9001 can be facilitated through the usage of ISO/IEC 90003 [B33].

1.6.2 Relationship to IEEE/EIA 12207.0 [B3]

IEEE/EIA 12207.0 establishes a common framework for the life cycle of software in terms of the processes that can be employed to

- Acquire, supply, develop, operate, and maintain software
- Manage, control, and improve the processes
- Provide the basis for world trade in software.

IEEE/EIA 12207.0 places requirements upon the characteristics of a designated set of life cycle processes, but does not specify the detailed implementation of those processes.

IEEE Std 1074 complements the application of IEEE/EIA 12207.0 and is written for a process architect, i.e., the individual or group responsible for establishing the software project life cycle process (SPLCP) to be followed on a particular project. The process architect may use IEEE Std 1074 in developing project-specific processes complying with the requirements of IEEE/EIA 12207.0.

1.6.3 Relationship to process improvement models

This standard supports the development of organizational standard processes and the selection of a standard, defined project process that is tailored from an organization's set of standard processes (organizational process definition) and integrated project management for each individual project.¹

1.6.4 Relationship to software engineering standards

Annex F contains a list of software engineering standards that may provide insight into, or implementation suggestions for, the performance of the activities in Annex A. These and other standards may be used, but their inclusion in Annex F does not imply that conformance to them is required as part of conformance to this standard.

1.7 Relationship to process improvement

This standard can be integrated into an organization's process improvement program by using this standard as the framework for the OPAs.

Building the OPAs around this standard's structure of activities, input information, and output information can

- a) Minimize the effort to develop an SPLCP
- b) Facilitate the reuse of existing OPAs
- c) Lead to improvement of the OPAs by incorporating lessons learned from the use of the OPAs in projects

A project's SPLCP, in part or as a whole, can become part of the organization's project assets for use by future projects.

¹ Process improvement models that meet these criteria include the SEI CMMI[®] model. This information is given for the convenience of users and does not constitute an endorsement by the IEEE of these models. Equivalent models may be used if they can be shown to lead to the same results.

1.8 Organization of this document

1.8.1 Clauses

Clause 1, Clause 2, and Clause 3 contain introductory information. Clause 4 provides a brief discussion of key concepts beneficial to the understanding and use of this standard. Clause 4 provides the requirements for the creation of an SPLCP. Requirements for the content of a SPLCP are presented in Annex A, which is normative. Annex B through Annex F are informative and include useful information, but no requirements.

Table 1 presents the organization of this standard.

Table 1—Organization of this standard

Element	Title
Clause 1	Introduction
Clause 2	Definitions
Clause 3	Key concepts
Clause 4	Implementing this standard
Annex A (normative)	Activities
Annex B (informative)	Mapping example
Annex C (informative)	Information mapping template
Annex D (informative)	Sample SPLCMs
Annex E (informative)	Glossary
Annex F (informative)	Bibliography

1.8.2 Activity grouping

The available components of the SPLCP are the 69 activities, grouped into 17 activity groups, in Annex A. The full set of 69 activities covers the entire software life cycle from concept exploration through the eventual retirement of the software system.

The activity groups are further grouped into five sections as shown in Table 2.

Table 2—Activity grouping

Section	Clause	Activity groups
Project Management	A.1	Project Initiation Project Planning Project Monitoring and Control
Pre-Development	A.2	Concept Exploration System Allocation Software Importation
Development	A.3	Software Requirements Design Implementation
Post-Development	A.4	Installation Operation and Support Maintenance Retirement
Support	A.5	Evaluation Software Configuration Management Documentation Development Training

The Support Section (A.5) includes the activity groups that are necessary to assure the successful completion of a project, but are considered as supporting activities rather than activities directly oriented to the development effort.

The activity groups in the Support Section contain two types of activities:

- a) Those that are performed discretely and are, therefore, mapped onto an SPLCM.
- b) Those that are invoked (see 4.3.3) by other activities

1.8.3 Activity sequencing

This standard offers users a set of activities, grouped administratively. The orders of the sections, activity groups within sections, and activities within activity groups are for convenience only. No required, nor defined, sequence or timing of the performance of the activities, activity groups, or sections is intended, nor should any be implied.

These activities are not processes themselves. They are not intended to be executed sequentially and are not requirements for specific supporting documents. Users of this standard map these activities into a selected model in order to form processes that are appropriate for a using organization or project.

2. Definitions and acronyms

2.1 Definitions

For the purposes of this document, the following terms and definitions apply. The glossary in Annex F and *The Authoritative Dictionary of IEEE Standards Terms* [B5]² should be referenced for terms not defined in this clause.

2.1.1 software project life cycle (SPLC): The portion of the entire software life cycle applicable to a specific project. It is the sequence of activities created by mapping the activities of IEEE Std 1074 onto a selected software project life cycle model (SPLCM).

2.1.2 software project life cycle model (SPLCM): The framework selected by each using organization on which to map the activities of IEEE Std 1074 to produce the software project life cycle (SPLC).

2.1.3 software project life cycle process (SPLCP): The project-specific description of the process developed by adding the organizational process assets (OPAs) to the software project life cycle (SPLC) and the OPAs.

2.1.4 Support Activity Group: An activity group that is necessary to assure the successful completion of a project, but consists of supporting activities rather than activities directly oriented to the development effort.

2.1.5 threat modeling: A systematic exploration technique to expose any circumstance or event having the potential to cause harm to a system in the form of destruction, disclosure, modification of data, and/or denial of service. It results in a vulnerability assessment.

²The numbers in brackets correspond to the numbers of the bibliographic references listed in Annex F.

2.1.6 security accreditation: Formal declaration by management that an information technology (IT) system is approved to operate in a particular security mode using a prescribed set of safeguards at an acceptable level of risk. **ALSO:** An independent accreditation body's certification that an IT system meets a predetermined security standard.

2.2 Acronyms

PR&RPI	problem reporting and resolution planned information
SCMPI	software configuration management planned information
SPLC	software project life cycle
SPLCM	software project life cycle model
SPLCP	software project life cycle process
SPMPI	software project management planned information
SRMPI	software release management planned information

3. Key concepts

3.1 Activities

An activity is a defined body of work to be performed, including its required input information and output information. Thus, it is a description of the required transformation of input information into output information. The performance of an activity is complete when all available input information has been processed and all applicable output information has been generated.

3.1.1 Format

An activity consists of three parts:

- a) Input information: A list of the required information to be transformed and its source(s)
- b) Description: Discussion of the value-added actions to be performed to accomplish the transformation
- c) Output information: A list of the information required to be generated by the transformation and the destination(s) of that information.

3.1.2 Entry and exit criteria

To *enter*, or start, an activity, at least one element of the specified input information shall be present. To *exit*, or complete, an activity, all input information from the activities appropriate to the selected life cycle shall have been processed, and all output information shall be generated. Each project is expected to determine information flow requirements during the mapping of activities to the SPLCM.

3.1.3 Required activities

Most activities in the Project Management Section of activities group and some activities of the Support Section are required. Depending on the SPLCM chosen, activities in the Pre-Development, Development, and Post-Development sections of activity groups may or may not be required.

3.1.4 Organizational structure

This standard does not presume nor dictate an organizational structure for a software project. Therefore, it is neither implied nor required that activities within an activity group be performed by the same organizational entity or that an organizational entity's involvement be concentrated in only one activity

group. This standard does, however, presume that persons or organizations will be assigned accountability for the performance of the activities and for the quality of the input information and output information sets.

3.2 Elements of the software project life cycle process (SPLCP)

Figure 1 depicts the key concepts involved in the development of an SPLCP.

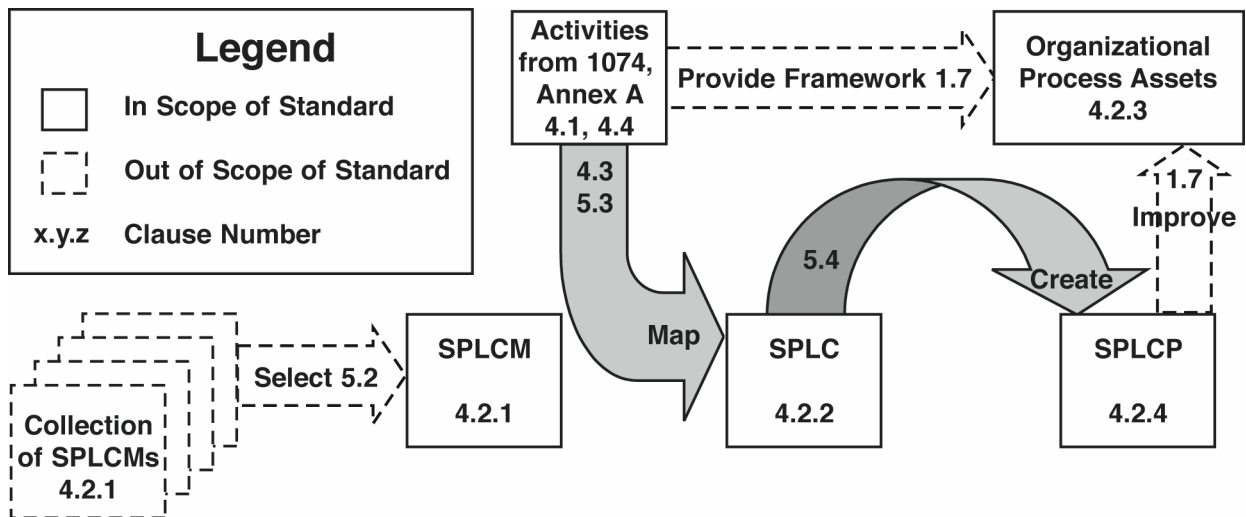


Figure 1—Developing an SPLCP

3.2.1 Software project life cycle model (SPLCM)

The SPLCM is the framework onto which the activities of this standard will be mapped to produce the software project life cycle (SPLC). To use this standard, a SPLCM shall be selected for a project. This selection is based on project attributes and organizational capabilities.

Annex D provides examples of typical models that might be considered.

3.2.2 Software project life cycle (SPLC)

The SPLC is the executable sequence of the activities to be performed during a project. The SPLC is developed by mapping the required and optional activities, provided in Annex A, onto the SPLCM selected for the project.

3.2.3 Organizational process assets (OPAs)

OPAs are the artifacts that define an organization's environment for software projects. These artifacts are selected and adapted for a particular project.

The content of an organization's process assets collection will vary from organization to organization. Definition of the collection of OPAs is the responsibility of the organization. It is recommended, however, that the organization consider including assets such as policies, standards, procedures, existing SPLCPs, metrics, tools, methodologies, etc.

3.2.4 Software project life cycle process (SPLCP)

The SPLCP is developed by augmenting the software project life cycle (SPLC) with the OPAs selected for the project. It provides the specific approach to be used for the project.

3.3 Mapping

Mapping establishes the executable sequence of activities onto a selected software project life cycle model (SPLCM). Activities can be mapped in three ways: instance, iteration, and invocation.

3.3.1 Instance

An activity is mapped as an instance if it takes all of its available specified inputs, processes them, and produces all of its applicable specified outputs. It is mapped once and appears as a single event in the SPLC. The Allocate Project Resources Activity (A.1.1.3) could be an example of a single instance mapping.

3.3.2 Iteration

An activity is mapped as an iteration if at least some input information is processed and some output information is created. Iterations are mapped until all available input information is processed and all applicable output information is created. The Manage the Project Activity (A.1.3.2) could require multiple iterations.

3.3.3 Invocation

In addition to the activities that are discretely mapped, there are activity groups of the Support Section that are invoked in parallel from many activities. An activity is invoked to further process specific information before that information is considered complete and permitted to be output by the creating activity. When invoked, these activities perform a specific function and then return to the invoking activity.

The following example is taken from the Plan Project Management Activity (A.1.2.7), with notes added. In this example, the software project management planned information (SPMPI) shall be “sent” to the three activities listed.

“Prior to distribution of the SPMPI¹, the following activities shall² be invoked³:

- a) Conduct Reviews (A.5.1.1)⁴
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)”

¹ This is the specified output information on which the invoked activities are to be performed. In other words, not all of this activity’s output information is required to be documented, controlled, and evaluated, just the SPMPI.

² Required invocations are indicated with *shall*; others are recommended (*should*) or optional (*may*). The *should* and *may* invocations are to be considered based on the needs of the project.

³ *Invoked* means to initiate a parallel task that is necessary to complete the required invoked activities and return here, before this activity can be considered complete.

⁴ This is the activity to which output information is sent. In this example, the SPMPI shall be sent to the three named activities. The evaluated, controlled, and documented information is then returned to the invoking Plan Project Management Activity (A.1.2.7).

3.4 Input information and output information

Figure 2 depicts the conceptual flow of input information and output information into and out from an activity, respectively.

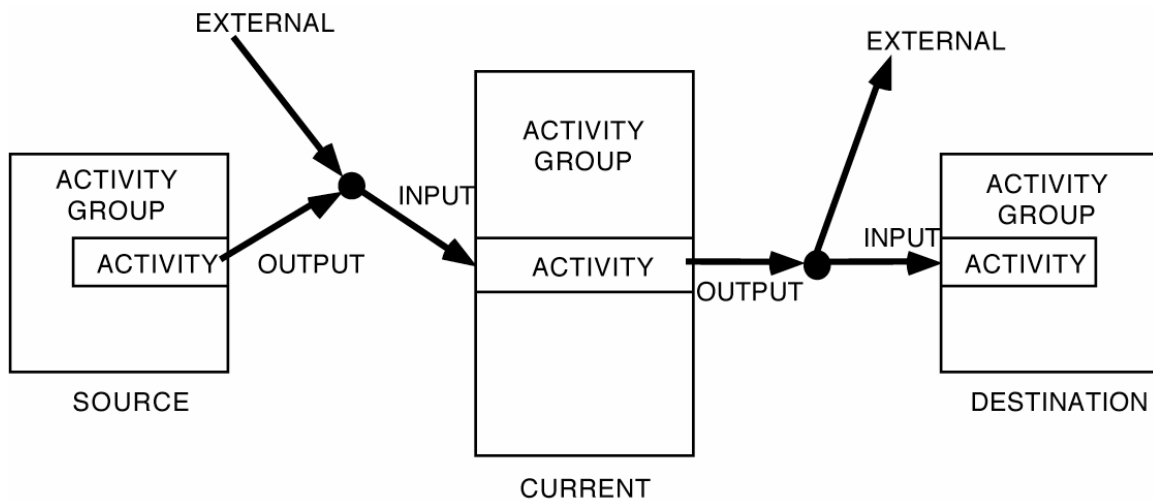


Figure 2—Information flow

Where information flows among activities, it can be traced from its original activity to the receiving activity through the input information and output information tables in Annex A.

3.4.1 Conventions

The input information and output information for each activity are listed in a two-column format in the Annex A tables. The input information or output information name is listed in the left-hand column. The information source or destination activity is shown in the right-hand column.

As a convention of this standard, input information and output information names are capitalized in an activity's description.

3.4.2 External information

External information sources and destinations are outside the purview of this standard.

External input information sources might or might not exist. If an external input does not exist, the processing listed for it is not required for completion of the activity. When an external input does exist, it shall be used.

External output information destinations will receive the information sent, if they exist. No assumption about the use of the output information by external destinations is made by this standard.

External sources and destinations are denoted by "External" in the activity column.

3.4.3 Generic information

In most cases, the input information and output information columns of the tables designate the specific information that enters or exits the activity. However, since many activities have output information whose destination is "Retain Records" (A.1.3.4), the various input information to Retain Records is collected under the term "Records." The corresponding "Activity group" and "Activity" columns refer simply to "Originating Activity Group" and "Originating Activity." The information for the Collect and Analyze Metric Data Activity (A.1.3.5) is received in the same way.

3.4.4 Information versus documents

This standard prescribes the activities of the software life cycle, not the form of the products of that life cycle. Therefore, when this standard refers to the creation of documentation, it does not require any specific

form. The information resulting from the execution of the activities is expected to be collected in whatever manner and form are consistent with the selected SPLCM and OPAs. Annex C provides a template for the assignment of output information into project-specific documents.

4. Implementing the standard

This clause describes the way in which implementation of this standard is to be approached. As stated in 1.4.1, the process architect has primary responsibility for creating and maintaining the software project life cycle process (SPLCP). This responsibility is implemented in five steps, as described below, and is performed as the Develop SPLCP Activity (A.1.1.1). An example of implementing this standard appears in Annex B.

4.1 Establish the requirements for a software project life cycle process (SPLCP)

The process architect shall assure that requirements for an SPLCP are established. Relevant stakeholders are identified, and their needs and expectations are transformed into a feasible set of requirements that are acceptable to the stakeholders.

4.2 Select software project life cycle model (SPLCM)

Initially, the process architect shall identify the SPLCM to which the activities will be mapped. This step encompasses locating, evaluating, selecting, and acquiring a SPLCM. It is possible for an organization to have multiple SPLCMs, but only one model is to be selected for a given project.

The process architect shall perform the following five steps to evaluate and select a SPLCM:

- a) Identify all the SPLCMs available to the software project.
- b) Identify the attributes that apply to the desired end system and the project environment.
- c) Identify any constraints that might be imposed on the selection of the SPLCM.
- d) Evaluate the various SPLCMs based on past experience and organizational capabilities.
- e) Select the SPLCM that will best satisfy the project attributes and constraints.

4.3 Develop software project life cycle (SPLC)

The required activities identified in Annex A, plus all others applicable to the project, shall be mapped onto the SPLCM. Note that the selected SPLCM, or the project itself, could benefit from or require activities that are not included in Annex A. Additional activities are acceptable in the SPLC. However, note that not mapping the activities identified in Annex A as “required” will result in an SPLC and, therefore, an SPLCP that are not conformant to this standard. The steps in mapping follow.

4.3.1 Place the activities in executable sequence

The order in which activities will be performed is determined by four major considerations:

- a) The selected SPLCM will dictate an initial ordering of activities; and, as mapping progresses, the actual order in which activities will be performed will be established.
- b) Schedule constraints might require the overlapping of activities in the SPLCM and, thus, impact ordering. In this case, activities might be mapped for parallel execution rather than serial execution.

- c) Selection and ordering of activities might be impacted by the entry and exit criteria of associated activities. The availability of output information from one activity could affect the start of another activity. The second activity might require, as inputs, one or more of the outputs of the first.
- d) If a selected activity is necessary to complete the mapping to the SPLCM, but one or more of its inputs come from activities not included for this SPLC, the source of the unavailable input shall be considered external. If an output is created for an activity that has not been selected and mapped, its destination for the output shall be considered external.

4.3.2 Develop and justify List of Activities Not Used

All activities that do not apply to this project shall be identified, along with the justification for their exclusion, in the List of Activities Not Used.

4.3.3 Verify the map

The process architect shall assure that the appropriate activities are fully mapped onto the selected SPLCM and that the resulting SPLC contains all of the activities necessary to successfully complete a software project.

The process architect shall also verify that the information flow into and out of the activities will support the relative order into which they have been mapped.

4.4 Establish software project life cycle process (SPLCP)

The preceding steps develop the software project life cycle (SPLC). As the next step, the available OPAs shall be applied to the SPLC activities, and known constraints shall be reconciled. The output information generated by each activity shall be assigned to the appropriate document(s). (Annex C can be used for assistance in the assignment of information to documents.) The result is the established SPLCP.

4.5 Validate software project life cycle process (SPLCP)

The SPLCP shall be validated against the set of requirements identified in 4.14.

Annex A

(normative)

Software project life cycle (SPLC) activities

The activities included in this annex are to be mapped, as described in Clause 4, to the selected software project life cycle model (SPLCM) in order to develop the project-specific life cycle process. Additional activities that are required by the project may also be mapped.

A.1 Project Management Section of activity groups

The activity groups of the Project Management Section include activities that initiate, monitor, and control a software project throughout its life cycle. The activities listed are not processes. They shall be mapped onto a software project life cycle model (SPLCM) or joined together by a process architect to become processes appropriately tailored and relevant to an organization or project.

A.1.1 Project Initiation Activity Group

The activities of the Project Initiation Activity Group create and update the infrastructure of a software development or maintenance project. They build the base for the full SPLCP.

The activities of the Project Initiation Activity Group are

- a) Develop SPLCP (A.1.1.1)
- b) Perform Estimations (A.1.1.2)
- c) Allocate Project Resources (A.1.1.3)
- d) Define Metrics (A.1.1.4)
- e) Determine Security Objectives (A.1.1.5)

A.1.1.1 Develop SPLCP (Required)

A.1.1.1.1 Input information

Input information	Source activity
Attributes	External
Available SPLCMs	External
Constraints	External
Contractual Requirements	External
Historical Records	External
IEEE Std 1074	External
OPAs	External
Environmental Improvement Needs	Identify SPLCP Improvement Needs (A.1.3.3)
Quality Policy and Quality Objectives	External
Statement of Need	Refine and Finalize the Idea or Need (A.2.1.4)
Software Requirements	Prioritize and Integrate Software Requirements (A.3.1.3)
Maintenance Recommendations	Reapply SPLCP (A.4.3.3)

A.1.1.1.2 Description

Using the input information, the process architect shall develop the SPLCP as described in the five steps of Clause 4. Any required activities not used shall be included in the List of Activities Not Used. Exclusion of any required activity, however, will preclude conformance to this standard.

Additional guidance on this topic is provided by IEEE Std 1490™ [B27].

Prior to the distribution of the SPLCP, the following activities shall be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)
- d) Develop Training Materials (A.5.4.1)

A.1.1.1.3 Output information

Output information	Destination activity
SPLCP	Perform Estimations (A.1.1.2)
	Allocate Project Resources (A.1.1.3)
	Define Metrics (A.1.1.4)
	Determine Security Objectives (A.1.1.5)
	Plan Documentation (A.1.2.5)
	Plan Training (A.1.2.6)
	Plan Project Management (A.1.2.7)
	Plan Release Management (A.1.2.9)
	Manage Risks (A.1.3.1)
	Manage the Project (A.1.3.2)
	Identify SPLCP Improvement Needs (A.1.3.3)
List of Activities Not Used	Manage Risks (A.1.3.1)

A.1.1.2 Perform Estimations (Required)

A.1.1.2.1 Input information

Input information	Source activity
Historical Project Records	External
SPLCP	Develop SPLCP (A.1.1.1)
Security Objectives	Determine Security Objectives (A.1.1.5)
Statement of Need	Refine and Finalize the Idea or Need (A.2.1.4)
System Functional Software Requirements	Allocate System Requirements (A.2.2.3)

A.1.1.2.2 Description

Based on the project requirements that are documented in the Statement of Need and the System Functional Software Requirements, size estimates of work products to be created (both deliverable and nondeliverable) shall be derived. The work products shall be decomposed to the level of granularity that is needed to plan

and track the project. Based on these size estimates, effort and cost estimates shall be created for all of the activities of the SPLC. In addition, target computer resource usage shall be estimated.

Historical Project Records should be used as the basis of estimation, when available and appropriate. All Estimation Assumptions that were made in deriving the estimates shall be specified. Project Estimates should be reaffirmed and revised throughout the SPLCP.

Prior to the distribution of Project Estimates, the following activities shall be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)

A.1.1.2.3 Output information

Output information	Destination activity
Project Estimates	Allocate Project Resources (A.1.1.3)
	Plan Project Management (A.1.2.7)
	Manage Risks (A.1.3.1)
	Manage the Project (A.1.3.2)
Estimation Assumptions	Manage Risks (A.1.3.1)

A.1.1.3 Allocate Project Resources (Required)

A.1.1.3.1 Input information

Input information	Source activity
Historical Project Records	External
Resources	External
SPLCP	Develop SPLCP (A.1.1.1)
Project Estimates	Perform Estimations (A.1.1.2)
Security Objectives	Determine Security Objectives (A.1.1.5)
Statement of Need	Refine and Finalize the Idea or Need (A.2.1.4)
System Functional Software Requirements	Allocate System Requirements (A.2.2.3)
Software Requirements	Prioritize and Integrate Software Requirements (A.3.1.3)

A.1.1.3.2 Description

Resource Allocations shall be identified at the activity level of the SPLC. Resources that are to be allocated include budget, personnel, equipment, space, and computer resources.

Historical Project Records and the Statement of Need can provide valuable insight into Resource Allocations.

A.1.1.3.3 Output information

Output information	Destination activity
Resource Allocations	Plan Project Management (A.1.2.7)
	Manage Risks (A.1.3.1)
	Manage the Project (A.1.3.2)

A.1.1.4 Define Metrics (Required)

A.1.1.4.1 Input information

Input information	Source activity
SPLCP	Develop SPLCP (A.1.1.1)
Security Objectives	Determine Security Objectives (A.1.1.5)
Evaluation Planned Information	Plan Evaluations (A.1.2.1)
SPMPI	Plan Project Management (A.1.2.7)
Software Requirements	Prioritize and Integrate Software Requirements (A.3.1.3)

A.1.1.4.2 Description

The metrics for the project, based on the SPLC, SPMP, and Software Requirements, shall be defined. Throughout the SPLC, metrics shall be applied to the products of the project and to the processes that affect the project. For each Defined Metric, Collection and Analysis Methods shall be specified.

Additional guidance on this topic is provided by IEEE Std 982.1™ [B10], IEEE Std 1044™ [B15], IEEE Std 1045™ [B16], and IEEE Std 1061™ [B18].

Prior to distributing the Defined Metrics, the Conduct Reviews Activity (A.5.1.1) should be invoked.

A.1.1.4.3 Output information

Output information	Destination activity
Defined Metrics	Plan Evaluations (A.1.2.1)
	Collect and Analyze Metric Data (A.1.3.5)
Collection and Analysis Methods	Plan Evaluations (A.1.2.1)
	Collect and Analyze Metric Data (A.1.3.5)

A.1.1.5 Determine Security Objectives (Required)

A.1.1.5.1 Input information

Input information	Source activity
Projected Value, Criticality and Data Sensitivity Assessment	External
Historical Security Records	External
Security Classification Policy	External
Validated Industry Protection Profile	External

Input information	Source activity
Internal Acceptability Standards (if required)	External
Security Stakeholders	External
OPAs	External
SPLCP	Develop SPLCP (A.1.1.1)
Preliminary Statement of Need	Identify Ideas or Needs (A.2.1.1)
Statement of Need	Refine and Finalize the Idea or Need (A.2.1.4)

A.1.1.5.2 Description

The Determine Security Objectives Activity shall determine the appropriate security assurance levels, if any, that are to be applied to the project and the product to mitigate organizational security risk. Security assurance levels shall be derived by estimating the potential likelihood and business impact of product security failure based on historical precedence and the security status of interdependent systems, in light of the high-level assessment of the proposed product's estimated business value, operational criticality, and data sensitivity. The level of security assurance shall be applied to achieve an acceptable security risk for the project, product, and specified categories of information.

First, OPAs related to security shall be collected and evaluated for relevance based on the Preliminary Statement of Need. Sources of requirements to be considered may include current organizational security, privacy and liability policy, contractor, vendor service level and partnership agreements, and the prevailing security technology and rules applied to users and groups to protect against unauthorized data access and modification. Second, opportunity for liability shall be explored through analysis of Historical Security Records, including analysis of security breach history and cost of impacts, the current security status of proposed interfacing systems, legal recommendations, and the results of security audits for similar systems. Finally, the Preliminary Statement of Need and the Projected Value, Criticality, and Data Sensitivity Assessment are combined and evaluated against the accumulated body of knowledge to determine the appropriate level of security that needs to be applied to the project. Taken together, this information will indicate an appropriate security level for the class of product under development.

Upon receipt of the Statement of Need, the Security Objectives shall be refined. If the product is required to meet industry accreditation standards, then the applicable Validated Industry Protection Profile shall be identified and applied.

If a Validated Industry Protection Profile does not exist or if such formal accreditation is not required, internal acceptability standards regarding security shall be applied. The security level identified for the product shall suggest the security level and budget also needed for the project. Therefore, policies and procedures shall be identified that are required to be applied to the project team and development environment to ensure the protection of proprietary project information and the responsible management of project assets.

Finally, Security Stakeholders shall be identified, and planning shall begin for security documentation, which may include specialized documentation for external organizational units such as legal, asset management, internal audit, maintenance and support, and business continuity.

Additional guidance on this topic is provided by ISO 15408 [B30].

Prior to the distribution of the Security Objectives, the following activities shall be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)

A.1.1.5.3 Output information

Output information	Destination activity
Security Objectives	Most activities of the Project Planning Activity Group (A.1.2)
	Perform Estimations (A.1.1.2)
	Allocate Project Resources (A.1.1.3)
	Define Metrics (A.1.1.4)
	Formulate Potential Approaches (A.2.1.2)
	Develop System Architecture (A.2.2.2)
	Identify Software Improvement Needs (A.4.3.1)
	Implement Problem Reporting Method (A.4.3.2)
	Reapply SPLCP (A.4.3.3)
	Retire System (A.4.4.3)
	Confirm Security Accreditation (A.5.1.8)

A.1.2 Project Planning Activity Group

The activities of the Project Planning Activity Group address the planning for all project management, including contingencies. These activities can be done as needed (mapped in several iterations), e.g., at every phase review or at each development iteration.

The activities of the Project Planning Activity Group are

- a) Plan Evaluations (A.1.2.1)
- b) Plan Configuration Management (A.1.2.2)
- c) Plan System Transition (if applicable) (A.1.2.3)
- d) Plan Installation (A.1.2.4)
- e) Plan Documentation (A.1.2.5)
- f) Plan Training (A.1.2.6)
- g) Plan Project Management (A.1.2.7)
- h) Plan Integration (A.1.2.8)
- i) Plan Release Management (A.1.2.9)

A.1.2.1 Plan Evaluations (Required)

A.1.2.1.1 Input information

Input information	Source activity
Defined Metrics	Define Metrics (A.1.1.4)
Collection and Analysis Methods	Define Metrics (A.1.1.4)
SPMPI	Plan Project Management (A.1.2.7)
Integration Planned Information	Plan Integration (A.1.2.8)
Risk Management Reported Information	Manage Risks (A.1.3.1)
Imported Software Requirements	Identify Imported Software Requirements (A.2.3.1)

Input information	Source activity
Preliminary Software Requirements	Define and Develop Software Requirements (A.3.1.1)
Software Requirements	Prioritize and Integrate Software Requirements (A.3.1.3)
Software Detailed Design	Perform Detailed Design (A.3.2.4)
Configuration Identification	Develop Configuration Identification (A.5.2.1)

A.1.2.1.2 Description

The Plan Evaluations Activity shall identify and describe the evaluation tasks that are necessary to assure that the software product and development efforts meet their goals, as specified in the SPMPI, and their requirements. Evaluation methods that are to be considered in this planning activity include audits, reviews, and testing. The activities and activity output information that are to be evaluated shall be identified; and the evaluation method, purpose, and scope of the evaluation for each of those activities and activity output information shall be defined. The size, complexity, and criticality of the software should dictate the minimum reviews, audits, testing, and the appropriate level of security accreditation (internal or industry). Reviews that are to be planned include peer, management, technical, operational, process improvement, security, and post-implementation. More information on reviews can be found in the Conduct Reviews Activity (A.5.1.1).

Audits shall be planned to provide an independent examination of software products and processes in order to assess their compliance with requirements and standards. More information on audits can be found in the Conduct Audits Activity (A.5.1.3).

Test planning shall be used to define the test strategy, the basic test environment, and the test structure that are needed to support the required levels of testing. Each planned test shall identify the items to be tested, the corresponding requirements, and the pass-fail criteria. Test planning shall also identify the test coverage criteria. Test planning shall be coordinated with the Plan Integration Activity (A.1.2.8).

The evaluation planning information shall include the evaluation team's organization and responsibilities and the tools, techniques, and methodologies that will be used to perform the evaluations. The planning shall include developing schedules, estimating resources, identifying special resources, staffing, and establishing exit or acceptance criteria. Evaluation planning shall also define the management controls and reporting procedures as well as the risks and contingencies. Special attention should be given to minimizing business, technical, and security risks. This planning shall be documented in the Evaluation Planned Information.

Additional guidance on this topic is provided by IEEE Std 730™ [B6], IEEE Std 828™ [B7], IEEE Std 829™ [B8], IEEE Std 982.1 [B10], IEEE Std 1008™ [B11], IEEE Std 1012™ [B12], IEEE Std 1028™ [B14], IEEE Std 1044 [B15], and IEEE Std 1045 [B16].

Prior to the distribution of the Evaluation Planned Information, the following activities shall be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)

A.1.2.1.3 Output information

Output information	Destination activity
Evaluation Planned Information	Define Metrics (A.1.1.4)
	Plan Integration (A.1.2.8)
	Manage Risks (A.1.3.1)
	Manage the Project (A.1.3.2)
	Collect and Analyze Metric Data (A.1.3.5)
	Identify Software Improvement Needs (A.4.3.1)
	Conduct Reviews (A.5.1.1)
	Conduct Audits (A.5.1.3)
	Develop Test Procedures (A.5.1.4)
	Create Test Data (A.5.1.5)
	Execute Tests (A.5.1.6)
	Confirm Security Accreditation (A.5.1.8)

A.1.2.2 Plan Configuration Management (Required)

A.1.2.2.1 Input information

Input information	Source activity
Deliverable List	External
Security Objectives	Determine Security Objectives (A.1.1.5)
SPMPI	Plan Project Management (A.1.2.7)
SRMPI	Plan Release Management (A.1.2.9)
Imported Software Requirements	Identify Imported Software Requirements (A.2.3.1)

A.1.2.2.2 Description

The Plan Configuration Management Activity shall plan and document specific software configuration management organizations and responsibilities, procedures, tools, techniques, and methodologies in the software configuration management planned information (SCMPI). The SCMPI shall also describe how and when such procedures are to be performed.

Overall software configuration management objectives are derived using internal guidelines as well as contractual or other agreed-upon requirements from the SPMPI. The software configuration management approach should be compatible with the approaches that are being used on associated systems.

Software configuration management includes the evaluation, coordination, approval or disapproval, and implementation of changes to product components (e.g., code, documentation) after a baseline has been established.

Items that are to be managed should include code, documentation, plans, specifications, project policies, procedures, and other artifacts. The configuration identification defined in the Develop Configuration Identification Activity (A.5.2.1) should be included in the planned information once it is developed.

The configuration management planning shall include developing schedules, estimating resources, identifying special resources, staffing, and defining management controls and reporting procedures.

Additional guidance on this topic is provided by IEEE Std 828 [B7].

Prior to the distribution of the SCMPI, the following activities shall be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)

A.1.2.2.3 Output information

Output information	Destination activity
SCMPI	Plan System Transition (if applicable) (A.1.2.3)
	Plan Release Management (A.1.2.9)
	Manage the Project (A.1.3.2)
	Retain Records (A.1.3.4)
	All activities of the Software Configuration Management Activity Group (A.5.2)

A.1.2.3 Plan System Transition

A.1.2.3.1 Input information

Input information	Source activity
Retirement Planned Information (for the system being replaced)	External
Security Objectives	Determine Security Objectives (A.1.1.5)
SCMPI	Plan Configuration Management (A.1.2.2)
SRMPI	Plan Release Management (A.1.2.9)
Preliminary Statement of Need	Identify Ideas or Needs (A.2.1.1)
Recommendations	Conduct Feasibility Studies (A.2.1.3)
Imported Software Requirements	Identify Imported Software Requirements (A.2.3.1)

A.1.2.3.2 Description

The Plan System Transition Activity is applicable only when an existing system (automated or manual) is being replaced with a new or revised system. The transition shall be planned and documented in accordance with the Retirement Planned Information of the system being replaced, the Preliminary Statement of Need, and the recommended solutions. Transition strategies and tools shall be part of the Transition Planned Information. A Transition Impact Statement shall also be produced, including an evaluation of transition security risk and mitigation strategy.

The transition planning information shall include the transition team's organization and responsibilities as well as the tools, techniques, and methodologies that are needed to perform the transition.

The planning shall include developing schedules, estimating resources, identifying special resources, and staffing. Transition planning shall also define management and security controls and reporting procedures as well as the risks and contingencies. Special attention should be given to minimizing operational risks, especially those that might compromise security. This planning shall be documented in the Transition Planned Information.

The transition planning activities shall include a review and revision of the SCMPI in respect of these plans, tools, and controls.

Prior to the distribution of the Transition Planned Information, the following activities should be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)

A.1.2.3.3 Output information

Output information	Destination activity
Transition Planned Information	Plan Installation (A.1.2.4)
	Manage the Project (A.1.3.2)
Transition Impact Statement	Manage Risks (A.1.3.1)
	Refine and Finalize the Idea or Need (A.2.1.4)

A.1.2.4 Plan Installation

A.1.2.4.1 Input information

Input information	Source activity
Security Objectives	Determine Security Objectives (A.1.1.5)
Transition Planned Information	Plan System Transition (A.1.2.3)
SPMPI	Plan Project Management (A.1.2.7)
SRMPI	Plan Release Management (A.1.2.9)
Imported Software Requirements	Identify Imported Software Requirements (A.2.3.1)
Installation Requirements	Define and Develop Software Requirements (A.3.1.1)
Operating Documentation	Create Operating Documentation (A.3.3.2)

A.1.2.4.2 Description

The tasks to be performed during the Plan Installation Activity shall be described in the Software Installation Planned Information. The Installation Requirements and the other input information are analyzed in order to guide the development of the Software Installation Planned Information. This planned information, the associated documentation, and the developed software are used to install the software product.

The Software Installation Planned Information shall include the required hardware and other constraints (e.g., minimum memory requirements, color monitor), detailed instructions for the installer, and any additional steps that are required prior to the operation of the system (e.g., registering the software). The type of software to be installed and the expected level of expertise of the installer shall be considered when writing installation instructions.

In some cases, the installation planning shall include defining the order of installation at several sites. It could also define one or more configurable options that are to be handled in the installation process.

Prior to the distribution of the Software Installation Planned Information, the following activities shall be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Perform Configuration Control (A.5.2.2)
- c) Implement Documentation (A.5.3.1)
- d) Develop Training Materials (A.5.4.1)

A.1.2.4.3 Output information

Output information	Destination activity
Software Installation Planned Information	Manage the Project (A.1.3.2)
	Distribute Software (A.4.1.1)

A.1.2.5 Plan Documentation (Required)

A.1.2.5.1 Input information

Input information	Source activity
Contractual Requirements	External
SPLCP	Develop SPLCP (A.1.1.1)
Security Objectives	Determine Security Objectives (A.1.1.5)
SPMPI	Plan Project Management (A.1.2.7)
Imported Software Requirements	Identify Imported Software Requirements (A.2.3.1)

A.1.2.5.2 Description

In the Plan Documentation Activity, information such as the SCMPI, product descriptions, organizational security requirements, schedules, and resource constraints shall be assimilated to create a consistent and disciplined approach to achieving the required documentation. The approach shall identify the required documents, the document production and delivery schedules, the documentation security classifications and access levels, and the documentation standards. Responsible organizations, information sources, and intended audiences shall be defined for each document. The approach shall be documented in the Documentation Planned Information. The Documentation Planned Information shall include resource allocations for this activity.

Additional guidance on this topic is provided by IEEE Std 1063™ [B20].

Prior to the distribution of the Documentation Planned Information, the following activities shall be invoked:

- a) Conduct Reviews (A.5.1.1)
- b) Implement Documentation (A.5.3.1)

The Perform Configuration Control Activity (A.5.2.2) should also be invoked.

A.1.2.5.3 Output information

Output information	Destination activity
Documentation Planned Information	Manage the Project (A.1.3.2)
	Retain Records (A.1.3.4)
	Create Operating Documentation (A.3.3.2)
	All activities of the Documentation Development Activity Group (A.5.3)

A.1.2.6 Plan Training

A.1.2.6.1 Input information

Input information	Source activity
Applicable Information	External
Skills Inventory	External
SPLCP	Develop SPLCP (A.1.1.1)
Security Objectives	Determine Security Objectives (A.1.1.5)
SPMPI	Plan Project Management (A.1.2.7)
Imported Software Requirements	Identify Imported Software Requirements (A.2.3.1)
Software Requirements	Prioritize and Integrate Software Requirements (A.3.1.3)
Training Feedback	Validate the Training Program (A.5.4.2)
	Implement the Training Program (A.5.4.3)

A.1.2.6.2 Description

The Plan Training Activity shall identify the needs for different types of training and the categories of people that require training for each need. Customer and project information shall be reviewed, along with existing personnel inventories. Both internal (e.g., project team, sales force) and external (e.g., customers, users, dealers) training needs shall be identified. Responsible organizations, information sources, and the intended audiences shall be defined for each type of training. Training tools, techniques, and methodologies shall be specified.

The planning shall include developing schedules, estimating resources, identifying special resources, staffing, and establishing exit or acceptance criteria. This planning shall be documented in the Training Planned Information.

Additional guidance on this topic is provided by Australian Standard (AS) 3563.1 [B1].

Prior to the distribution of the Training Planned Information, the following activities shall be invoked:

- Conduct Reviews (A.5.1.1)
- Perform Configuration Control (A.5.2.2)
- Implement Documentation (A.5.3.1)