SPICE

Consolidated product

Software Process Assessment –
Part 1 : Concepts and introductory guide

Version 1.00

(Formerly IG Version 1.00)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

These terms and conditions apply to the set of documents developed by the SPICE Project, and published within the Project as Version 1.0, with the following titles:

– Part 1 : *Concepts and introductory guide*
– Part 2 : *A model for process management*
– Part 3 : *Rating processes*
– Part 4 : *Guide to conducting assessment*
– Part 5 : *Construction, selection and use of assessment instruments and tools*
– Part 6 : *Qualification and training of assessors*
– Part 7 : *Guide for use in process improvement*
– Part 8 : *Guide for use in determining supplier process capability*
– Part 9 : *Vocabulary*

1. You may copy and distribute verbatim copies of any or all of the Documents as you receive them, in any medium, provided that you conspicuously and appropriately publish with each copy a copy of these Terms and Conditions. You may charge a fee for the physical act of transferring a copy.
2. You may copy extracts from these documents in materials for internal or public use, providing you provide clear acknowledgment of the source of the material, by citation or other appropriate means.
3. You may not copy, modify, sub-license, or distribute the Documents except as expressly provided under these Terms and Conditions.

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout
Members:	Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : *Concepts and introductory guide*

Product Manager: Terry Rout

– Part 2 : *A model for process management*

Product Managers: Al Graydon, Mark Paulk

– Part 3 : *Rating processes*

Product Manager: Harry Barker

– Part 4 : *Guide to conducting assessment*

Product Manager: Harry Barker

– Part 5 : *Construction, selection and use of assessment instruments and tools*

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : *Qualification and training of assessors*

Product Manager: Ron Meegoda

– Part 7 : *Guide for use in process improvement*

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : *Guide for use in determining supplier process capability*

Product Manager: John Hamilton

– Part 9 : *Vocabulary*

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper.

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 1

Concepts and Introductory Guide

Contents

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialing to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being co-ordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

This part of the International Standard (part 1) is for guidance only.

Introduction

Overview

This International Standard provides a framework for the assessment of software processes. This framework can be used by organizations involved in planning, managing, monitoring, controlling, and improving the acquisition, supply, development, operation, evolution and support of software.

The Standard provides a structured approach for the assessment of software processes for the following purposes:

a) by or on behalf of an organization with the objective of understanding the state of its own processes for process improvement;

b) by or on behalf of an organization with the objective of determining the suitability of its own processes for a particular requirement or class of requirements;

c) by or on behalf of one organization with the objective of determining the suitability of another organization's processes for a particular contract or class of contracts.

The framework for process assessment:

a) encourages self-assessment;

b) takes into account the context in which the assessed processes operate;

c) produces a set of process ratings (a process profile) rather than a pass/fail result;

d) through the generic practices, addresses the adequacy of the management of the assessed processes;

e) is appropriate across all application domains and sizes of organization.

The sophistication and complexity required of a process is dependent upon its context. For instance the planning required for a five person project team is much less than for a fifty person team. This context influences how a qualified assessor judges a practice when assessing its adequacy and influences the degree of comparability between process profiles.

The process assessment framework is based on assessing a specific process instance. A process instance is a singular instantiation of a process that is uniquely identifiable and about which information can be gathered in a manner that provides repeatable ratings. Each process instance is characterized by a set of five process capability level ratings, each of which is an aggregation of the practice adequacy ratings that belong to that level. Hence the practice adequacy ratings are the foundation for the rating system.

Practice adequacy is a rating of the extent to which a practice meets its purpose as defined in part 2 of this International Standard. The Standard therefore provides a rating framework that is as much an assessment of effectiveness as it is of conformance to the practice definition. From the ratings of process instances, a number of derived or average ratings can be determined that provide better insight into the capability of a process within an organizational unit as a whole.

Field of application

Within a process improvement context, process assessment provides the means of characterizing the current practice within an organizational unit in terms of the capability of the selected processes. Analysis of the results in the light of the organization's business needs identifies strengths, weakness and risks inherent in the processes. This, in turn, leads to the ability to determine whether the processes are effective in achieving their goals, and to identify significant causes of poor quality, or over runs in time or cost. These provide the drivers for prioritizing improvements to processes.

Process capability determination is concerned with analysing the proposed capability of selected processes against a target process capability profile in order to identify the risks involved in undertaking a project using the selected processes. The proposed capability may be based on the results of relevant previous process assessments, or may be based on an assessment carried out for the purpose of establishing the proposed capability.

Two of the parts of this International Standard (parts 7 and 8) address the use of process assessment for process improvement and for process capability determination. Other documents in the suite address various aspects relating to process assessment.

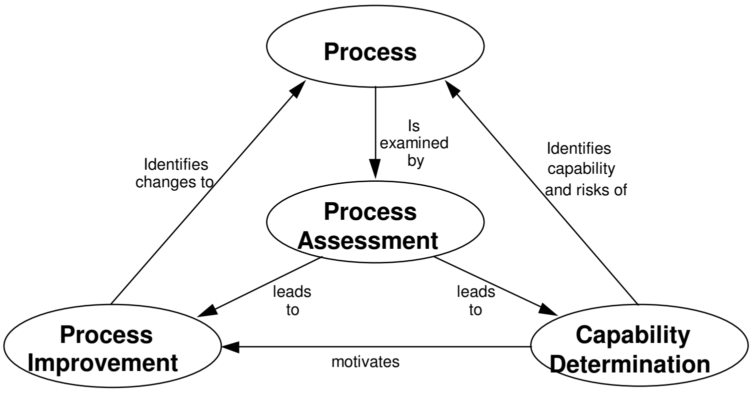


Figure 1 – Software Process Assessment

This International Standard has been designed to satisfy the needs of acquirers, suppliers and assessors, and their individual requirements from within a single source.

The benefits arising from the use of this suite of documents include:

For **acquirers:**

- an ability to determine the current and potential capability of a supplier's software processes.

For **suppliers:**

- an ability to determine the current and potential capability of their own software processes;
- an ability to define areas and priorities for software process improvement;
- a framework that defines a route map for software process improvement.

For **assessors:**

- a framework that defines all aspects of conducting assessments.

Components of this International Standard

This International Standard is comprised of nine parts. This section describes each of the parts and its role within the Standard.

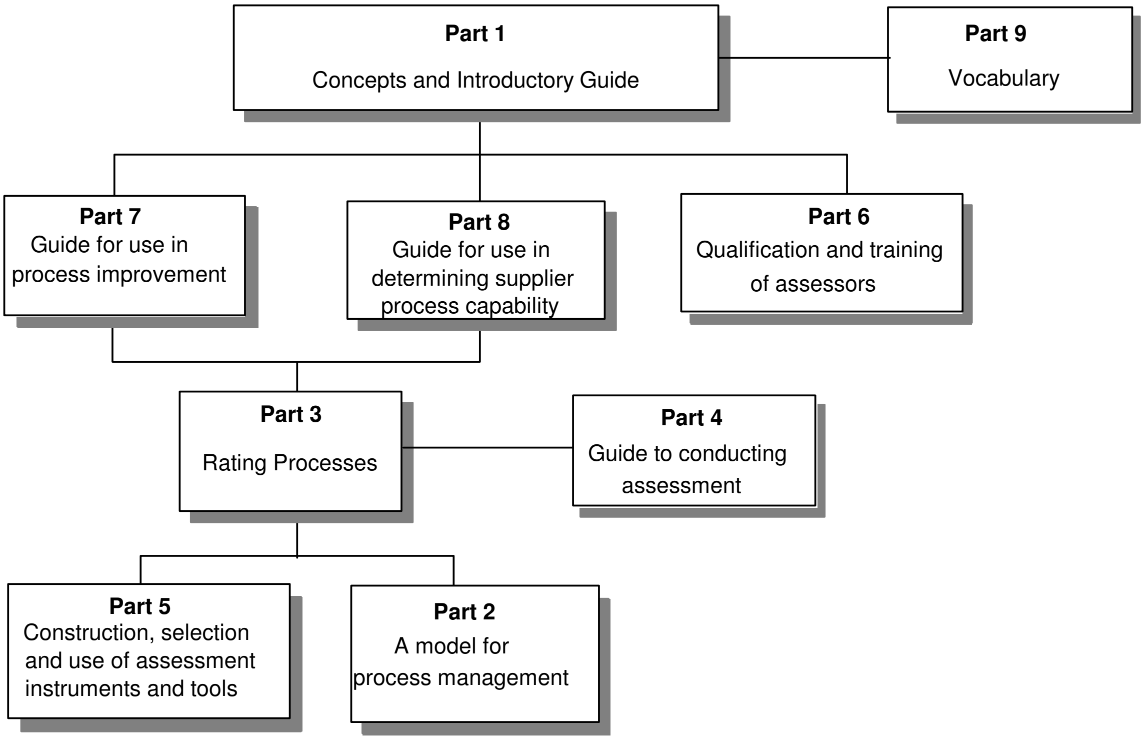

Part 1
Concepts and Introductory Guide
Part 9
Vocabulary
Part 7
Guide for use in process improvement
Part 8
Guide for use in determining supplier process capability
Part 6
Qualification and training of assessors
Part 3
Rating Processes
Part 4
Guide to conducting assessment
Part 5
Construction, selection and use of assessment instruments and tools
Part 2
A model for process management

Figure 2 – Components of this International Standard

Part 1 (informative) is an entry point into this International Standard. It describes how the parts of the suite fit together, and provides guidance for their selection and use. It explains the requirements contained within the Standard and their applicability to the conduct of an assessment, to the construction and selection of supporting tools, and to the construction of extended processes. Extended processes are processes which include base practices additional to those defined in the part 2 of the Standard, or which are entirely new processes, for example to meet industry specific requirements.

Part 2 (normative) of this International Standard defines, at a high level, the fundamental activities that are essential to software engineering, structured according to increasing levels of process capability. These baseline practices may be extended, through the generation of application or sector specific practice guides, to take account of specific industry, sector or other requirements.

Part 3 (normative) of this International Standard defines a framework for conducting an assessment, and sets out the basis for rating, scoring and profiling process capabilities.

Part 4 (informative) of this International Standard provides guidance on the conduct of team-based software process assessments. This guidance is generic enough to be applicable across all organizations, and also for performing assessments using a variety of different methods and techniques, and supported by a range of tools.

Part 5 (normative) of this International Standard defines the framework elements required to construct an instrument to assist an assessor in the performance of an assessment. In addition, it provides guidance to acquirers or designers on the selection and usability aspects of various types of assessment instruments.

Part 6 (informative) of this International Standard describes the competence, education, training and experience of assessors that are relevant to conducting process assessments. It describes mechanisms that may be used to demonstrate competence and to validate education, training and experience.

Part 7 (informative) of this International Standard describes how to define the inputs to and use the results of an assessment for the purposes of process improvement. The guide includes examples of the application of process improvement in a variety of situations.

Part 8 (informative) of this International Standard describes how to define the inputs to and use the results of an assessment for the purpose of process capability determination. It addresses process capability determination in both straightforward situations and in more complex situations involving, for example, future capability. The guidance on conducting process capability determination is applicable either for use within an organization to determine its own capability, or by a acquirer to determine the capability of a (potential) supplier.

Part 9 (informative) is a consolidated vocabulary of all terms specifically defined for the purposes of this International Standard.

Relationship to other International Standards

This International Standard is complementary to several other International Standards and other models for evaluating the capability and effectiveness of organizations and processes. This section describes the relationship between this International Standard and the major related International Standards.

This International Standard incorporates the intent of the ISO 9000 series to provide confidence in a supplier's quality management whilst providing acquirers with a framework for assessing whether potential suppliers have the capability to meet their needs. Process assessment provides users with the ability to evaluate process capability on a continuous scale in a comparable and repeatable way, rather than using the pass/fail characteristic of quality audits based on ISO 9001. In addition, the framework described in this International Standard provides the opportunity to adjust the scope of assessment to cover specific processes of interest, rather than all of the processes used by an organizational unit.

This International Standard is related in particular to the following components of the ISO 9000 series:

- ISO 9001 - 1994, *Model for quality assurance in design, development, production, installation and servicing;*
- ISO 9000-3 - 1991, *Quality management and quality assurance standards - Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software;*
- ISO 9004-4 - 1993, *Quality management and quality system elements - Part 4: Guidelines for quality improvement.*

This International Standard, and particularly part 2, is strongly related to:

- ISO/IEC12207-1-1994, *Software life cycle processes.*

Where software-based tools are developed or used to support assessments their conformance to the requirements of part 5 of this International Standard may be evaluated following the requirements of:

- ISO/IEC12119-1995, *Software products - Evaluation and test.*

Criteria for the development and/or acquisition of software-based tools are based on the characteristics defined in:

- ISO/IEC 9126-1991, *Software quality characteristics.*

1 Scope

This part of this International Standard provides overall information on the concepts of software process assessment and its use in the two contexts of process improvement and process capability determination. It describes how the parts of the suite fit together, and provides guidance for their selection and use. It explains the requirements contained within this International Standard, and their applicability to the conduct of an assessment, to the construction and selection of supporting tools, and to the construction of extended processes.

Readers of this guide should familiarize themselves with the terminology and structure of the document suite, and then reference the appropriate parts of the suite for the context in which they propose to conduct an assessment. If the assessment is to be conducted for the purposes of internal process improvement within an organization, the relevant context is in part 7 of this International Standard. If the results of the assessment are to be used for the purposes of determining the process capability of the organizational unit in the context of a specified requirement, the guidance is in part 8 of this International Standard.

More detailed description of the use of this International Standard is given in clause 4 of this guide.

2 Normative references

There are no normative references for this part of the International Standard.

3 Definitions

For the purposes of this part of this International Standard, the definitions in *Software Process Assessment - Part 9 : Vocabulary* apply.

4 Overview

4.1 General

This International Standard provides a framework for the assessment of software processes. This framework can be used by organizations involved in planning, managing, monitoring, controlling and improving the acquisition, supply, development, operation, evolution and support of software.

Process assessment examines the processes used by an organization to determine whether they are effective in achieving their goals. The assessment characterizes the current practice within an organizational unit in terms of the capability of the selected processes. The results may be used to drive process improvement activities or process capability determination by analysing the results in the context of the organization's business needs, identifying strengths, weaknesses and risks inherent in the processes.

The documents provide a structured approach to software process assessment for the following purposes:

a) by or on behalf of an organization with the objective of understanding the state of its own processes for process improvement;

b) by or on behalf of an organization with the objective of determining the suitability of its own processes for a particular requirement or class of requirements;

c) by or on behalf of one organization with the objective of determining the suitability of another organization's processes for a particular contract or class of contracts.

The high level view of the relationships between process assessment, process improvement and process capability determination is shown in figure 3, along with an indication of the places of the various components of this International Standard in the processes.

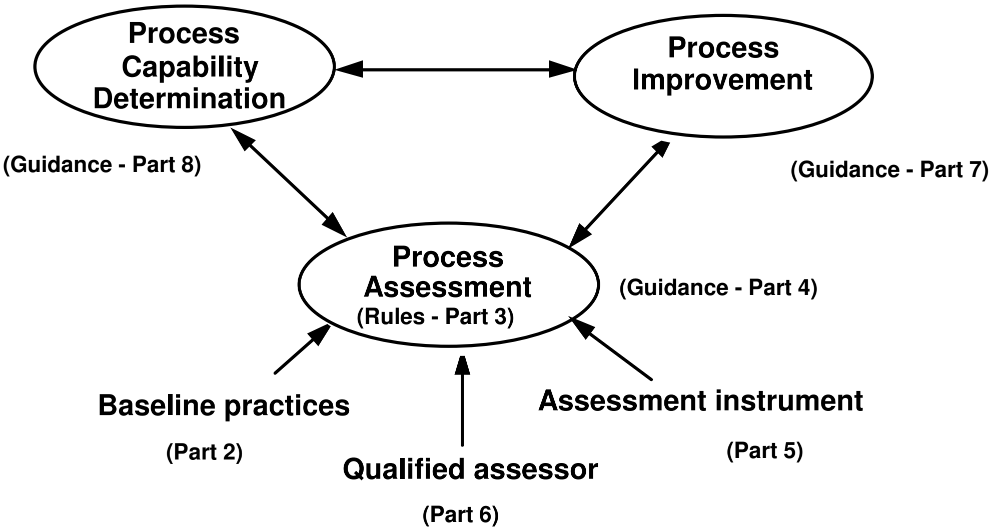


Figure 3 – Overview of relationships of elements of this Standard

The standard is designed to provide assessment results that are repeatable, objective, comparable within similar contexts, and able to be used for either process improvement or process capability determination.

Dependable assessment results are achieved through the definition of a framework for the conduct of assessments. The framework includes an architecture for rating practices and processes and for presenting assessment ratings. The assessment framework also provides guidance on the conduct of the assessment, supported by an assessment instrument to assist in the objective rating of processes. This International Standard provides guidance in the contexts of both process improvement and process capability determination. It further provides a definition of the required skills and experience for assessors.

This section describes how the other parts of this International Standard can be used to conduct and use process assessments. The key determinant in the use of the Standard is the purpose for which the assessment is being conducted. This may be:

- to promote an understanding of the software process;
- to support process improvement;
- to support process capability determination.

4.2 The Assessment framework

4.2.1 The context of process assessment

The context of a process assessment is summarized in figure 4. Part 3 of this International Standard defines the requirements for conducting an assessment, sets out the basis for rating, scoring and profiling process capabilities, and defines the circumstances under which assessment results may be compared. Part 4 provides guidance on conducting a team-based assessment and interpreting the requirements in part 3. This guidance is generic enough to be applicable across all organizations, and for conducting assessments using a variety of methods, techniques and tools.

Process assessment is an activity that is performed either during a process improvement initiative as described in part 7 of this International Standard, or as part of a process capability determination exercise as described in part 8. In either case, the formal entry to the assessment processes occurs with the compilation of the assessment input which defines the purpose of the assessment (why it is being carried out), the scope of the assessment (which processes are to be assessed) and what constraints, if any, apply to the assessment. The assessment input also defines the responsibility for carrying out the assessment and gives definitions for any processes within the scope of the assessment that are extensions of the processes defined in part 2.

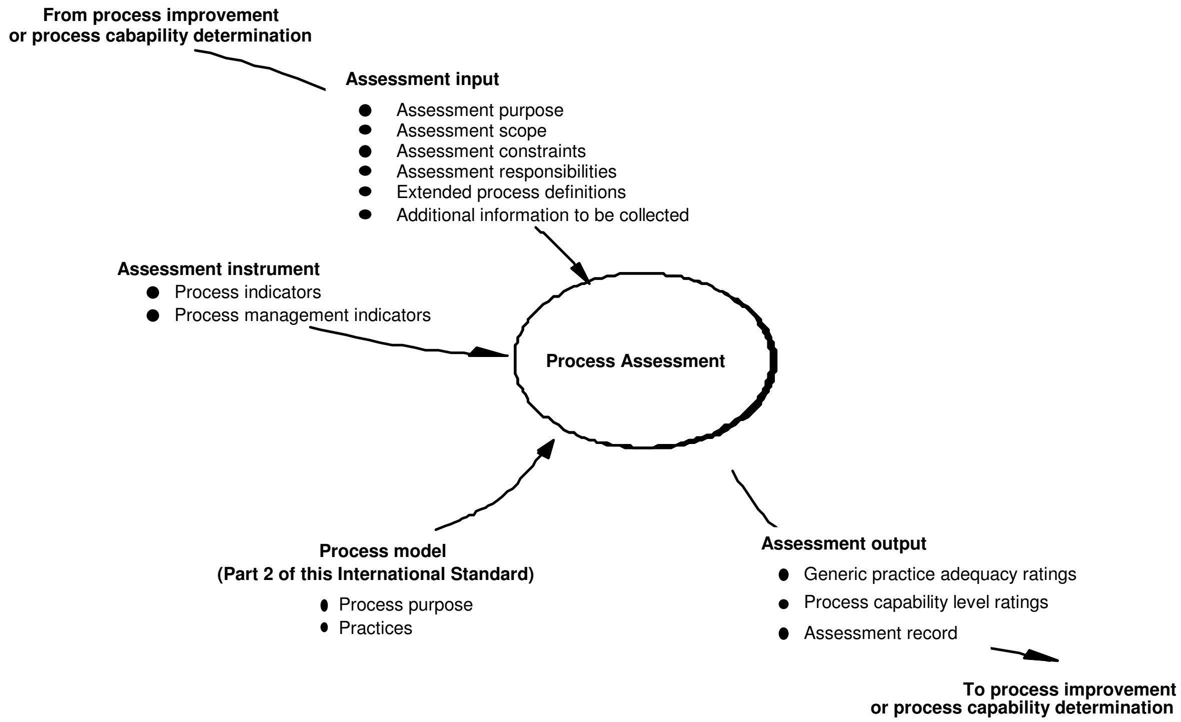

From process improvement
or process cabapility determination
Assessment input
● Assessment purpose
● Assessment scope
● Assessment constraints
● Assessment responsibilities
● Extended process definitions
● Additional information to be collected
Assessment instrument
● Process indicators
● Process management indicators
Process Assessment
Process model
(Part 2 of this International Standard)
● Process purpose
● Practices
Assessment output
● Generic practice adequacy ratings
● Process capability level ratings
● Assessment record
To process improvement
or process capability determination

Figure 4 – Context of process assessment

An assessment is carried out by assessing selected processes against the process model defined in part 2 of this International Standard. This two-dimensional model consists of a set of process-specific base practices and a set of generic practices. The generic practices apply across all processes. The generic practices are grouped into common features and capability levels that may be used to determine how well the process is managed. The assessment output includes a set of process capability level ratings for each process instance assessed.

An assessment is supported by an assessment instrument, or set of instruments, constructed according to part 5 of this International Standard. The process assessment is carried out either by a team with at least one qualified assessor who has the competence described in part 6; or, on a continuous basis using suitable tools for data collection and verified by a qualified assessor.

4.2.2 An architecture for software processes

Part 2 of this International Standard defines, at a high level, the fundamental activities that are essential to good software engineering. It describes what activities are required, not how they are to be implemented. The baseline practices may be extended through the generation of application or sector specific practice guides to take account of specific industry, sector, or other requirements.

Each process in part 2 is described by base practices, which are the essential activities of the specific process. Processes are grouped in turn into five process categories as shown in the table below.

Table 1 – Description of process categories

Process category	**Brief description**
Customer-Supplier	Processes that directly impact the customer
Engineering	Processes that specify, implement, or maintain a system and software product
Project	Processes that establish the project, and co-ordinate and manage its resources
Support	Processes that enable and support the performance of the other processes on the project
Organization	Processes that establish the business goals of the organization and develop process, product, and resource assets which will help the organization achieve its business goals

Evolving process capability is expressed in terms of capability levels, common features, and generic practices. Generic practices are applicable to all process. These practices represent the activities necessary to manage a process and improve its capability to achieve desired outputs. They are grouped into common features and capability levels which help define how the process will be managed towards achieving its defined purpose.

4.2.3 Tools to support process assessment

Part 5 of this International Standard provides the framework for building an assessment instrument. An assessment instrument is a tool, or set of tools, used during the performance of an assessment to assist the assessor in obtaining reliable, consistent and repeatable results.

An assessment instrument built according to requirements and guidance in part 5 contains a set of assessment indicators which help the assessor to analyse the process under review, and to make consistent judgements about the implemented practices. In addition, an assessment instrument provides a mechanism for the assessor to record notes and results, and may provide a means of capturing other types of information for use in process improvement or process capability determination. An assessment instrument may also provide assistance to the assessor in analysing ratings and compiling process profiles.

Part 5 does not attempt to prescribe a particular format for an assessment instrument, which could be implemented as a manual, paper-based tool; a questionnaire; an automated, on-line tool; or even as an expert system. It does, however, provide a common framework, and prescribes a set of elements that should be incorporated into any type of assessment instrument. In addition, it provides guidance for designers, users and acquirers of assessment instruments about the characteristics and usability of different types of instrument.

4.3 Assessor training and qualification

The qualified assessor in a team has the pivotal role of ensuring that other team members collectively have the right blend of specialized knowledge and assessment skills. The qualified assessor provides the necessary guidance to the team, and helps to moderate the judgements and ratings made by the other members of the team to ensure consistency of interpretation.

Part 6 of this International Standard provides guidance for the preparation and qualification of assessors to perform assessments. Specifically, its purpose is to define initial and ongoing qualification of assessors. It is concerned with assessor competencies and appropriate education, training and experience, and includes mechanisms that may be used to demonstrate competence and to validate education, training and experience.

4.4 Process improvement context

Successful software process improvement occurs in a business context by addressing specific needs and business goals of the organization that are clearly stated and understood.

Part 7 of this International Standard provides guidance on using software process assessment as part of a complete framework and method for performing software process improvement in a continuous cycle although there is no reason why the organization could not employ the guidance for a single cycle of improvement activity. The guidance covers:

- invoking a software process assessment;
- using the results of a software process assessment;
- measuring software process effectiveness and improvement effectiveness;
- identifying improvement actions aligned to business goals;
- using the process model in part 2 of this International Standard as a route map for improvement;
- cultural issues in the context of software process improvement;
- dealing with management issues for software process improvement.

The guidance provided does not presume specific organizational structures, management philosophies, software life cycle models or software development methods. The concepts and principles are appropriate for the full range of different business needs, application domains and size of organization, so that they may be used by all types of software organizations to guide their improvement activities.

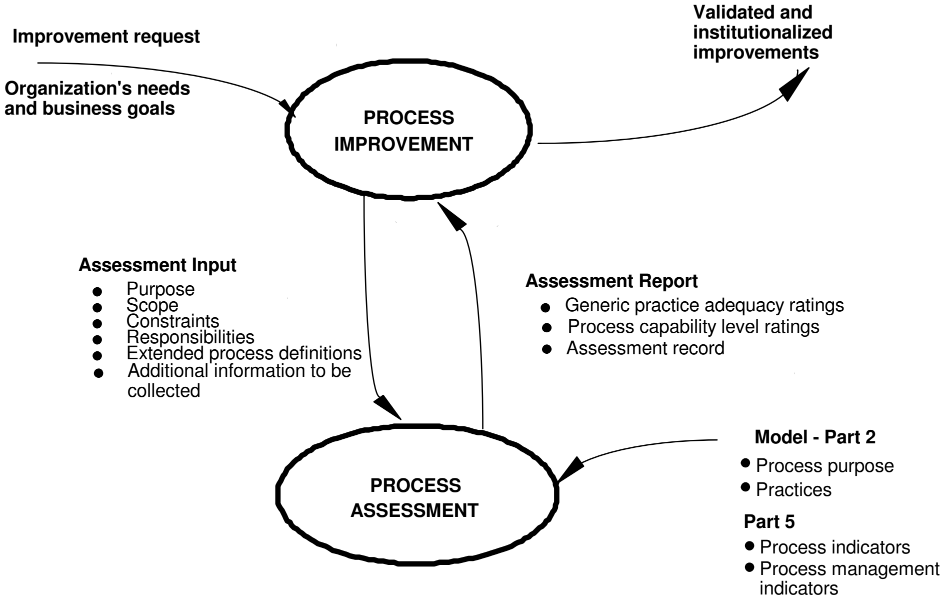

Improvement request
Organization's needs and business goals
Validated and institutionalized improvements
PROCESS IMPROVEMENT
Assessment Input
● Purpose
● Scope
● Constraints
● Responsibilities
● Extended process definitions
● Additional information to be collected
Assessment Report
● Generic practice adequacy ratings
● Process capability level ratings
● Assessment record
PROCESS ASSESSMENT
Model - Part 2
● Process purpose
● Practices
Part 5
● Process indicators
● Process management indicators

Figure 5 – Process improvement

4.5 Process capability determination context

The procedure for process capability determination is described in part 8 of this International Standard. Process capability determination is mainly built upon process assessment as described in the Standard. Processes are rated against the process model defined in part 2, using the measurement and rating framework defined in part 3. The context of process capability determination is shown in figure 6.

An acquirer of software products or services has technical and other needs as expressed in the specified requirements. Before making a contract the acquirer may need to determine the process capability of the prospective contractor, or a supplier may want to ascertain its own process capability before responding to an acquirer's proposal. The technical and other needs for process capability determination are documented in the specified requirements.

The specified requirement is translated into a target capability that represents the required process capability, and process assessment input that will scope the process assessment. The supplier may put forward a proposed process capability as a set of process-by-process capability level ratings to be offered by the organizational unit concerned. In a straightforward situation, the proposed process capability may be based on a recent self-assessment or by other means. In more complex cases, a supplier may propose a process capability to be achieved in the future based on the supplier's current profile and relevant improvement plans, backed up if possible with improvement records, or a constructed capability including the capability of one or more sub-contractors or partners.

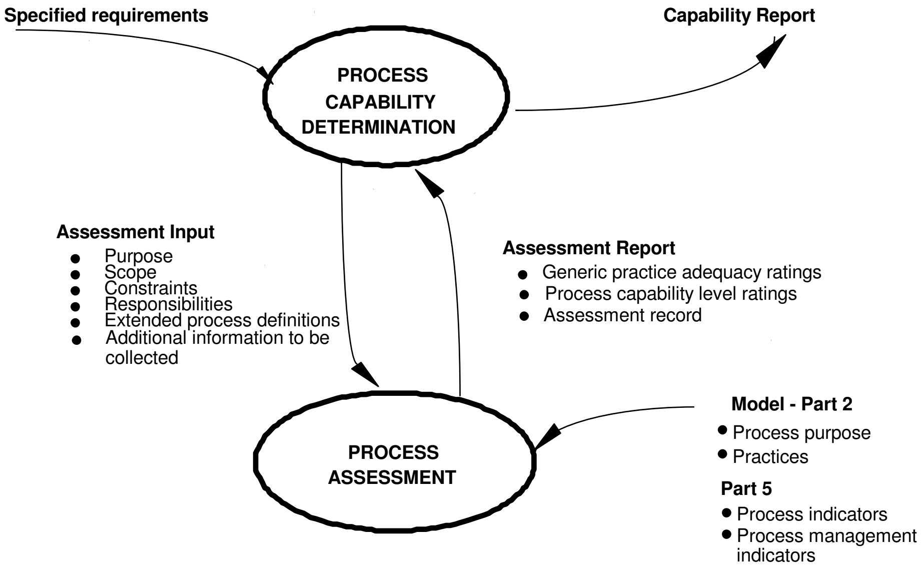


Figure 6 – Process capability determination

The credibility of the proposed process capability is analysed together with the risks involved and reported on in the process capability report.

Part 8 of this International Standard provides guidance on how to use the results of an assessment for the purpose of determining the process capability of suppliers. The guide specifically addresses process capability determination both for use within an organization to determine the risks associated with undertaking a new project (sometimes called first party use) and for use by an acquirer for assessing external suppliers (sometimes called second party or contractual use).

5 Conformance

This International Standard contains three principal areas where conformance may be claimed: in the conduct of a software process assessment, in the development of processes which extend those listed in the defined model, and in the construction of assessment instruments. This clause explains the nature of conformance in these three areas.

5.1 Conducting software process assessments

5.1.1 Overview of the requirements

This guidance describes how to conduct assessments in such a way that conformance to the requirements can be readily demonstrated.

The requirements, set out in part 3 of this International Standard for conducting a software process assessment are designed to ensure that the results are reliable, consistent and repeatable. This is important where an organization wishes to compare its assessment results with those of other similar organizations. It is especially important in process capability determination in a contractual situation where competing or collaborating suppliers are being compared.

In general terms, an assessment meeting the requirements of this standard is one which:

- is conducted by an assessor, or a team containing an assessor, qualified as described in part 6 of this International Standard;
- uses an assessment process that at minimum meets the requirements specified in part 3 of this International Standard;
- is based on a set of practices that at minimum include those defined in part 2 of this International Standard for the processes assessed;
- uses an assessment instrument meeting the requirements of part 5 of this International Standard;
- utilizes the process rating scheme defined in part 3 of this International Standard; and
- has objective evidence retained that demonstrates that the above conditions have been met.

It is the responsibility of the assessment team, and specifically the qualified assessor, to ensure that the requirements for conducting an assessment are met. The sponsor of the assessment will normally be the party requiring that the assessment conforms to the requirements.

5.1.2 Assessment team membership

When a decision is taken to perform an assessment, the sponsor of the assessment should be responsible for assembling the assessment inputs, as described in part 7 or part 8 of this International Standard. In particular, the assessment input includes the nomination of an assessor qualified as described in part 6 (the qualified assessor).

The qualifications of an assessor may be verified by following the procedure defined in part 6. This may be performed by the organization being assessed; by the employer of the assessor (if different); or by a third party. Details of the verification should be available, on request, to the sponsor of the assessment. Sufficient records of the assessor's personal history should be retained.

5.1.3 The assessment process

When performing an assessment, the team is responsible for ensuring that the requirements for this process are followed and documented. Specific requirements cover documenting the

- assessment purpose;
- assessment scope;
- assessment constraints;
- assessment responsibilities;
- any extended processes used;
- any additional information to be collected for process improvement or capability determination.

Some of the items - particularly assessment scope and constraints - contain a number of elements, and care must be taken to ensure that all of these have been addressed.

The requirements for rating are contained in part 3 of this International Standard. Guidance on reviewing the inputs of an assessment is contained in part 4. The assembly of the inputs themselves will depend on the purpose of the assessment, and guidance is contained in parts 7 and 8. Documentation of the assessment inputs should be retained in the assessment record, and traceability provided to ensure that it can be verified that the necessary reviews have been performed.

5.1.4 Selecting processes for assessment

The Assessment scope sets out which of the organizational unit’s processes are to be assessed, and their mapping into the corresponding processes as defined in part 2 of this International Standard. The mapping will normally be a task for the qualified assessor. Defining the scope in this way ensures that there is a common basis for rating and measurement.

The requirements for identifying processes and process instances and for selecting process instances for the assessment are set out in part 3 of this International Standard. Guidance on mapping to the defined processes, and for rating the process instances is contained in part 4 of this International Standard.

Apart from guiding the assessment, the mapping of organizational processes to the model defined in part 2 of this International Standard forms an essential part of the assessment record. It should be possible, after the assessment has been completed, for any person examining the assessment record to be able to relate the mapping to records of organizational structure, procedures, and standards within the assessed organization.

5.1.5 Reporting assessment results

One of the main reasons for conducting a conformant assessment is to ensure comparability with other assessment outputs. This is made possible by the requirements for rating processes and calculating results within the measurement framework, and reporting them in a way that makes the results of the calculation obvious.

Part 3 of this International Standard defines the requirements for conducting ratings, and for calculating practice adequacy ratings, process capability ratings and process profiles. Detailed guidance on deriving these ratings is contained in part 4 of this International Standard. Requirements for recording the assessment outputs are contained in part 3 of this International Standard. It should be noted that this International Standard does not mandate any specific format for the process profile; a variety of numerical or graphical presentation formats would meet the requirements.

Whatever the final format of the process profile, it is essential that clear traceability to the practices and processes contained in the model defined in part 2 of this International Standard is provided, to enable the process of calculation to be verified. In addition, it should be noted that assessment output contains full details of the process context in the assessment record. This record will also include additional information collected as part of the assessment, and required as inputs to the process improvement or process capability determination activities to follow on from assessment.

5.2 Extensions to the baseline practices

Part 2 of this International Standard provides a model for assessing and improving processes. These processes identify critical attributes that a process should have to be considered complete and effective, but without unduly constraining the implementation of the process. Further guidance on implementing processes may be found in related software standards such as ISO/IEC 12207-1 or ISO 9000-3.

Variant process models may be built that address the unique needs of an industry sector or organization by selecting specific processes from the model in part 2, providing guidance on how to interpret the practices, and/or developing extended processes. Extended processes may include additional base practices, guidance on how to interpret practices for adequacy, or be an entirely new process.

Extended processes and variant models may be developed by organizations for their own internal use; by acquirers of software systems for use in specific acquisition situations; or by professional organizations defining requirements for specific application domains or use situations.

The requirements for building conformant variants and extended processes are set out in Annex A of part 2 of this International Standard together with a style guide in Annex G. The essence of the requirements is to allow only variants that are equivalent to or extend the process model in part 2, and to provide traceability of the base practices of variants and extended processes to the base practices in part 2.

Documentation of the extended process or variant and its differences from the model in part 2 is essential for assessment and rating, and for demonstrating conformance to the requirements. New processes and new practices in extended processes should be identified. The assessment results generated for an extended process should explicitly identify the variations from the standard model. In an individual assessment, the team leader should document any extended processes used, and refer to the location of evidence of their conformance.

5.3 Constructing and selecting an assessment Instrument

An assessment instrument is used during an assessment to assist assessors in identifying work products, practices and processes, in making consistent judgements, and in analysing and presenting the results. The instrument may be in the form of a simple questionnaire, or it may be an automated tool. The instrument for a given assessment might be constructed specifically by the assessment team; it might be a tailored version of an available tool; or it might be a commercial product. The minimum set of requirements to be met by an assessment instrument of any type are defined in part 5 of this International Standard.

It is the responsibility of the qualified assessor to ensure that the assessment instrument chosen meets the requirements expressed in the part 5, and to document the relevant evidence demonstrating conformance. Documentation may consist of point-by-point analysis of the chosen tool against the requirements in the part 5; alternatively, it may be that the supplier of a tool provides certification of conformance.

Documentation of the conformance of the assessment instrument to the requirements forms part of the records of the assessment. Where software-based tools are developed or used to support assessments, ISO 12119 - 1995 may provide a useful mechanism for demonstrating or verifying their conformance to the requirements of part 5 of this International Standard.

Annex A (informative)

References

These references provide background information on the theoretical and practical applications of software process assessment. They are for information purposes only, and should not be taken as implying support for any or all of the approaches described. The list of references is limited to material that has been published officially and is available widely.

1. Craigmyle, M., and I. Fletcher, "Improving IT effectiveness through software process assessment", *Software Quality Journal*, Vol. 2, pp 257-264 (1993).
2. Humphrey, W.S., *Managing the Software Process*, Addison Wesley, 1989.
3. Kuvaja, P., Simila, J., Krzanik, L., Bicego, A., Koch, G. and Saukkonen, S., *Software Process Assessment and Improvement: The BOOTSTRAP Approach*. Blackwell, 1994.
4. Mackie, C.A. and Rigby, P.J., "Practical experience in assessing the health of the software process", *Software Quality Journal*, Vol. 2, pp 265-275, 1993.
5. Paulk, M.C., Curtis, B., Chrissis, M.B. and Weber, C.V. "Capability Maturity Model, Version 1.1," *IEEE Software*, Vol. 10, No. 4, July 1993, pp. 18-27.
6. ISO 9001 - 1994, *Model for quality assurance in design, development, production, installation and servicing.*
7. ISO 9000-3 - 1991, *Quality management and quality assurance standards - Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software.*
8. ISO/IEC 9126 - 1991, *Software quality characteristics.*
9. ISO/IEC12207-1 - 1994, *Software life cycle process*es.
10. ISO/IEC12119 -1 995, *Software products - Evaluation and test.*

Consolidated product

Software Process Assessment – Part 2 : A model for process management Version 1.00

(Formerly BPG 1.01)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout

Members: Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : ***Concepts and introductory guide***

Product Manager: Terry Rout

– Part 2 : ***A model for process management***

Product Managers: Al Graydon, Mark Paulk

– Part 3 : ***Rating processes***

Product Manager: Harry Barker

– Part 4 : ***Guide to conducting assessment***

Product Manager: Harry Barker

– Part 5 : ***Construction, selection and use of assessment instruments and tools***

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : ***Qualification and training of assessors***

Product Manager: Ron Meegoda

– Part 7 : ***Guide for use in process improvement***

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : ***Guide for use in determining supplier process capability***

Product Manager: John Hamilton

– Part 9 : ***Vocabulary***

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards).

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 2

A model for process management

Contents

Annexes

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialing to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being coordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

In this part of the International Standard (Part 2) Annex A is normative and Annexes B to H are informative.

Introduction

The purpose of this part of the International Standard is to document the set of practices fundamental to good software engineering, which can be used by the other parts of this International Standard. The model contained in this document describes processes that an organization may perform to acquire, supply, develop, operate, evolve and support software and the generic practices that characterize the capability of those processes.

A set of practices forms the lowest level of the architecture. The architecture organizes the practices into a number of categories using two different approaches (see Annex C for details). The architecture distinguishes between:

- base practices which are the essential activities of a specific process, grouped into processes and process categories by the type of activity they address;
- generic practices, applicable to any process, which represent the activities necessary to manage a process and improve its capability to perform.

The model categorizes processes into five process categories.

The **Customer-Supplier** process category consists of processes that directly impact the customer, support development and transition of the software to the customer, and provide for its correct operation and use.

The **Engineering** process category consists of processes that directly specify, implement, or maintain a system and software product and its user documentation.

The **Project** process category consists of processes which establish the project, and co-ordinate and manage its resources to produce a product or provide a service which satisfies the customer.

The **Support** process category consists of processes which enable and support the performance of the other processes on a project.

The **Organization** process category consists of processes which establish the business goals of the organization and develop process, product, and resource assets which will help the organization achieve its business goals.

Each process in the model is described in terms of base practices, which are its unique software engineering or management activities. Process categories, processes, and base practices provide a grouping by type of activity. These processes and activities characterize performance of a process, even if that performance is not systematic. Performance of the base practices may be ad hoc, unpredictable, inconsistent, poorly planned, and/or result in poor quality products, but those work products are at least marginally usable in achieving the purpose of the process. Implementing only the base practices of a process may be of minimal value and represents only the first step in building process capability, but the base practices represent the unique, functional activities of the process when instantiated in a particular environment.

Evolving process capability is expressed in terms of capability levels, common features, and generic practices. A capability level is a set of common features (sets of activities) that work together to provide a major enhancement in the capability to perform a process. Each level provides a major enhancement in capability to that provided by its predecessors in the performance of a process. They constitute a rational way of progressing through the generic practices.

Capability levels provide two benefits: they acknowledge dependencies among the practices of a process, and they help an organization identify which improvements it might perform first, based on a plausible sequence of process implementation. There are six capability levels in the model.

Level 0; *Not-Performed:* There is general failure to perform the base practices in the process. There are no easily identifiable work products or outputs of the process.

Level 1; *Performed-Informally:* Base practices of the process are generally performed. The performance of these base practices may not be rigorously planned and tracked. Performance depends on individual knowledge and effort. Work products of the process testify to the performance. Individuals within the organization recognize that an action should be performed, and there is general agreement that this action is performed as and when required. There are identifiable work products for the process.

Level 2; *Planned-and-Tracked:* Performance of the base practices in the process is planned and tracked. Performance according to specified procedures is verified. Work products conform to specified standards and requirements.

The primary distinction from the Performed-Informally Level is that the performance of the process is planned and managed and progressing towards a well-defined process

Level 3; *Well-Defined:* Base practices are performed according to a well-defined process using approved, tailored versions of standard, documented processes.

The primary distinction from the Planned-and-Tracked Level is that the process of the Well-Defined Level is planned and managed using an organization-wide standard process.

Level 4; *Quantitatively-Controlled:* Detailed measures of performance are collected and analyzed. This leads to a quantitative understanding of process capability and an improved ability to predict performance. Performance is objectively managed. The quality of work products is quantitatively known.

The primary distinction from the Well-Defined Level is that the defined process is quantitatively understood and controlled.

Level 5; *Continuously-Improving:* Quantitative process effectiveness and efficiency goals (targets) for performance are established, based on the business goals of the organization. Continuous process improvement against these goals is enabled by quantitative feedback from performing the defined processes and from piloting innovative ideas and technologies.

The primary distinction from the Quantitatively-Controlled Level is that the defined process and the standard process undergo continuous refinement and improvement, based on a quantitative understanding of the impact of changes to these processes.

A common feature in the model is a set of generic practices that address the same aspect of process implementation or institutionalization. A generic practice is an implementation or institutionalization practice (activity) that enhances the capability to perform any process. The generic practices characterize good process management that results in an increasing process capability for any process. A planned, well-defined, measured, and continuously improving process is consistently performed as the generic practices are implemented for a process. This process capability is built on the foundation of the base practices that describe the unique, functional activities of the process.

Table 1 shows the main audiences for this part of the International Standard, why each group needs the model, and how and when it will be used.

Table 1. Use of this model.

Who	Why	How	When
Software organization	Understand what to do to improve software processes	As a working guide to software project management on processes and practices to implement	During the implementation of the organization's software processes
		As a reference guide to highlight process and practices considerations	During the development/review of the organization's software processes and as a part of continuous improvement activities.
		As a training document	Same as above
	Understand which processes and practices an assessor may evaluate	As a practice checklist	Prior to an assessment
Software process assessors	Determine how an organization manages software processes and their results	As a practice checklist	Prior to and during a software process assessment

Within this part of the International Standard:

- clause 4 defines the nomenclature used to identify processes and practices and classify them within the architecture of this model;
- clause 5 defines the capability levels, common features, and generic practices that describe the capability of the processes listed in clause 6;
- clause 6 categorizes processes into five process categories and describes each process in terms of its unique base practices;
- annex A (normative) contains the requirements for creating an extended process when tailoring this part of the International Standard to the unique needs of an industry sector or organization;
- annex B provides a helpful summary list of the generic practices and base practices within the model;
- annex C describes the architecture of the model graphically and in detail.
- annexes D, E, F, and G provide detailed guidance on understanding the intent of the processes and practices in this model by, respectively:
  - providing guidance on how to interpret the generic practices when applied to a specific process;
  - mapping the processes in this model to ISO 12207;
  - mapping the processes in this model to ISO 9001; and
  - listing the sources for the processes and practices referenced in this document;
- annex H contains a style guide for constructing extended processes.

1 Scope

This part of the International Standard defines a model for software processes and practices that forms the basis for software process assessment. The model defines, at a high level, the fundamental activities (practices) that are essential to good software engineering. It describes what activities are required, not how they are to be implemented. This model for processes and process management is applicable to all software organizations and does not presume particular organizational structures, management philosophies, software life cycle models, software technologies, or development methodologies.

An organization may implement the practices to establish and subsequently improve its capabilities in the acquisition, supply, development, operation, evolution and support of software. The architecture of this model organizes the practices to help software personnel understand and use them for continuous improvement of the management of software processes.

During a software process assessment, the architecture helps the assessor to make judgements about the organization’s processes.

Evolving process capability is expressed in terms of capability levels, common features, and generic practices. Generic practices are applicable to all process. They represent the activities necessary to manage a process and improve its capability to achieve desired outputs. They are grouped into common features and, in turn, into capability levels which help define how effective the process will be at achieving its defined purpose. There are six capability levels in the model.

Each process in the model is described by base practices, which are the essential activities of the specific process. Processes are grouped in turn into five process categories.

The base practices may be extended through the generation of application or sector specific practice guides to take account of specific industry, sector, or other requirements.

2 Normative references

There are no normative references in this part og the International Standard

3 Definitions

For the purposes of this part of the International Standard, the definitions in *Software Process Assessment – Part 9: Vocabulary* apply.

4 Nomenclature

4.1 General

A nomenclature for practices is defined in order to identify them unambiguously and relate them to the architecture of the model. The nomenclature for base practices facilitates the identification of process categories, the process that belong to each process category, and the base practices that belong to each process. For generic practices, the nomenclature facilitates the identification of capability levels, the common features that belong to each capability level, and the generic practices that belong to each common feature. In software process assessments using the process model, the nomenclature and identifiers contained in this model shall be used to identify processes to be assessed and to identify the ratings in the assessment output (see part 3 of this International Standard). The nomenclature shall also be used in constructing extended processes and variant models (see Annex A)

4.2 Definition

Each practice is assigned an identifier consisting of a three-character alphanumeric code.

For a base practice, the identifier is of the form PC.PR.PT.

For a generic practice, the identifier is of the form CL.CF.PT.

The codes are:

PC	process category identifier
PR	process number (within the process category)
CL	capability level number
CF	common feature number (within the capability level)
PT	practice number (within the process or common feature)

For example, “ENG.3.1” denotes a base practice in process category ENG (Engineering), process 3 (Develop software design), base practice 1 “Develop software architectural design”.

Similarly, “2.3.1” denotes a generic practice in capability level 2 (Planned-and-Tracked), common feature 3 (Verifying Performance), generic practice 1 “Verify process compliance”.

5 Generic practices

During an assessment, generic practices, grouped according to common feature and capability level, are used to determine the capability of a process.

The generic practices, defined in the following tables, shall apply to all processes defined within this part of the International Standard.

Table 2 – Generic Practices for Level 1: Performed-Informally

	Common Feature 1.1: Performing Base Practices
1.1.1	**Perform the process.** Perform the base practices of the process to provide work products and/or services to a customer. Note: The customer of the process may be internal or external to the organization.

Table 3. – Generic Practices for Level 2: Planned-and-Tracked

	Common Feature 2.1: Planning Performance
2.1.1	**Allocate resources.** Allocate adequate resources (including people) for performing the process. Note: Resource management is described in project process, "Manage resources and schedule" PRO.7.
2.1.2	**Assign responsibilities.** Assign responsibilities for developing the work products and/or providing the services of the process. Note: Assigning responsibility does not necessarily entail detailed job descriptions. Responsibility could be assigned via living documents, such as a task plan. Dynamic assignment of roles is another legitimate implementation of this practice, so long as there are mechanisms in place to assure that the responsibility is assumed.
2.1.3	**Document the process.** Document the approach to performing the process in standards and/or procedures. Note: Employee participation in developing standards and procedures is essential to creating a usable process definition.
2.1.4	**Provide tools.** Provide appropriate tools to support performance of the process.
2.1.5	**Ensure training.** Ensure that the individuals performing the process are appropriately trained in how to perform the process.
2.1.6	**Plan the process.** Plan the performance of the process. Note: Project planning is described in the process, "Establish project plan" PRO.2. Plans are based on estimates, which implies the use of software measurements. See 2.4.1 “Track with measurement” for the related tracking practice.

Table 3. (concluded) **– Generic Practices for Level 2: Planned-and-Tracked**

	Common Feature 2.2: Disciplined Performance
2.2.1	**Use plans, standards, and procedures.** Use documented plans, standards, and/or procedures in implementing the process. Note: The standards and procedures used were documented in 2.1.3. The plans used were documented in 2.1.6.
2.2.2	**Do configuration management.** Place work products of the process under version control or configuration management, as appropriate. Note: The process for configuration management is described in support process, "Perform configuration management" SUP.2.
	Common Feature 2.3: Verifying Performance
2.3.1	**Verify process compliance.** Verify compliance of the process with applicable standards and/or procedures. Note: The applicable standards and procedures were documented in 2.1.3 and used in 2.2.1. The quality assurance process is described in support process, "Perform quality assurance" SUP.3.
2.3.2	**Audit work products.** Verify compliance of work products with the applicable standards and/or requirements. Note: Requirements are identified in Customer-Supplier process, "Identify customer needs" CUS.3, and managed in project process, "Manage requirements" PRO.4.
	Common Feature 2.4: Tracking Performance
2.4.1	**Track with measurement.** Track the status of the process against the plan using measurement. Note: The use of measurement implies that the measures have been defined and selected, and data has been collected.
2.4.2	**Take corrective action.** Take corrective action as appropriate when progress varies significantly from that planned. Note: Progress may vary because estimates were inaccurate, performance was affected by external factors, or the requirements, on which the plan was based, have changed. Corrective action may involve changing the process or changing the plan or both.

Table 4 – Generic Practices for Level 3: Well-Defined

	Common Feature 3.1: Defining a Standard Process **(Organization-Level Common Feature)**
3.1.1	**Standardize the process.** Document a standard process or family of processes for the organization, which describes how to implement the base practices for the process. Note: The critical distinction between 2.1.3 and 3.1.1 is the scope of application of the standards and procedures. In 2.1.3, the standards and procedures may be used in only a specific instance of the process, e.g., in a particular project. In 3.1.1, standards and procedures are being established at an organizational level for common use. The process definition process is described in the Organization process category, "Define the process" ORG.2.
3.1.2	**Tailor the standard process.** Tailor the organization's standard process family to create a defined process which addresses the particular needs of a specific use. Note: The phrase “which addresses the particular needs of a specific use” caters to the general case of organization-level, as opposed to project-level, processes. For defined processes at the project level, the tailoring addresses the particular needs of the project. The organization's standard process family is documented in 3.1.1 and ORG.2.
	Common Feature 3.2: Performing the Defined Process
3.2.1	**Use a well-defined process.** Use a well-defined process in implementing the process. Note: The defined process will typically be tailored from the organization’s standard process, as described in 3.1.2. A well-defined process is one with inputs, entry criteria, tasks, validation, outputs, and exit criteria that are documented, consistent, and complete.
3.2.2	**Perform peer reviews.** Perform peer reviews of appropriate work products of the process. Note: The process for peer reviews is described in support process, "Perform peer reviews" SUP.5.
3.2.3	**Use well-defined data.** Use data on performing the defined process to manage the defined process. Note: This is an evolution of 2.4.2; corrective action taken here is based on a well-defined process, which has objective completion (exit) criteria (see 3.2.1) for determining progress.

Table 5 – Generic Practices for Level 4: Quantitatively-Controlled

	Common Feature 4.1: Establishing Measurable Quality Goals
4.1.1	**Establish quality goals.** Establish measurable quality goals for the work products of the organization's standard process family. Note: These quality goals should be tied to the strategic quality goals of the organization, the particular needs and priorities of the customer, and/or to the tactical needs of the project.
	Common Feature 4.2: Objectively Managing Performance
4.2.1	**Determine process capability.** Determine the process capability of the defined process quantitatively. Note: This is a quantitative process capability based on a well-defined (3.1.1) and measured process.
4.2.2	**Use process capability.** Take corrective action as appropriate when the process is not performing within its process capability. Note: Special causes of variation, identified based on an understanding of process capability, are used to understand when and what kind of corrective action is appropriate. This is an evolution of 3.2.3, with the addition of quantitative process capability to the defined process.

Table 6 – Generic Practices for Level 5: Continuously-Improving

	Common Feature 5.1: Improving Organizational Capability (Organization-Level Common Feature)
5.1.1	**Establish process effectiveness goals.** Establish quantitative goals for improving process effectiveness of the standard process family, based on the business goals of the organization and the current process capability.
5.1.2	**Continuously improve the standard process.** Continuously improve the process by changing the organization's standard process family to increase its effectiveness. Note: Changes to the organization's standard process family may come from innovations in technology or incremental improvements. Innovative improvements will usually be externally driven by new technologies. Incremental improvements will usually be internally driven by improvements identified during tailoring (3.1.2) or defect prevention (5.2.2) activities. The approach to improving the standard process is to attack common causes of variation. Special causes of variation are controlled in 4.2.2. Common causes of variation across instantiations of the process are changed in this practice.
	Common Feature 5.2: Improving Process Effectiveness
5.2.1	**Perform causal analysis.** Perform causal analysis of defects.
5.2.2	**Eliminate defect causes.** Eliminate the causes of defects in the defined process selectively. Note: Defect causes are selectively eliminated because it may be impractical to perform causal analysis (5.2.1) on all defects, so some screening criteria may be used. These criteria for prioritizing defect removal should be identified and documented. Also note the desirability of eliminating similar, as yet undiscovered, defects in the product, as well as eliminating the cause of the defect.
5.2.3	**Continuously improve the defined process.** Continuously improve process performance by changing the defined process to increase its effectiveness. Note: The improvements may be based on incremental improvements (5.2.2) or new technologies (perhaps as part of pilot testing in ORG.3.8.) Improvements will typically be driven by the goals established in 5.1.1.

6 Base practices

In an assessment conducted according to the provisions of this International Standard, the processes included within the scope of the assessment shall be mapped to one or more of the processes defined in this clause, or to a variant model constructed and documented in accordance with the requirements in Annex A..

The assessment shall include all of the base practices of each process within the scope of the assessment.

The rest of this clause lists all the base practices organized as a hierarchy, first by process category, then by process. All process categories, processes, and base practices are given both a name and description.

The five process categories are:

CUS	**Customer - Supplier**
ENG	**Engineering**
PRO	**Project**
SUP	**Support**
ORG	**Organization**

The description of each process category includes a characterization of the processes it contains, followed by a list of process names.

The first paragraph in the description of each process states the purpose of the process. Subsequent paragraphs, if present, clarify what is said in the purpose, describe inputs/outputs to other processes, and describe when the process is invoked.

Base practices follow the process description. Each base practice is numbered (see clause 4) to allow easy identification and reference. This number and name of the base practice appear in bold. Immediately following is a statement of what the practice does. This is sometimes followed by a note which provides an example, a more detailed explanation, or a cross-reference note.

6.1 Customer-Supplier process category (CUS)

The *Customer-Supplier* process category consists of processes that directly impact the customer, supporting development and transition of the software to the customer, and provide for its correct operation and use.

Note: Throughout the base practices, "customer" can mean either an external customer or an internal customer.

The processes belonging to the Customer-Supplier process category are:

CUS.1 Acquire software product and/or service

CUS.2 Establish contract

CUS.3 Identify customer needs

CUS.4 Perform joint audits and reviews

CUS.5 Package, deliver, and install the software

CUS.6 Support operation of software

CUS.7 Provide customer service

CUS.8 Assess customer satisfaction

CUS.1 Acquire software product and/or service

The purpose of the *Acquire software product and/or service* process is to define the activities that must be performed by the customer or the acquirer to obtain the software product or service. The acquirer is the party that obtains the software product and/or service from the supplier. In some cases the acquirer and the customer (the party who will utilize the software product and/or service) may be the same party. In other cases, where there exists a separate party (organization) that is solely performing the acquisition duties, the acquirer and the customer will not be the same party.

The activities involved in this process include the definition of the need to acquire a software product and/or service through to the proposal, supplier selection, and product and/or service acceptance.

Note: This process would typically be applied in an assessment when the software organization being assessed is acting as a customer to a software subcontractor or vendor who is supplying it with software products or services.

CUS.1.1 Identify the need. Identify a need to acquire, develop, or enhance a software product.

Note: The need may be necessitated by a number of circumstances including: business, regulatory, research, safety, security.

CUS.1.2 Define the requirements. Prepare the system and software requirements to satisfy the need for a new product and/or service.

Note: This definition of the requirements may be done completely or partially by the supplier. See "Develop system requirements and design" ENG.1, and "Develop software requirements" ENG.2.

Also see "Obtain customer requirements and requests" CUS.3.1. CUS.1.2 is focusing on defining requirements when the software organization is acting as a customer. CUS.3.1 is focusing on obtaining requirements when the software organization is acting as a supplier. The primary difference is one of perspective, depending on the role being performed.

CUS.1.3 Prepare acquisition strategy. Prepare a strategy for the acquisition of the product including:

- make/buy risk analysis (off-the-shelf, develop internally, develop through contract, enhance existing software product);
- acceptance strategy.

CUS.1.4 Prepare request for proposal. Prepare a request for proposal tender including acquisition requirements and project schedule.

CUS.1.5 Select software product supplier. Select a supplier for the acquired software product and/or service based upon an evaluation of supplier proposals, capabilities and other factors which may be particular to the product.

Note: After the supplier has been chosen, a contract is established between the customer and the supplier. See process, "Establish contract" CUS.2. Monitoring of the supplier and acceptance of the product are performed through the process, "Perform joint audits and reviews" CUS.4. Management of subcontracted work is performed by "Manage subcontractors" PRO.8.

CUS.2 Establish contract

The purpose of the *Establish contract* process is to develop a contract which clearly expresses the expectations, responsibilities, and liabilities of both the supplier and the customer.

Note: Process, "Perform joint audits and reviews" CUS.4 addresses joint audits and reviews of this contract.

CUS.2.1 Review before contract finalization. Review the contents of the contract prior to its finalization.

Note: This review typically includes:

- scope of contract and requirements;
- possible contingencies or risks;
- alignment of the contract with the strategic business plan of the organization;
- protection of proprietary information;
- requirements which differ from those in the original documentation;
- capability to meet contractual requirements;
- responsibility for subcontracted work;
- terminology;
- the customer ability to meet contractual obligations.

CUS.2.2 Negotiate contract. Negotiate a contract with the customer.

Note: This contract typically includes

- schedules for product delivery;
- terms of payment;
- customer's acceptance criteria;
- procedures for handling changes in customer requirements;
- procedures for handling customer requests for process/product quality monitoring;
- procedures for handling customer-detected problems;
- customer's role in the development and maintenance process;
- resources to be provided by the customer;
- standards and procedures to be used;
- servicing and maintenance requirements.

CUS.2.3 Determine interfaces to independent agents. Determine the supplier and customer interfaces to independent verification, validation, and/or test agents, and document in the contract.

CUS.2.4 Determine interfaces to subcontractors. Determine the supplier and customer interfaces to other parties, such as subcontractors, who will be involved in the work described in the contract, or whose work will impact its success; document in the contract.

CUS.3 Identify customer needs

The purpose of the *Identify customer needs* process is to manage the gathering, processing, and tracking of customer requirements and requests toward a better understanding of what will satisfy the customer.

Compare with process, "Manage requirements" PRO.4, which addresses coming to an understanding within the software project of the requirements to build to (and which become part of the development baseline). Instead, the current process emphasizes interaction with the customer to promote an understanding of customer requirements, and tracks all requirements and requests.

CUS.3.1 Obtain customer requirements and requests. Obtain customer requirements and requests through direct solicitation of customer and user input and through review of: customer business proposals, target hardware environment, and other documents bearing on customer requirements.

CUS.3.2 Understand customer expectations. Review with customers and users their requirements and requests to better understand their needs and expectations.

Note: This may include joint meetings with the customer and as appropriate, models and prototypes. The performance of this base practice may coincide with base practice, "Evaluate requirements with customer" ENG.2.4.

CUS.3.3 Keep customers informed. Keep customers informed about the status and disposition of their requirements and requests.

Note: This may include joint meetings with the customer or formal communication to review the status for their requests including requirements.

CUS.4 Perform joint audits and reviews

The purpose of the *Perform joint audits and reviews* process is to maintain a common understanding with the customer of the progress against the objectives of the contract and what should be done to help ensure development of a product that satisfies the customer.

The work of this process is accomplished through different kinds of audits and reviews, including: contract audits, management reviews, technical reviews, and the acceptance review. Many of these reviews will be scheduled to coincide with development milestones and other milestones specified in the contract. The process "Establish contract" CUS.2, deals with contract reviews.

CUS.4.1 Establish joint reviews and audits. Establish which joint reviews and audits will be performed with the customer.

CUS.4.2 Prepare for customer audits and reviews. Prepare for customer audits and reviews by establishing

- scope of review;
- a checklist for the review;
- the desired outputs;
- a schedule;
- who should attend;
- an approach to problem identification and resolution;
- facility needs.

CUS.4.3 Conduct joint management reviews. Conduct regular joint management reviews with the customer to discuss and assess

- proposal against the requirements;
- status against project plans;
- schedules;
- risks;
- compliance with appropriate standards;
- readiness for the next development steps.

CUS.4.4 Conduct joint technical reviews. Conduct regular joint technical reviews with the customer to discuss technical issues and assess technical status against customer requirements and acceptance criteria documented in the contract.

CUS.4.5 Support customer acceptance review. Support the customer in his evaluation of the software product, providing evidence that the software product is complete and correct, complies with appropriate standards and specifications, and satisfies the acceptance criteria documented in the contract.

Note: This practice may include acceptance testing by the customer.

CUS.4.6 Perform joint process assessment. Conduct regular joint software process assessments with the customer to jointly review both the organization's processes along with interfaces with the customer processes.

CUS.5 Package, deliver, and install the software

The purpose of the *Package, deliver, and install the software* process is to package, deliver, and install the software at the customer site to ensure its effective operation, handling, and storage. The base practices in this process category are critical to preserving the quality of the software and all associated deliverables with the software.

CUS.5.1 Identify installation requirements. Identify requirements for packing, delivering, and installing the software, addressing as appropriate

- type of media;
- documentation;
- copyrights and licensing;
- custody of master and backup copies;
- provision for copying;
- critical safety and security issues.

CUS.5.2 Prepare site for installation. Prepare the site for the installation of the software product.

Note: This may be performed by the customer.

CUS.5.3 Pack software. Label and pack the software and its accompanying documentation, including a packing list identifying the contents along with information such as what is new or changed in the software.

Note: The software product will be created through base practice, "Build product releases" SUP.2.6 and will typically identify release/version numbers.

CUS.5.4 Deliver software. Deliver the software, excluding any non-deliverable items used during its development or maintenance.

CUS.5.5 Verify correct receipt. Verify the correctness and completeness of the delivered software, including the released package, delivery instructions and associated documentation.

Note: This base practice will be usually carried out with or by the customer.

CUS.5.6 Install software. Install the software at the customer site, recording the steps taken and the results.

CUS.5.7 Provide handling and storage procedures. Define and provide procedures for handling and storing the software and its documentation including

- providing for master copies of code and documentation;
- disaster recovery;
- addressing appropriate critical safety and security issues.

CUS.6 Support operation of software

The purpose of the *Support operation of software* process is to support the correct and effective operation of the software for the duration of its intended operation.

This process assumes that the software has already been installed (see process, "Package, deliver, and install the software" CUS.5) and is operating as part of a larger system, with an operator responsible for ensuring continuing operation of the system, and users who use the system.

The management of changes to the software and system to support the software operation will be accomplished by process, "Maintain system and software" ENG.7.

One possible input to this process would be as the quality requirements for the software operation determined by base practice, "Determine software requirements" ENG.2.1.

Note: Some practices may be performed by a different party. For example, the operator may be the customer, the developer, or a third party (see, "Operate the software" CUS.6.3). The operator is the individual/organization whose responsibility it is to ensure operation of the software system. The user is the individual/organization who utilizes the software system to perform its tasks.

CUS.6.1 Identify operational risks. Identify and mitigate risks to system operation and functionality that are due to such factors as: environmental failure, hardware or software failure, or network failure.

CUS.6.2 Perform operational testing. Perform operational testing of each release of the software, assessing satisfaction against specified criteria.

CUS.6.3 Operate the software. Operate the software in its intended environment and in the specified way.

CUS.6.4 Resolve operational problems. Identify, record, and resolve problems arising from operation of the software (i.e. problems encountered by the operator as opposed to a user).

Note: The elimination of the cause of the operational problem will be handled by base practice, "Correct the defect" SUP.4.6.

CUS.6.5 Handle user requests. Monitor, record, and respond to all user requests and problems relating to the software, forwarding as appropriate to the maintenance function.

Note: This ties to process, "Maintain system and software" ENG.7.

CUS.6.6 Document temporary work-arounds. Provide documented temporary work-arounds as appropriate to maintain operation of the system until a permanent solution to a problem can be found.

Note: Keep the customer informed of the status and availability of the permanent solution.

CUS.6.7 Monitor system capacity and service. Provide the capability to monitor system capacity and operational service on a regular basis.

CUS.7 Provide customer service

The purpose of the *Provide customer service* process is to establish and maintain an acceptable level of service to the customer to support effective use of the software.

CUS.7.1 Train customer. Provide training and documentation, as appropriate, to the customer so that the software can be effectively used.

CUS.7.2 Establish product support. Establish a service by which the customer can raise problems and questions encountered in use of the software, and receive help in resolving them.

CUS.7.3 Monitor performance. Monitor the software performance in order to be aware of performance problems which might impact level of service.

CUS.7.4 Install product upgrades. Plan, test, and install modifications to the software and the documentation to correct defects or improve performance of the software for the customer, and thus maintain or improve the level of service.

Note: This ties to process, "Maintain system and software" ENG.7 and process, "Package, deliver, and install the software" CUS.5.

CUS.8 Assess customer satisfaction

The purpose of the *Assess customer satisfaction* process is to determine the level of customer satisfaction with the software and services received (operation, customer support).

CUS.8.1 Determine customer satisfaction level. Determine the level of customer satisfaction with the software products and services received through, as appropriate, field performance data, surveys, interviews, and studies.

Note: In some instances the end-user of the software may be different from the customer of the software. In this case, both the customer and end-user satisfaction levels should be determined.

CUS.8.2 Compare with competitors. Compare the level of customer satisfaction obtained for your software and services received relative to that of your competitors.

Note: It may be necessary to obtain information on your competitors from third party sources. It may also be necessary to include information on how competitors define customer satisfaction, measurement techniques, criteria, collection and evaluation methods, etc., to provide a meaningful comparison.

CUS.8.3 Communicate customer satisfaction. Communicate customer satisfaction data throughout the organization.

6.2 Engineering process category (ENG)

The *Engineering* process category consists of processes that directly specify, implement, or maintain a system and software product and its user documentation.

In some circumstances, there is no "system" so the scope of the engineering processes is reduced to only software and user documentation, and processes ENG.1 and ENG.6 become "not applicable."

While the processes listed below appear in "waterfall model" sequence, the intent is not to preclude either their concurrent or iterative execution. (The sequence is determined and documented by base practice, "Determine release strategy" ENG.1.4 and by process, "Plan project life cycle" PRO.1.)

Inputs to the "Engineering" process category possibly include a contract or agreement describing what work is to be done, and a plan(s) on how that is to be accomplished (see processes, "Establish contract" CUS.2, and "Establish project plan" PRO.2.)

The processes belonging to the "Engineering" process category are:

ENG.1 Develop system requirements and design

ENG.2 Develop software requirements

ENG.3 Develop software design

ENG.4 Implement software design

ENG.5 Integrate and test software

ENG.6 Integrate and test system

ENG.7 Maintain system and software

ENG.1 Develop system requirements and design

The purpose of the *Develop system requirements and design* process is to establish the system requirements and system design, identifying which system-level requirements should be allocated to which elements of the system design and to which releases of the system.

Note: This process will typically not be performed by a software group, but the group which does perform it should include a member(s) with software expertise.

ENG.1.1 Specify system requirements. Determine the required functions and capabilities of the system and document in a system requirements specification.

Note: the system requirements specification describes such things as

- functions and capabilities of the system;
- performance of the system;
- safety;
- reliability;
- security;
- human engineering;
- interface;
- operations, and maintenance requirements;
- design constraints and qualification requirements.

See CUS.3 for discussion of customer requirements used as an input to system requirements analysis.

ENG.1.2 Describe system architecture. Establish the top-level system architecture, identifying elements of

- hardware;
- software;
- manual operations.

ENG.1.3 Allocate requirements. Allocate all system requirements to the elements of the top-level system architecture, including software.

Note: The result of performing base practices ENG.1.2 and ENG.1.3 is a documented product configuration which describes the position of each element in the system architecture and the requirements which it must address.

ENG.1.4 Determine release strategy. Prioritize the system requirements and map them to future releases of the system.

Note: Rather than wait to release a product in which all requirements are achieved, it may instead be desirable to prioritize the system requirements and to release a sequence of products which address increasing subsets of the prioritized requirements, e.g. to establish market share early. A possible input to this base practice is the project's software life cycle model, produced by process, "Plan project life cycle" PRO.1, or the equivalent at the system level.

ENG.2 Develop software requirements

The purpose of the *Develop software requirements* process is to establish, analyze and refine the software requirements.

ENG.2.1 Determine software requirements. Determine the software requirements and document in a software requirements specification.

Note: The software requirements specification describes such things as

– functions to be performed and their performance characteristics;
– software interfaces (to hardware, operating system, and user) and their characteristics;
– reliability characteristics;
– installation and maintenance requirements;
– safety requirements;
– security characteristics.

In addition, there is value in specifying requirements, particularly quality requirements, in quantitative terms, so that an objective evaluation of their satisfaction can later be made.

ENG.2.2 Analyze software requirements. Analyze the software requirements for correctness.

Note: Aspects of correctness to analyze include

– completeness;
– understandability;
– testability;
– verifiability;
– feasibility;
– validity;
– consistency;
– adequacy of content.

Depending on the software life cycle model chosen, it may be desirable to have only a select set of requirements "correct" (implemented), leaving the others to be addressed in subsequent iterations of this process. See base practice, "Update requirements for next iteration" ENG.2.5, below.

ENG.2.3 Determine operating environment impact. Determine the impact of the software requirements on the operating environment.

Note: The operating environment includes tasks performed by or other systems used by the intended uses of the software product.

ENG.2.4 Evaluate requirements with customer. Communicate the software requirements to the customer, and revise if necessary, based on what is learned through this communication.

Note: Prototyping or simulation may be appropriate methods of evaluating the requirements with the customer. The performance of this base practice may coincide with base practice, "Understand customer expectations" CUS.3.2.

ENG.2.5 Update requirements for next iteration. After completing an iteration of requirements, design, code, and test, use the feedback obtained from use to modify the requirements for the next iteration.

ENG.3 Develop software design

The purpose of the *Develop software design* process is to establish a software design that effectively accommodates the software requirements; at the top-level this identifies the major software components and refines these into lower level software units which can be coded, compiled, and tested.

ENG.3.1 Develop software architectural design. Transform the software requirements into a software architecture that describes the top-level structure and identifies its major components.

ENG.3.2 Design interfaces at top level. Develop and document a top-level design for the external and internal interfaces.

ENG.3.3 Develop detailed design. Transform the top-level design into a detailed design for each software component. The software components are refined into lower levels containing software units that can be coded, compiled, and tested.

Note: The detailed design includes the specification of external and internal interfaces between the software units.

The result of this base practice is a documented software design document which describes the position of each software unit in the software architecture and the functional, performance, and quality characteristics which each must address.

ENG.3.4 Establish traceability. Establish traceability between the software requirements and the software designs.

ENG.4 Implement software design

The purpose of the *Implement software design* process is to produce executable and independently tested units of software code which implement the components of the software design.

ENG.4.1 Develop software units. Develop and document each software unit, including

- the code;
- data structures;
- database.

Note: This base practice involves creating, documenting, and compiling representations of each software unit using expressions in the appropriate programming language(s).

ENG.4.2 Develop unit verification procedures. Develop and document procedures for verifying that each software unit satisfies its design requirements.

Note: The normal verification procedure will be through unit testing, and the verification procedure will include unit test cases and unit test data.

ENG.4.3 Verify the software units. Verify that each software unit satisfies its design requirements and document the results.

ENG.5 Integrate and test software

The purpose of the *Integrate and test software* process is to integrate the software units with each other producing software that will satisfy the software requirements.

This process is accomplished through developing aggregates of software units and testing them as an aggregate, and then testing the resulting integrated software to ensure it satisfies the software requirements.

Note: 1) Testing is normally done by individuals or teams independent of the developers. 2) Software test planning should be started early, e.g. at the same time as developing the software requirements and design, to allow for adequate test preparation.

ENG.5.1 Determine regression test strategy. Determine the conditions for retesting aggregates against their tests should a change in a given software unit be made.

ENG.5.2 Build aggregates of software units. Identify aggregates of software units and a sequence or partial ordering for testing them.

Note: Typically, the software architecture and the release strategy will have some influence on the selection of aggregates.

ENG.5.3 Develop tests for aggregates. Describe the tests to be run against each software aggregate, indicating input data and acceptance criteria.

ENG.5.4 Test software aggregates. Test each software aggregate ensuring that it satisfies the test criteria, and document the results.

ENG.5.5 Develop tests for software. Describe the tests to be run against the integrated software, indicating software requirements being checked, input data, and acceptance criteria.

Note: Tests can be developed during process, "Develop software requirements" ENG.2, “Develop software design” ENG.3, and “Implement software design” ENG.4. Test development should not wait until software integration, covered in base practices ENG.5.2-5.4).

The set of tests should demonstrate compliance with the software requirements and provide coverage of the internal structure of the software.

ENG.5.6 Test integrated software. Test the integrated software ensuring that it satisfies the software requirements, and document the results.

ENG.6 Integrate and test system

The purpose of the *Integrate and test system* process is to integrate the software with the manual operations and hardware elements producing a system that will satisfy the system requirements.

This process is accomplished through developing aggregates of system elements and testing them as an aggregate, and then testing the resulting integrated system to ensure it satisfies the system requirements.

Note: 1) This process will typically not be performed by a software group, but the group which does perform it should include a member(s) with software expertise.

2) System test planning should be started early, e.g. about the same time as developing the system requirements and design, to allow for adequate test preparation.

ENG.6.1 Build aggregates of system elements. Identify aggregates of system elements and a sequence or partial ordering for testing them.

Note: Typically, the system architecture and the release strategy will have some influence on the selection of aggregates.

ENG.6.2 Develop tests for aggregates. Describe the tests to be run against each system aggregate, indicating input data, system components needed to perform the test, and acceptance criteria.

ENG.6.3 Test system aggregates. Test each system aggregate ensuring that it satisfies its requirements, and document the results.

ENG.6.4 Develop tests for system. Describe the tests to be run against the integrated system, indicating system requirements being checked, input data, and acceptance criteria.

Note: 1) This can be performed during process, "Develop system requirements and design" ENG.1 (it should not wait until integration, covered in base practices ENG 6.1-6.3 above).

2) The set of tests should demonstrate compliance with the system requirements.

3) For some products, it may be appropriate to conduct extensive field testing.

ENG.6.5 Test integrated system. Test the integrated system ensuring that it satisfies the system requirements, and document the results.

ENG.7 Maintain system and software

The purpose of the *Maintain system and software* process is to modify the system, its hardware, the network system (if any), software, and associated documentation in response to user requests while preserving the integrity of the system design.

There are several sources which create the need for modifying the system or software. Example sources include

- detected error;
- deficiency;
- problem in the operation of the system or software;
- particular improvement or modification of the system or software required or requested by the customer (internal or external).

Note: This process interacts closely with several customer processes and their base practices, and may be even partially subsumed by them, for example

- "Support operation of software" CUS.6;
- its base practice, "Handle user requests" CUS.6.6;
- "Provide customer service" CUS.6;
- its base practice, "Install product upgrades" CUS.7.6.

ENG.7.1 Determine maintenance requirements. Determine the system and software maintenance requirements, identifying the system and software elements to be maintained, and their required enhancements.

Note: Some of the required enhancements may have been previously planned but deferred.

ENG.7.2 Analyze user problems and enhancements. Analyze user problems and requests and required enhancements, evaluating the possible impact of different options for modifying the operational system and software, system interfaces, and requirements.

Note: Several base practices address the collection and tracking of user problems and requests

- "Handle user requests" CUS.6.6;
- "Obtain customer requirements and requests" CUS.3.1;
- base practices in process, "Perform problem resolution" SUP.4.

ENG.7.3 Determine modifications for next upgrade. Based on the above analyses, determine which modifications should be applied in the next system or software upgrade, documenting which software units and other system elements and which documentation will need to be changed and which tests will need to be run.

ENG.7.4 Implement and test modifications. Use the other engineering processes, as appropriate, to implement and test the selected modifications, demonstrating that the unmodified system and software requirements will not be compromised by the upgrade.

ENG.7.5 Upgrade user system. Migrate the upgraded system and software with applied modifications to the user's environment, providing for, as appropriate

- parallel operation of the previous and upgraded systems;
- additional user training;
- support options;
- retirement of the previous system.

6.3 Project process category (PRO)

The *Project* process category consists of processes which establish the project, and co-ordinate and manage its resources to produce a product or provide services which satisfy the customer.

The input is a contract or agreement to do the work (see process, "Establish contract" CUS.2). The focus in this process category is on the effective use of resources (time, effort, people, money) toward accomplishing the purpose and objectives of the project.

The processes belonging to the "Project" process category are:

PRO.1 Plan project life cycle

PRO.2 Establish project plan

PRO.3 Build project teams

PRO.4 Manage requirements

PRO.5 Manage quality

PRO.6 Manage risks

PRO.7 Manage resources and schedule

PRO.8 Manage subcontractors

PRO.1 Plan project life cycle

The purpose of the *Plan project life cycle* process is to establish an appropriate software life cycle model for the project.

The input to this process is an agreement or contract to develop a software product, which identifies the project objectives with regard to product quality, cost, and/or schedule.

The output of this process is a software life cycle model with descriptions of software activities and tasks to be performed by the project and identification of project controls (management and technical reviews, etc.).

PRO.1.1 Evaluate options for product development. Evaluate options for product development, identifying the risks associated with each.

Note: Example product development options include

- using internal resources;
- subcontracting;
- using off-the-shelf products;
- using customer-furnished products;
- combination of all four.

PRO.1.2 Select software life cycle model. Select a software life cycle model for the project which is appropriate to the scope, magnitude, and complexity of the project.

Note: Examples of software life cycle models include waterfall, spiral and serial build.

Note that the life cycle might be selected by the customer or by contract terms.

PRO.1.3 Describe activities and tasks. Describe the project's software activities and their associated tasks, and purpose of each.

PRO.1.4 Establish task sequence. Establish the software task sequence or partial ordering within the identified software life cycle, identifying where the following occur

- management and technical reviews;
- software audits;
- peer reviews.

PRO.1.5 Document activities. Identify and document software development activities and tasks, including inputs, outputs, and interfaces.

PRO.2 Establish project plan

The purpose of the *Establish project plan* process is to establish reasonable plans for performing the software engineering and to form a basis for managing the software project.

Project plans typically document

- project purpose and objectives;
- work products to be developed;
- software life cycle model;
- software estimates;
- project risks and mitigation plans;
- schedule;
- resources allocated to the project activities.

Inputs to this process can include the project's software life cycle model produced by process, "Plan project life cycle" PRO.1.

PRO.2.1 Develop work breakdown structure. Develop a work break down structure relating project tasks and sequence with the resources required to accomplish them.

PRO.2.2 Identify project standards. Identify the software standards which will guide the project's software activities, consistent with the needs of the project.

PRO.2.3 Identify specialized facilities. Identify any specialized tools, equipment, or rooms beyond those normally available which will be needed to meet any unusual technical requirements of the project.

Note: Examples include

- target hardware;
- simulators;
- specialized test equipment;
- test laboratories.

The process, "Provide software engineering environment" ORG.6 deals with providing the (normal) software engineering environment. Specialized facilities may be needed due to security requirements of the project.

PRO.2.4 Determine reuse strategy. Identify opportunities for reuse, analyze their impacts, and determine the strategy for reuse.

Note: This base practice benefits from process, "Enable reuse" ORG.5.

PRO.2.5 Develop project estimates. Develop estimates of what is needed to satisfy the software requirements for the entire software life cycle.

Note: Parameters to estimate include

– size;
– effort;
– cost;
– schedule;
– resources.

PRO.2.6 Identify initial project risks. Identify, assess, and document an initial (or baseline) set of project risks and their mitigation plans.

Note: Software project risks include

– risks to staying within budgeted software size, cost, effort, schedule;
– risks in availability of resources;
– technical risks.

See the process, "Manage risks" PRO.6.

PRO.2.7 Identify project measures. Identify the basic set of status and other measures which will be used to track project progress and help determine if the project is meeting its objectives.

PRO.2.8 Establish project schedule. Establish the project schedule, based on the software life cycle model, work breakdown structure, estimates, and risk mitigation plans.

PRO.2.9 Establish project commitments. Establish commitments to the estimates and plans with all affected groups.

PRO.2.10 Document project plans. Document the results of the activities in this process within the project plans.

Note: This includes documenting the project's software life cycle model.

PRO.3 Build project teams

The purpose of the *Build project teams* process is to establish project teams with qualified members who can fulfil their responsibilities on their team and work together as a cohesive group.

The term "team" is intended to cover a number of different kinds and durations of teams, including

- mixed teams consisting of both customer and supplier representatives;
- review teams, development teams, problem analysis teams, process improvement teams etc.;
- teams which exist for a short duration to solve a specific problem, or long-term as an established part of the environment.

A likely input is the software project plans from process, "Establish project plan" PRO.2.

PRO.3.1 Define project teams. Define the teams which will be needed to perform the work of the project, defining the structure and operating rules for the team, required knowledge and skills.

PRO.3.2 Empower project teams. Empower teams to perform their job, by ensuring that they have

- an understanding of their job;
- a shared vision or sense of common interest;
- appropriate mechanisms or facilities for communication and work;
- support from the appropriate management for what they are trying to accomplish;

PRO.3.3 Maintain project team interactions. Obtain and maintain agreement on the implementation of interactions between teams.

Note: Example areas on which to agree include

- responsibilities;
- commitments;
- interfaces;
- interaction methods;
- conflict resolution methods.

PRO.3.4 Manage inter-team issues. Identify, track, and resolve issues that affect the progress of more than one team or threaten project unity.

PRO.4 Manage requirements

The purpose of the *Manage requirements* process is to establish a software requirements baseline, which serves as the basis for the project's software work, products, and activities; and manage changes to that baseline.

Note: A key difference between this process and "Identify customer needs" CUS.3, is that in CUS.3, the managed requirements are not necessarily specific to a particular project or to just software; here they are specific to both a particular project and to software. See also processes, "Establish contract" CUS.2, and "Develop software requirements" ENG.2, for other processes addressing the requirements.

PRO.4.1 Agree on requirements. Obtain agreement across teams on the customer's requirements, obtaining the appropriate sign-offs by representatives of all teams and other parties contractually bound to work to these requirements.

PRO.4.2 Establish customer requirements baseline. Document the customer's requirements and establish as a baseline for project use.

PRO.4.3 Manage customer requirements changes. Manage all changes made to the customer requirements to ensure those who are affected by the changes are able to assess the impact and risks, and initiate appropriate change control and mitigation actions.

PRO.4.4 Use customer requirements. Use the customer requirements as the basis for

- software project plans;
- requirements specifications;
- work products and activities.

PRO.4.5 Maintain traceability. Establish and maintain traceability of requirements to the project's work products throughout the software life cycle.

PRO.5 Manage quality

The purpose of the *Manage quality* process is to manage the quality of the project's products and services to ensure the resulting products and services satisfy the customer.

Managing quality involves identifying the required quality characteristics of the project’s products, working to achieve this quality, and demonstrating that this quality was achieved.

Inputs are the customer requirements and selected elements of the software project plans (see process PRO.2). Outputs should be integrated into the software project plans.

Note: There is another process with a similar name, "Perform quality assurance" SUP.3. Process PRO.5 focuses on identifying what needs to be done to build quality into the products and establishing management controls to ensure this gets done; whereas SUP.3 focuses more on an audit and review approach and on ensuring compliance.

PRO.5.1 Establish quality goals. Based on the customer's requirements for quality, establish quality goals for various checkpoints within the project's software life cycle (e.g. at the end of each phase).

PRO.5.2 Define quality metrics. Define metrics that measure the results of project activities to help assess whether the relevant quality goals have been achieved.

PRO.5.3 Identify quality activities. For each quality goal, identify activities which will help achieve that quality goal and integrate these activities into the software life cycle model.

PRO.5.4 Perform quality activities. Perform the identified quality activities.

PRO.5.5 Assess quality. At the identified checkpoints within the project's software life cycle, apply the defined quality metrics to assess whether the relevant quality goals have been achieved.

PRO.5.6 Take corrective action. When quality goals are not achieved, take corrective action.

Note: The corrective action can involve fixing the product generated by a particular project activity or changing the planned set of activities in order to better achieve the quality goals or both.

PRO.6 Manage risks

The purpose of the *Manage risks* process is to continuously identify and mitigate the risks in a project throughout the life-cycle of a project.

Managing risks involves continuously identifying new risks, working to effectively mitigate these risks, and evaluating the success of risk mitigation efforts.

A few base practices in other processes are likely to be more effective if this process is broadly performed. For example, base practices

- "Plan against failure" CUS.6.2;
- "Identify initial project risks" PRO.2.5.

PRO.6.1 Establish risk management scope. Determine the scope of risk management to be performed for this project: including the severity, probability, and type of risks to identify and manage.

PRO.6.2 Identify risks. Identify risks to the project as they develop.

Note: Risks include cost, schedule, effort, resource, and technical risks.

PRO.6.3 Analyze and prioritize risks. Assess the probability of occurrence, impact, time-frame, causes and interrelationships of risks for determining the priority in which to apply resources to mitigate these risks.

PRO.6.4 Develop mitigation strategies. Define appropriate strategies to take to mitigate each risk or set of related risks.

PRO.6.5 Define risk metrics. For each risk (or set of related risks) define the metrics that measure the change in the risk state (probability, impact, time-frame) and the progress of mitigation activities.

PRO.6.6 Implement mitigation strategies. Carry out the defined mitigation strategies.

PRO.6.7 Assess results of mitigation strategies. At identified checkpoints, apply the defined metrics to assess the expected progress and level of success of the mitigation strategies.

PRO.6.8 Take corrective action. When expected progress is not achieved, take corrective action.

Note: Corrective action may involve developing and implementing new mitigation strategies or adjusting the existing strategies.

PRO.7 Manage resources and schedule

The purpose of the *Manage resources and schedule* process is to co-ordinate and manage the project's resources and schedule during the life of the project toward the end of achieving the project's objectives and software requirements.

The project plans developed in process PRO.2 are inputs to and revised by this process.

PRO.7.1 Acquire resources. Allocate and distribute appropriate and sufficient resources to the software project activities, including both technical and management resources.

PRO.7.2 Track progress. Regularly compare and report the status of the project against the project plans.

Note: Particular aspects of the project to address include

– size;
– effort;
– cost;
– schedule;
– resources.
– risks

PRO.7.3 Conduct management reviews. Regularly and at major milestones, conduct management reviews to discuss and assess

– status against project plans and schedules;
– status of software risks;
– compliance with appropriate standards;
– and readiness for the next development steps.

PRO.7.4 Conduct technical reviews. Regularly and at major milestones, conduct technical reviews to discuss and assess

– technical status against the plans;
– technical issues not yet resolved;
– and technical readiness for the next development steps.

PRO.7.5 Manage commitments. Manage commitments to the estimates and plans with all affected groups, taking action when appropriate.

PRO.8 Manage subcontractors

The purpose of the *Manage subcontractors* process is to select qualified subcontractor(s) and manage their performance.

The base practices of this process help ensure that the supplier and supplier's subcontractors have the same understanding of project objectives and customer requirements and how they will work to jointly meet these successfully.

PRO.8.1 Establish statement of work. Establish a statement of the work to be performed under subcontract.

PRO.8.2 Qualify potential subcontractors. Qualify potential subcontractors through an assessment of their capability to perform the required software function.

Note: Subcontractors may be performing a variety of software related tasks including: software development, maintenance, documentation and training.

PRO.8.3 Select subcontractor. Select qualified subcontractors to perform defined portions of the contract.

PRO.8.4 Establish and manage commitments. Establish and manage commitments from and to the subcontractor.

PRO.8.5 Maintain communications. Exchange information on technical progress regularly with the subcontractor in order to maintain a common understanding of the work being performed.

PRO.8.6 Assess compliance. Assess compliance of the subcontractor against the agreed upon standards and procedures.

PRO.8.7 Assess subcontractor quality. Assess the quality of the subcontractors' delivered products and services to ensure the completeness, correctness and compliance with standards and specifications.

6.4 Support process category (SUP)

The *Support* process category consists of processes which may be employed by any of the other processes (including other supporting processes). The supporting processes can be employed at various life-cycle points and may be performed by the organization employing them, the customer, or by an independent organization.

To employ a supporting process within another process may require some tailoring of the support process, e.g. the formality and rigor of configuration management depends on the work product and severity of need for control and stability of the work product.

When assessing a particular support process, special consideration should be given to whether it has been implemented as broadly as it should be implemented. Also, it will require the assessor's judgement as to whether the implementation is satisfactorily formal and rigorous for the particular needs of the situation.

The processes belonging to the "Support" process category are:

SUP.1 Develop documentation

SUP.2 Perform configuration management

SUP.3 Perform quality assurance

SUP.4 Perform problem resolution

SUP.5 Perform peer reviews

SUP.1 Develop documentation

The purpose of the *Develop documentation* process is to develop and maintain documents needed by managers, engineers, users, or customers of the system or software.

Note: This process covers the development of documents such as

- project management documentation, such as plans;
- engineering work product documentation, such as design rationale;
- process documentation, such as a peer review procedure;
- and end user documentation, which describes the intended use of the system and software to a user.

SUP.1.1 Determine documentation requirements. Identify the requirements for the document to be built, including

- title;
- audience;
- purpose;
- objectives to be achieved;
- outline of its content;
- media and distribution requirements.

Note: The schedule for project documentation should be identified and integrated into the project's software plans.

SUP.1.2 Develop document. Develop the identified document according to its requirements.

SUP.1.3 Check document. Check the completed document against its requirements using, as appropriate, audience representatives, subject matter experts, or documentation experts. Revise the documentation.

Note: The value of performing this base practice is variable and depends on the audience and purpose. In the case of user documentation, this is a particularly important base practice, because it is important that documentation intended for use by system and software users adequately describe the system and software and how it is to be used in a manner which is clear and useful to the user.

SUP.1.4 Distribute document. Package and distribute the document as paper, electronic, or other media to the appropriate parties.

SUP.1.5 Maintain document. Maintain the document, and when it becomes necessary to modify it, perform the previous activities as appropriate.

Note: 1) If the document is part of a product baseline or if its control and stability are important, it should be modified and distributed in accordance with process, "Perform configuration management" SUP.2.

2) If the document is part of a product baseline under maintenance, its maintenance is covered by process ENG.7.

SUP.2 Perform configuration management

The purpose of the *Perform configuration management* process is to establish and maintain the integrity of all of the products of the software project throughout the project's software life cycle.

Note: Generic practice 2.2.2 at the Planned-and-Tracked Level instantiates this process for selected work products.

SUP.2.1 Establish configuration management library system. Establish and manage a repository with access controls that provides for

– storage and retrieval of configuration items (and their versions);
– sharing and transfer of configuration items between affected groups;
– recovery of archive versions of configuration items;
– correct creation of products from the library.

SUP.2.2 Identify configuration items. Identify each work product to be placed under configuration management.

Note: Examples include

– requirements, designs, code, tests;
– other product baselines (e.g., user documentation);
– software project plans;
– standards and procedures.

SUP.2.3 Maintain configuration item descriptions. Provide and maintain a description of each configuration item, identifying

– its decomposition into lower level configuration components;
– the person responsible for each item;
– when placed under configuration management.

SUP.2.4 Manage change requests. Record, review, approve, and track all change requests and problem reports for all configuration items and their versions.

SUP.2.5 Control changes. Provide access control to help maintain the correctness and integrity of the software items in the configuration management library system.

SUP.2.6 Build product releases. Build product releases only from configuration items in the library and only when authorized.

SUP.2.7 Maintain configuration item history. Maintain a history of each configuration item, recording configuration management actions against the item in sufficient detail to allow for recovery of previous versions.

SUP.2.8 Report configuration status. Regularly report on the results of performing the above activities and the status of each configuration item.

SUP.3 Perform quality assurance

The purpose of the *Perform quality assurance* process is to ensure that work products and activities comply with all applicable standards, procedures, and requirements.

The key requirement here is that an objective view of the quality of the process and work products be determined and reported. Quality Assurance can be implemented in different ways; it does not need to be performed by a separate group (e.g. a group called the "Software Quality Assurance Group").

Note: There is another process with a similar name, "Manage quality" PRO.5. Process PRO.5 focuses on identifying what needs to be done to build quality into the products and establishing management controls to ensure this gets done; whereas SUP.3 focuses more on an audit and review approach and on ensuring compliance.

SUP.3.1 Select project standards. Contribute to the project's software plans, by evaluating completeness of the plans and helping select the standards and procedures that will be used on the project.

SUP.3.2 Review software engineering activities. Review the software engineering activities against the plans and the selected standards and procedures.

SUP.3.3 Audit work products. Audit software work products against the selected standards and procedures.

SUP.3.4 Report results. Report the results of the above activities, in particular, deviations, to the appropriate levels of management and staff.

Note: The above is addressed by generic practices 2.3.1 and 2.3.2 at the Planned-and-Tracked Level.

SUP.3.5 Handle deviations. Deviations are addressed at the appropriate level of management, going to the next higher level, where necessary, until resolved.

SUP.4 Perform problem resolution

The purpose of the *Perform problem resolution* process is to ensure that all discovered problems are analyzed and removed, and trends are identified.

The problems being addressed here are any detected problems whatever their nature or source. Example sources are base practices

- CUS.4.3-4.5 (problems identified in customer audits and reviews);
- CUS.6.4 (problems relating to operation of software);
- CUS.6.5 (problems relating to use of software);
- CUS.7.3 (problems identified by monitoring performance);
- ENG.5.4, 5.6 (problems identified during software testing);
- ENG 6.3, 6.5 (problems identified during system testing);
- ENG.7.2 (user problems identified during maintenance);
- SUP.3.5 (deviations from requirements, standards, or procedures);
- SUP.5.7 (problems identified during peer reviews).

There are many other sources among the base practices, for example, those involving taking corrective action.

SUP.4.1 Prepare problem report. Prepare a problem report promptly after each problem is detected describing the nature of the problem.

Note: It may be the user or customer who does this, in which case this base practice can be considered, "not applicable."

SUP.4.2 Track problem report. Track the resolution of the problem/change report.

SUP.4.3 Prioritize problems. Categorize and prioritize according to the priority and category of the problem.

SUP.4.4 Determine resolution. Analyze the problem and, if possible, determine the problem's cause and document its resolution.

Note: The order in which problems are resolved may be prioritized by severity or underlying trends.

SUP.4.5 Correct the defect. Eliminate the defect in the product.

SUP.4.6 Distribute the correction. Distribute the corrected product.

Note: Each release of the product will normally include a number of defect corrections.

SUP.5 Perform peer reviews

The purpose of the *Perform peer reviews* process is to efficiently find and remove defects from products produced by the project.

When reviews involve the customer or management, usually at the end of a task, they are called, "technical reviews." For important work products (e.g., requirements, designs, code), it is important to hold a review early in the task so that defects can be efficiently found and rework significantly reduced. In such reviews, the emphasis is on technical correctness, and so the reviewers are typically the colleagues of the work product's producer (as opposed to management or the customer). Such reviews are called, "peer reviews."

There are different methods for carrying out a peer review by one's peers. Some of these go by the name, "inspections." This process description does not favour a particular method, only that there be (at least) one.

The work products to be peer reviewed are identified in the project's software plans (see process, "Establish project plan" PRO.2).

The action items which are output by this process are input to "Perform problem resolution" SUP.4 process.

Note: Generic practice 3.2.2 at the Well-Defined Level instantiates this process for selected work products. For example, in process, "Develop software design" ENG.3, an important work product is the software architecture (see base practice ENG.3.1). When assessing process ENG.3 against generic practice 3.2.2, we might ask whether software architectures are peer reviewed. The process we would expect to see applied when peer reviewing the software architecture, would correspond, at a very high level of description, with what is described here for process SUP.5.

SUP.5.1 Select work products. Identify the work products that are to undergo peer review.

Note: This is one aspect of planning the peer reviews, which is covered by a generic practice.

SUP.5.2 Identify review standards. Identify the standards (including checklists) to be used in conducting the peer reviews.

SUP.5.3 Establish completion criteria. Establish the completion criteria for successful completion of peer reviews.

SUP.5.4 Establish re-review criteria. Establish criteria for when and how to re-review a work product.

SUP.5.5 Distribute review materials. Distribute the materials for peer reviews well in advance of the reviews.

SUP.5.6 Conduct peer review. Conduct peer review on the selected work product.

SUP.5.7 Document action items. Document action items identified during peer reviews.

SUP.5.8 Track action items. Track to closure action items identified during peer reviews.

6.5 Organization process category (ORG)

The *Organization* process category consists of processes which establish the business goals of the organization and develop process, product, and resource assets which, when used by the projects in the organization, will help the organization achieve its business goals.

These organizational processes:

- build organizational infrastructure;
- leverage off the best of what is available in any one part of the organization (effective processes, advanced skills, quality code, good support tools);
- make it available to all;

When applying the common features to organization-level processes, note that the defined process described in the generic practices may be implemented at the organization as well as project level.

The processes belonging to the "Organization" process category are:

ORG.1 Engineer the business

ORG.2 Define the process

ORG.3 Improve the process

ORG.4 Perform training

ORG.5 Enable reuse

ORG.6 Provide software engineering environment

ORG.7 Provide work facilities

ORG.1 Engineer the business

The purpose of the *Engineer the business* process is to provide the individuals in the organization and projects with a vision and culture which empowers them to function effectively.

Although business re-engineering and Total Quality Management have a much broader scope than that of software process, software process improvement occurs in a business context and, to be successful, must address business goals.

ORG.1.1 Establish strategic vision. Establish a strategic vision for the organization that identifies what business the (software producing part of) the organization is in.

ORG.1.2 Deploy vision. Deploy the organization’s strategic vision to all individuals working for the organization.

ORG.1.3 Establish quality culture. Establish an organizational culture which supports a Total Quality focus on customer satisfaction.

ORG.1.4 Build integrated teams. Build teams with an integrated product perspective whose goal is to satisfy the customer.

ORG.1.5 Provide incentives. Provide incentives to team members to work as a team to accomplish the team's goal.

Note: Incentives may be provided by:

- establishing joint responsibility for team performance from problem formulation (beginning) to solution implementation (end);
- including peer inputs in performance reviews;
- linking rewards and recognition to team performance.

ORG.1.6 Define career plans. Define career plans for the individuals in the organization.

Note: In the case of small organizations with limited growth potential, this may simply take the form of an employee development plan (a plan for improving the knowledge, skills, and experience base of the employee).

ORG.2 Define the process

The purpose of the *Define the process* process is to build a reusable library of process definitions (including standards, procedures, and models) that will support stable and repeatable performance of the software engineering and management process (all the processes covered in this guide).

This process is an integral part of the common feature "Defining a Standard Process" at the Well-Defined Level. At least partial performance of this process is therefore a prerequisite for any other process achieving the Well-Defined Level, although it need not achieve the Well-Defined Level itself.

ORG.2.1 Define goals. Define the process goals that are to be achieved by following the process.

Note: One input to defining the goals is the organization's strategic vision. When defining a goal, it is important to state it in a manner so that its achievement can be measured and determined with some objectivity. See base practice, "Define process measures" ORG.2.9.

ORG.2.2 Identify current activities, roles & responsibilities. Identify the activities that comprise the way the process is currently and/or should be performed and identify the roles and responsibilities for these activities.

ORG.2.3 Identify inputs and outputs. Identify the inputs and outputs for the process.

ORG.2.4 Define entry and exit criteria. Define the criteria for entering and exiting the process.

ORG.2.5 Define control points. Define the points in the process where key reviews and decisions are made.

ORG.2.6 Identify external interfaces. Identify the interfaces with related processes, which supply inputs and consume outputs.

Note: Base practice, "Identify inputs and outputs" ORG.2.3 identifies the work products entering and leaving the process. This base practice identifies the relationship to other processes that this process may need to work with (e.g. management processes as well as supplier/consumer processes) and inputs and outputs between them.

ORG.2.7 Identify internal interfaces. Identify the interfaces between the activities in the process.

ORG.2.8 Define quality records. Define the quality records that will demonstrate conformance to the process and determine their retention period..

ORG.2.9 Define process measures. Define measures for the process that can be used to determine achievement of the process goals.

Note. These measures will address such goals as process efficiency and quality.

ORG.2.10 Document the standard process. Document the standards, procedures, and models for performing the process and characterizing its outputs.

Note: Example areas that might be covered by software engineering standards include

- requirements specification;
- design methods;
- coding style;
- programming languages;
- testing;
- security;
- human factors;
- documentation;
- project management plans;
- software quality assurance plans;
- configuration management plans.

ORG.2.11 Establish policy. Establish a written organization policy for performing the process.

ORG.2.12 Establish performance expectations. Establish expectations for process performance when using the organization's standard process family.

Note: These expectations will typically be quantitative. The policy in ORG.2.11 sets expectations for what process will be performed and how; this practice quantifies the expectations for performance. Even when these expectations are not be well-bounded in the sense of a Level 4 capability, they may include quantitative criteria that guide effective implementation of the process.

ORG.2.13 Deploy the process. Deploy the organization's standard process family available throughout the organization.

Note: Deployment of the organization’s standard process family will frequently involve training.

ORG.3 Improve the process

The purpose of the *Improve the process* process is to continually improve the effectiveness and efficiency of the processes used by the organization in line with the busines need.

Note: The focus of this process will typically be the standard family of processes, which are built according to generic practice 3.1.1. That is why this process is considered an organizational process: organizational standards and process assets provide a common basis for continual process improvement. Although there will not usually be sufficient time to go through a full process improvement cycle at the project level, this process can also be used to improve the processes of large, long-duration projects.

ORG.3.1 Identify improvement opportunities. Identify opportunities for software process improvement by regularly analyzing

- measures of software quality and productivity, possible process and technology changes;
- internal and external comparisons (benchmarks);
- contract and product requirement changes.

ORG.3.2 Define scope of improvement activities. Define the purpose, objectives, scope, and priorities of the process improvement activities in accordance with the business goals of the organization.

ORG.3.3 Understand the process. Assess the process to understand its strengths and weaknesses

ORG.3.4 Identify improvements. Identify where the process needs to be improved to achieve its process goals.

Note: See base practice, "Define goals" ORG.2.1, for more on process goals.

ORG.3.5 Prioritize improvements. Prioritize the improvements which can be made in the process based on an analysis of the impact of potential improvements on achieving the goals of the process.

ORG.3.6 Define measures of impact. Define measures that can be used to determine the impact of the process changes on achieving the process's goals.

Note: These measures would include those identified in base practice, "Define process measures" ORG.2.9.

ORG.3.7 Change the process. Change the process to improve it.

ORG.3.8 Confirm the improvement. Pilot test changes to confirm that they improve the process based on analysis of appropriate data.

ORG.3.9 Deploy improvement. Deploy improved processes across the organization as appropriate.

ORG.4 Perform training

The purpose of the *Perform training* process is to provide the organization and projects with individuals who possess the needed skills and knowledge to perform their roles effectively.

Training encompasses more than classroom hours of instructor-led education. Other activities also serve to satisfy the purpose of this process such as on-the-job training and mentoring.

Note: "Perform training" supports the generic training practice 2.1.5 in "Planning performance" at the Planned-and-Tracked Level, so organization-level training is not required to satisfy that practice.

ORG.4.1 Identify training needs. Identify common training needs across the organization based on organizational and project inputs to build the knowledge and skills of the staff.

ORG.4.2 Develop or acquire training. Develop or acquire training that addresses the common training needs.

ORG.4.3 Train personnel. Train personnel to have the knowledge and skills needed to perform their roles.

ORG.4.4 Maintain training records. Maintain appropriate records of training and experience for the staff.

ORG.5 Enable reuse

The purpose of the *Enable reuse* process is to maximize reuse of existing system and software components, leading to lower development and maintenance costs and higher quality product lines.

ORG.5.1 Determine organizational reuse strategy. Determine which product lines and types of work products should be supported with reuse.

Note: Much reuse can be accomplished within product lines. When software architectures are fairly constant across projects, high levels of design reuse are possible.

ORG.5.2 Identify reusable components. Identify system and software components which could profitably be reused within the product lines established by the organization.

Note: Consider only high quality system and software components for reuse.

ORG.5.3 Develop reusable components. Develop system and software components which are designed for reuse.

ORG.5.4 Establish a reuse library. Establish a library of reusable components, including the mechanisms for identifying and retrieving components.

ORG.5.5 Certify reusable components. Certify that components placed in the reuse library have been appropriately packaged for reuse.

ORG.5.6 Integrate reuse into life cycle. Modify the software life cycle and/or standard process to address the incorporation of reusable components as appropriate.

ORG.5.7 Propagate change carefully. Before making a global change to a reusable component, evaluate the impact of the change, and the impact to systems and software in which it was incorporated.

Note: Sometimes due to defects, improvements in algorithms, etc., it is desirable to propagate a change in a reusable component to many of the systems and software in which it was incorporated. Before doing so, carefully evaluate the impact of propagating the change.

ORG.6 Provide software engineering environment

The purpose of the *Provide software engineering environment* process is to provide an integrated set of software development tools for use by the projects in the organization, consistent and supportive of the process standard.

ORG.6.1 Identify software engineering environment requirements. Determine requirements for the software engineering environment, identifying

- process roles and activities it should support;
- security issues it should address;
- throughput and data sharing requirements;
- backup and recovery.

ORG.6.2 Provide a software engineering environment. Acquire and provide a software engineering environment which satisfies the requirements.

ORG.6.3 Provide support for developers. Provide support for the developers who will utilize the software engineering environment.

Note: Support includes identifying and resolving problems arising from the use of provided tools and facilities.

ORG.6.4 Maintain software engineering environment. Perform maintenance on the software engineering environment for the purposes of

- correcting defects;
- improving performance;
- modifying the environment to keep up with changes in the process activities and tools it supports;
- controlling changes to enable regression if necessary.

ORG.7 Provide work facilities

The purpose of the *Provide work facilities* process is to provide a secure and reliable environment in which software project activities may be carried out.

ORG.7.1 Provide productive workspace. Provide a productive workspace, with appropriate furnishings and office equipment.

Note: Possible attributes of a productive workspace include: safety, quietness, and comfort.

ORG.7.2 Ensure data integrity. Provide the means to ensure that data resulting from the software project's activities are appropriately archived and protected from corruption.

ORG.7.3 Provide data backups. Provide the means for performing regular backup and archival of data generated by the software project's activities to prevent loss.

ORG.7.4 Provide building facilities. Provide supportive building facilities which may include

- access to meeting rooms;
- adequate storage space;
- access to parking;
- access to rest rooms;
- adequate communication facilities;
- adequate lighting;
- adequate environmental control (e.g. heating/cooling);
- adequate refreshments.

ORG.7.5 Provide remote access facility. Provide the software project's technical and managerial staff with the means to access their work environment and data from a remote location, as appropriate.

Annex A (normative)

Extended processes and variant models

A.1 Requirements for extended process and variant models

In order to address the unique needs of a specific organisation or industry sector, other models may be built by selecting specific processes from the model defined in this part of the International Standard and providing guidance on how to interpret the practices. Alternatively, processes may be extended by including additional base practices, possibly supplemented by guidance on how to interpret the adequacy of practices. An extended process may also be an entirely new process. These models, referred to as variant models, shall satisfy the following criteria.

A variant model shall be a proper (non-empty) subset of the processes defined in this part of the International Standard. Model practices should be separately identifiable from any tailoring that may be made for purposes of comparison across conformant variants.

An extended process shall not change or delete base practices.

New processes may be added to a variant model, but these new processes shall not contain base practices drawn from other processes within this part of the International Standard. This means that it is not possible to construct a variant model by selecting a subset of base practices from a number of processes and then deleting all of those processes.

A variant model shall be documented to identify its differences from this part of the International Standard.

A variant model shall provide direct traceability between its base practices and the base practices in this part of the International Standard.

Extended processes shall be documented according to the nomenclature defined in clause 4.

A.2 Guidelines for the construction of extended processes

Extended processes should be documented according to the style guide in Annex H.

New processes should have a defined purpose.

New processes should have a sufficient set of base practices to permit the transformation of its inputs into its outputs and to meet the defined purpose of the process.

The common features and generic practices in this part of the International Standard should be applicable to extended processes.

Extended processes may provide guidance on the appropriate interpretation of a practice. This may be implemented as adequacy indicators to be incorporated into an assessment instrument as described in part 6 of this International Standard.

Extended processes may add new base practices to those in its equivalent in this part of the International Standard.

The base practices in an extended process should be checked for completeness.

The base practices in an extended process should be checked to ensure they are necessary for achieving the purpose of the process.

Annex B (informative)

Summary list of practices

B.1 Summary list of base practices

As in clause 6, the base practices are grouped under the process they belong to and processes are grouped under the process category they belong to. Titles only are shown. This summary provides a high-level overview of the primary activities the model prescribes for each process. It can be used to get a quick understanding of the content of this part of the International Standard, except for the generic practices. This annex is for informational purposes only - it is not intended to replace clause 6 in assessments.

CUS Customer-Supplier process category

CUS.1	**Acquire software product and/or service**
CUS.1.1	Identify the need
CUS.1.2	Define the requirements
CUS.1.3	Prepare acquisition strategy
CUS.1.4	Prepare request for proposal
CUS.1.5	Select software product supplier

CUS.2	**Establish contract**
CUS.2.1	Review before contract finalization
CUS.2.2	Negotiate contract
CUS.2.3	Determine interfaces to independent agents
CUS.2.4	Determine interfaces to subcontractors

CUS.3	**Identify customer needs**
CUS.3.1	Obtain customer requirements and requests
CUS.3.2	Understand customer expectations
CUS.3.3	Keep customers informed

CUS.4	**Perform joint audits and reviews**
CUS.4.1	Establish joint audits and reviews
CUS.4.2	Prepare for customer audits and reviews
CUS.4.3	Conduct joint management reviews
CUS.4.4	Conduct joint technical reviews
CUS.4.5	Support customer acceptance review
CUS.4.6	Perform joint process assessment

CUS.5	**Package, deliver, and install the software**
CUS.5.1	Identify installation requirements
CUS.5.2	Prepare site for installation
CUS.5.3	Pack software
CUS.5.4	Deliver software
CUS.5.5	Verify correct receipt
CUS.5.6	Install software
CUS.5.7	Provide handling and storage procedures

CUS.6 Support operation of software

CUS.6.1	Identify operational risks
CUS.6.2	Perform operational testing
CUS.6.3	Operate the software
CUS.6.4	Resolve operational problems
CUS.6.5	Handle user requests
CUS.6.6	Document temporary work-arounds
CUS.6.7	Monitor system capacity and service

CUS.7 Provide customer service

CUS.7.1	Train customer
CUS.7.2	Establish product support
CUS.7.3	Monitor performance
CUS.7.4	Install product upgrades

CUS.8 Assess customer satisfaction

CUS.8.1	Determine customer satisfaction level
CUS.8.2	Compare with competitors
CUS.8.3	Communicate customer satisfaction

ENG Engineering process category

ENG.1 Develop system requirements and design

ENG.1.1	Specify system requirements
ENG.1.2	Describe system architecture
ENG.1.3	Allocate requirements
ENG.1.4	Determine release strategy

ENG.2 Develop software requirements

ENG.2.1	Determine software requirements
ENG.2.2	Analyze software requirements
ENG.2.3	Determine operating environment impact
ENG.2.4	Evaluate requirements with customer
ENG.2.5	Update requirements for next iteration

ENG.3 Develop software design

ENG.3.1	Develop software architectural design
ENG.3.2	Design interfaces at top level
ENG.3.3	Develop detailed design
ENG.3.4	Establish traceability

ENG.4 Implement software design

ENG.4.1	Develop software units
ENG.4.2	Develop unit verification procedures
ENG.4.3	Verify the software units

ENG.5 Integrate and test software

ENG.5.1	Determine regression test strategy
ENG.5.2	Build aggregates of software units
ENG.5.3	Develop tests for aggregates
ENG.5.4	Test software aggregates
ENG.5.5	Develop tests for software
ENG.5.6	Test integrated software

ENG.6 Integrate and test system

ENG.6.1	Build aggregates of system elements
ENG.6.2	Develop tests for aggregates
ENG.6.3	Test system aggregates
ENG.6.4	Develop tests for system
ENG.6.5	Test integrated system

ENG.7 Maintain system and software

ENG.7.1	Determine maintenance requirements
ENG.7.2	Analyze user problem and enhancements
ENG.7.3	Determine modifications for next upgrade
ENG.7.4	Implement and test modifications
ENG.7.5	Upgrade user system

PRO Project process category

PRO.1 Plan project life cycle

PRO.1.1	Evaluate options for product development
PRO.1.2	Select software life cycle model
PRO.1.3	Describe activities and tasks
PRO.1.4	Establish task sequences
PRO.1.5	Document activities

PRO.2 Establish project plan

PRO.2.1	Develop work breakdown structure
PRO.2.2	Identify project standards
PRO.2.3	Identify specialized facilities
PRO.2.4	Determine reuse strategy
PRO.2.5	Develop project estimates
PRO.2.6	Identify initial project risks
PRO.2.7	Identify project measures
PRO.2.8	Establish project schedule
PRO.2.9	Establish project commitments
PRO.2.10	Document project plans

PRO.3 Build project teams

PRO.3.1	Define project teams
PRO.3.2	Empower project teams
PRO.3.3	Maintain project team interactions
PRO.3.4	Manage inter-team issues

PRO.4 Manage requirements

PRO.4.1	Agree on requirements
PRO.4.2	Establish customer requirements baseline
PRO.4.3	Manage customer requirements changes
PRO.4.4	Use customer requirements
PRO.4.5	Maintain traceability

PRO.5 Manage quality

PRO.5.1	Establish quality goals
PRO.5.2	Define quality metrics
PRO.5.3	Identify quality activities
PRO.5.4	Perform quality activities
PRO.5.5	Assess quality
PRO.5.6	Take corrective action

PRO.6 Manage risks

PRO.6.1	Establish risk management scope
PRO.6.2	Identify risks
PRO.6.3	Analyze and prioritize risks
PRO.6.4	Develop mitigation strategies
PRO.6.5	Define risk metrics
PRO.6.6	Implement mitigation strategies
PRO.6.7	Assess results of mitigation strategies
PRO.6.8	Take corrective action

PRO.7 Manage resources and schedule

PRO.7.1	Acquire resources
PRO.7.2	Track progress
PRO.7.3	Conduct management reviews
PRO.7.4	Conduct technical reviews
PRO.7.5	Manage commitments

PRO.8 Manage subcontractors

PRO.8.1	Establish statement of work
PRO.8.2	Qualify potential subcontractors
PRO.8.3	Select subcontractor
PRO.8.4	Establish and manage commitments
PRO.8.5	Maintain communications
PRO.8.6	Assess compliance
PRO.8.7	Assess subcontractor quality

SUP Support process category

SUP.1 Develop documentation

SUP.1.1	Determine documentation requirements
SUP.1.2	Develop document
SUP.1.3	Check document
SUP.1.4	Distribute document
SUP.1.5	Maintain document

SUP.2 Perform configuration management

SUP.2.1	Establish configuration management library system
SUP.2.2	Identify configuration items
SUP.2.3	Maintain configuration item descriptions
SUP.2.4	Manage change requests
SUP.2.5	Control changes
SUP.2.6	Build product releases
SUP.2.7	Maintain configuration item history
SUP.2.8	Report configuration status

SUP.3 Perform quality assurance

SUP.3.1	Select project standards
SUP.3.2	Review software engineering activities
SUP.3.3	Audit work products
SUP.3.4	Report results
SUP.3.5	Handle deviations

SUP.4 Perform problem resolution

SUP.4.1	Prepare problem report
SUP.4.2	Track problem report
SUP.4.3	Prioritize problems
SUP.4.4	Determine resolution
SUP.4.5	Correct the defect
SUP.4.6	Distribute the correction

SUP.5 Perform peer reviews

SUP.5.1	Select work products
SUP.5.2	Identify review standards
SUP.5.3	Establish completion criteria
SUP.5.4	Establish re-review criteria
SUP.5.5	Distribute review materials
SUP.5.6	Conduct peer review
SUP.5.7	Document action items
SUP.5.8	Track action items

ORG Organization process category

ORG.1 Engineer the business

ORG.1.1	Establish strategic vision
ORG.1.2	Deploy vision
ORG.1.3	Establish quality culture
ORG.1.4	Build integrated teams
ORG.1.5	Provide incentives
ORG.1.6	Define career plans

ORG.2 Define the process

ORG.2.1	Define goals
ORG.2.2	Identify current activities, roles and responsibilities
ORG.2.3	Identify inputs and outputs
ORG.2.4	Define entry and exit criteria
ORG.2.5	Define control points
ORG.2.6	Identify external interfaces
ORG.2.7	Identify internal interfaces
ORG.2.8	Define quality records
ORG.2.9	Define process measures
ORG.2.10	Document the standard process
ORG.2.11	Establish policy
ORG.2.12	Establish performance expectations
ORG.2.13	Deploy the process

ORG.3 Improve the process

ORG.3.1	Identify improvement opportunities
ORG.3.2	Define scope of improvement activities
ORG.3.3	Understand the process
ORG.3.4	Identify improvements
ORG.3.5	Prioritize improvements
ORG.3.6	Define measures of impact
ORG.3.7	Change the process
ORG.3.8	Confirm the improvement
ORG.3.9	Deploy improvement

ORG.4 Perform training

ORG.4.1	Identify training needs
ORG.4.2	Develop or acquire training
ORG.4.3	Train personnel
ORG.4.4	Maintain training records

ORG.5 Enable reuse

ORG.5.1	Determine organizational reuse strategy
ORG.5.2	Identify reusable components
ORG.5.3	Develop reusable components
ORG.5.4	Establish a reuse library
ORG.5.5	Certify reusable components
ORG.5.6	Integrate reuse into life cycle
ORG.5.7	Propagate change carefully

ORG.6 Provide software engineering environment

ORG.6.1	Identify software engineering environment requirements
ORG.6.2	Provide a software engineering environment
ORG.6.3	Provide support for developers
ORG.6.4	Maintain software engineering environment

ORG.7 Provide work facilities

ORG.7.1	Provide productive workspace
ORG.7.2	Ensure data integrity
ORG.7.3	Provide data backups
ORG.7.4	Provide building facilities
ORG.7.5	Provide remote access facility

B.2 Summary list of generic practices

This summary provides a high-level overview of the capability levels in the model. As in clause 5, the generic practices are grouped under common features and capability levels. Titles only are shown. This annex is for informational purposes only - it is not intended to replace Section 5 in assessments.

Level 1:Performed-Informally Level

Common Feature 1.1: Performing Base Practices

1.1.1 Perform the process.

Level 2:Planned-and-Tracked Level

Common Feature 2.1: Planning Performance

2.1.1 Allocate resources.
2.1.2 Assign responsibilities.
2.1.3 Document the process.
2.1.4 Provide tools.
2.1.5 Ensure training.
2.1.6 Plan the process.

Common Feature 2.2: Disciplined Performance

2.2.1 Use plans, standards, and procedures.
2.2.2 Do configuration management.

Common Feature 2.3: Verifying Performance

2.3.1 Verify process compliance.
2.3.2 Audit work products.

Common Feature 2.4: Tracking Performance

2.4.1 Track with measurement.
2.4.2 Take corrective action.

Level 3: Well-Defined Level

Common Feature 3.1: Defining a Standard Process

3.1.1 Standardize the process.
3.1.2 Tailor the standard process.

Common Feature 3.2: Performing the Defined Process

3.2.1 Use a well-defined process.
3.2.2 Perform peer reviews.
3.2.3 Use well-defined data.

Level 4:Quantitatively-Controlled Level

Common Feature 4.1: Establishing Measurable Quality Goals

4.1.1 Establish quality goals.

Common Feature 4.2: Objectively Managing Performance

4.2.1 Determine process capability.
4.2.2 Use process capability.

Level 5: Continuously-Improving Level

Common Feature 5.1: Improving Organizational Capability

5.1.1 Establish process effectiveness goals.
5.1.2 Continuously improve the standard process.

Common Feature 5.2: Improving Process Effectiveness

5.2.1 Perform causal analysis.
5.2.2 Eliminate defect causes.
5.2.3 Continuously improve the defined process.

Annex C (Informative)

Components of the model

C.1 Introduction

The process model in this part of the International Standard defines processes and practices that may be implemented to establish and improve an organization's software acquisition, development, maintenance, operation and support capabilities. Practices in this model are organized using an architecture that will help software personnel understand how to continuously improve the management of software processes. The architecture has been designed to facilitate the assessment of an organization's software processes and make judgements and recommendations regarding improvements to them.

Each process in the model is described by base practices, which are the essential activities of that specific process. Processes, in turn, are grouped into five process categories.

Evolving process capability is expressed in terms of capability levels, common features, and generic practices. Generic practices implement or institutionalize a process in a general way. These practices represent the activities necessary to manage a process and improve its capability to perform. They are potentially applicable to any process. They are grouped into common features and capability levels according to the aspect of implementation or institutionalization they address.

C.2 Definitions of architecture components

C.2.1 Grouping by type of activity

The first set of architectural components — process category, process, base practices — is a grouping by *type of activity*.

Process Category Each process category is a set of processes addressing the same general area of activity.

The process categories covered in this part of the International Standard address five general areas of activity: customer-supplier, engineering, project, support, and organization.

The **Customer-Supplier** process category consists of processes that directly impact the customer, supporting development and transition of the software to the customer, and provide for its correct operation and use.

The **Engineering** process category consists of processes that directly specify, implement, or maintain a system and software product and its user documentation.

The **Project** process category consists of processes which establish the project, and co-ordinate and manage its resources to produce a product or provide services which satisfy the customer.

The **Support** process category consists of processes which enable and support the performance of the other processes on a project.

The **Organization** process category consists of processes which establish the business goals of the organization and develop process, product, and resource assets which will help the organization achieve its business goals.

These process categories are related as illustrated in figure 1. "Customer-Supplier" process category consists of the processes closest to the customer. We can think of each succeeding process category as one layer removed from the preceding layer. Thus the "Engineering" process category consists of processes which build the product which is delivered to the customer. The "Project" process category consists of processes which manage the development of the product. The "Support" process category consists of processes which enable and support the performance of the previous processes, whether management, engineering, or customer-related. Finally, the "Organizational" process category consists of processes that build organizational infrastructure on which all previous processes can potentially exercise the most benefit to the organization.

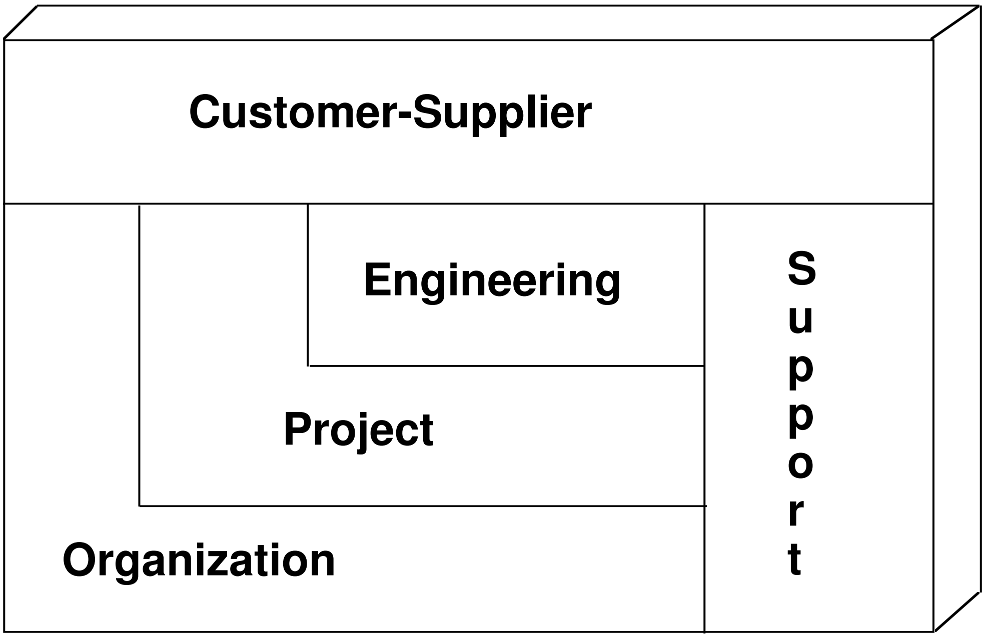


Figure 1 – Interrelationship of process categories.

Process A process is a set of activities that achieves a purpose. Each process in this part of the International Standard has a purpose and consists of a set of practices that address that purpose.

For example, the Engineering process "Develop software design" ENG.3 contains this purpose statement: "The purpose of the *Develop software design* process is to establish a software design that effectively accommodates the software requirements; at the top-level this identifies the major software components and refines these into lower level software units which can be coded, compiled, and tested." The purpose statement indicates the reason for performing the "Develop software design" ENG.3 process.

The processes in this part of the International Standard are not processes in the sense of being complete process models or descriptions. These processes contain essential practices, but they do not describe how to perform the process. This part of the International Standard is a descriptive model, not a prescriptive model.

A summary list of all the processes in this model are listed in Annex B.

Base Practice A base practice is a software engineering or management activity that addresses the purpose of a particular process. Consistently performing the base practices associated with a process will help in consistently achieving its purpose.

Thus a process consists of a set of *base practices.* For example, "Develop software design" ENG.3 consists of the following set of base practices:

ENG.3.1 Develop software architectural design

ENG.3.2 Design interfaces at top level

ENG.3.3 Develop detailed design

ENG.3.4 Establish traceability

Note that these practices should be performed to achieve the defined purpose of the process.

The base practices in this part of the International Standard are described at an abstract level, identifying "what" should be done without specifying "how". See other ISO JTC1/SC7 standards for further information and guidance on implementing these practices.

These processes and activities characterize performance of a process, even if that performance is not systematic. Performance of the base practices may be ad hoc, unpredictable, inconsistent, poorly planned, and/or result in poor quality products, but those work products are at least marginally usable in achieving the purpose of the process. Implementing only the base practices of a process may be of minimal value and represents only the first step in building process capability, but the base practices represent the unique, functional activities of the process.

C.2.2 Grouping by type of implementation or institutionalization activity

The second set of architectural components - capability levels, common features, and generic practices - groups by *type of implementation or institutionalization activity*. They deal, at different levels of abstraction, with the activities necessary to manage a process and improve its capability to perform.

Capability Level A capability level is a set of common features (sets of activities) that work together to provide a major enhancement in the capability to perform a process.

For example, to effectively attain conformance to the practices at the Well-Defined Level (Level 3), for some processes the organization will require support through an effective implementation of another process, namely "Define the process" ORG.2.

Capability levels provide two benefits: they acknowledge dependencies among the practices of a process, and they help an organization identify which improvements it might perform first, based on a rational sequence of process implementation. Capability levels can be represented either by number or by title, but it is recommended that the titles be generally used instead of the numbers as they are more descriptive of the practices the levels contain.

Each level provides a major enhancement in capability to that provided by its predecessors in the performance of a process. Together, they constitute a route map for improving a specific process in a logical fashion. The capability levels are a rational way of progressing through enhancements to a process. This may not, however, apply to all software environments. Organization-wide improvement considerations and taking advantage of what is already implemented in an organization may influence progression through the capability levels. It should also be noted that there are dependencies between processes, and the successful satisfaction of a capability level within a process may require support from another process.

Capability levels

0 Not-Performed

The Not-Performed level has no common features. There is general failure to perform the base practices in the process. There are no easily identifiable work products or outputs of the process.

1 Performed-Informally

Base practices of the process are generally performed. The performance of these base practices may not be rigorously planned and tracked. Performance depends on individual knowledge and effort. Work products of the process testify to the performance. Individuals within the organization recognize that an action should be performed, and there is general agreement that this action is performed as and when required. There are identifiable work products for the process.

2 Planned-and-Tracked

Performance of the base practices in the process is planned and tracked. Performance according to specified procedures is verified. Work products conform to specified standards and requirements.

The primary distinction from the Performed-Informally Level is that the performance of the process is planned and managed and progressing towards a well-defined process

3 Well-Defined

Base practices are performed according to a well-defined process using approved, tailored versions of standard, documented processes.

The primary distinction from the Planned-and-Tracked Level is that the process of the Well-Defined Level is planned and managed using an organization-wide standard process.

4 Quantitatively-Controlled

Detailed measures of performance are collected and analyzed. This leads to a quantitative understanding of process capability and an improved ability to predict performance. Performance is objectively managed. The quality of work products is quantitatively known.

The primary distinction from the Well-Defined Level is that the defined process is quantitatively understood and controlled.

5 Continuously-Improving

Quantitative process effectiveness and efficiency goals (targets) for performance are established, based on the business goals of the organization. Continuous process improvement against these goals is enabled by quantitative feedback from performing the defined processes and from piloting innovative ideas and technologies.

The primary distinction from the Quantitatively-Controlled Level is that the defined process and the standard process undergo continuous refinement and improvement, based on a quantitative understanding of the impact of changes to these processes.

Common Feature A common feature is a set of practices that address an aspect of process implementation or institutionalization.

As an example, the Planned-and-Tracked Level contains the Planning Performance, Disciplined Performance, Verifying Performance, and Tracking Performance common features (See clause 5). The Tracking Performance common feature consists of practices that track process status using measurement and take corrective action as appropriate.

The common features associated with capability levels 1-5 are given below. The names of the common features are italicized. Under each is a bulleted short description of the kinds of things that need to be addressed by the organization in order to implement the common feature. (The generic practices belonging to each common feature specifically address these bullets.)

Performed-Informally Level

Performing Base Practices

- perform the process

Planned-and-Tracked Level

When interpreting the common features for the Planned-and-Tracked Level, remember that the particular place in the organization where these practices are performed is not specified. It can be at the level of an individual group, a project, or the entire organization. In most instances, the following are expected to be performed at the level of a project.

Planning Performance

- allocate resources
- assign responsibilities
- document the process
- provide tools
- ensure training
- plan the process

Disciplined Performance

- use plans, standards, and procedures
- do configuration management

Verifying Performance

- verify process compliance
- audit work products

Tracking Performance

- track with measurement
- take corrective action

Well-Defined Level

When interpreting the common features for the Well-Defined Level, remember that: 1) there can be more than one "standard" process for the same process in an organization (e.g. due to developing software for different applications or markets), and 2) the place in the organization where tailoring and use of the defined process takes place is not specified, though it will normally be at the level of the project. For example, organization processes are typically not performed by software projects.

Defining a Standard Process

- standardize the process
- tailor the standard process

Performing the Defined Process

- use a well-defined process
- perform peer reviews
- use well-defined data

Quantitatively-Controlled Level

When interpreting the common features for the Quantitatively-Controlled Level, note that each defined process is a tailored version of the standard process, and that each standard process covers the base practices for that process.

Establishing Measurable Quality Goals

- establish quality goals

Objectively Managing Performance

- determine process capability
- use process capability

Continuously-Improving Level

When interpreting the common features for the Continuously-Improving Level, note that change is based on a quantitative understanding of the effectiveness of process changes.

Improving Organizational Capability

- establish process effectiveness goals
- continuously improve the standard process

Improving Process Effectiveness

- perform causal analysis
- eliminate defect causes
- continuously improve the defined process

Generic Practice A generic practice is an implementation or institutionalization practice that enhances the capability to perform any process.

As an example, the Tracking Performance common feature contains this generic practice (see clause 5): “2.4.1 Track with measurement. Track the status of the process against the plan using measurement.” Recording data on conditions and results while performing a process increases understanding of its performance, and addresses the "tracking performance" aspect of performing the process. This is one of many generic practices that contribute to improved capability to manage the performance of the process (the data conveys status and results to the one tracking the performance of the process).

The generic practices apply to a process as a whole and can be aggregated by common features and by capability levels to describe the capability of a process. The generic practices characterize good process management that results in an increasing process capability for any process. A planned, well-defined, measured, and continuously improving process is consistently performed as the generic practices are implemented for a process. This process capability is built on the foundation of the base practices that describe the unique, functional activities of the process.

When using the generic practices, it may be helpful to substitute the specific name of the process for the phrase “the process.” For example, when judging generic practice 2.1.1 for the "Perform configuration management" SUP.2 process, “allocate adequate resources (including people) for performing the process” becomes “allocate adequate resources (including people) for performing configuration management.”

C.3 Relationship of components

The model’s architecture, shown in figure 2, contains two hierarchies. The hierarchy on the left consists of process categories, which are composed of processes, which are composed of base practices. This is a decomposition by type of activity.

Processes are rated in terms of the right-side hierarchy. Processes may be rated at a capability level; capability levels are composed of common features (except for level 0: Not-Performed); common features, in turn, are composed of generic practices.

Combining these two hierarchies in rating a process is illustrated in figure 3. Rating processes is described in detail in part 3 of this International Standard.

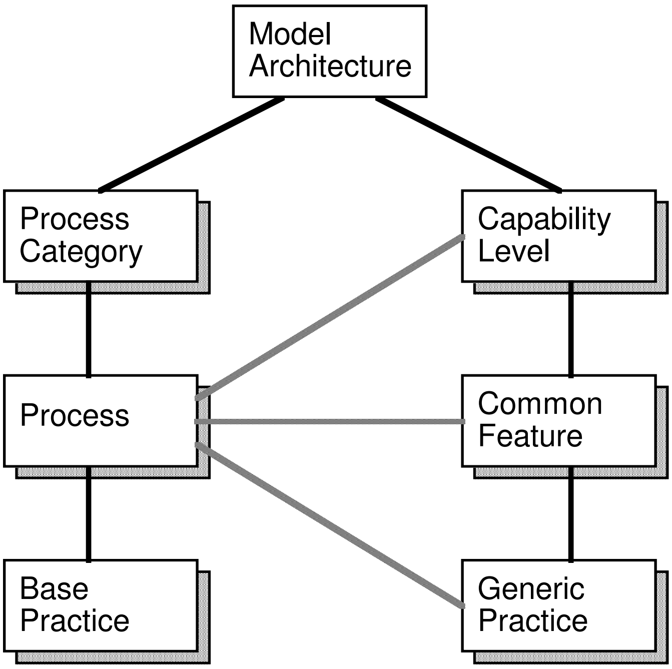


Figure 2 – Model architecture: two ways to classify practices.

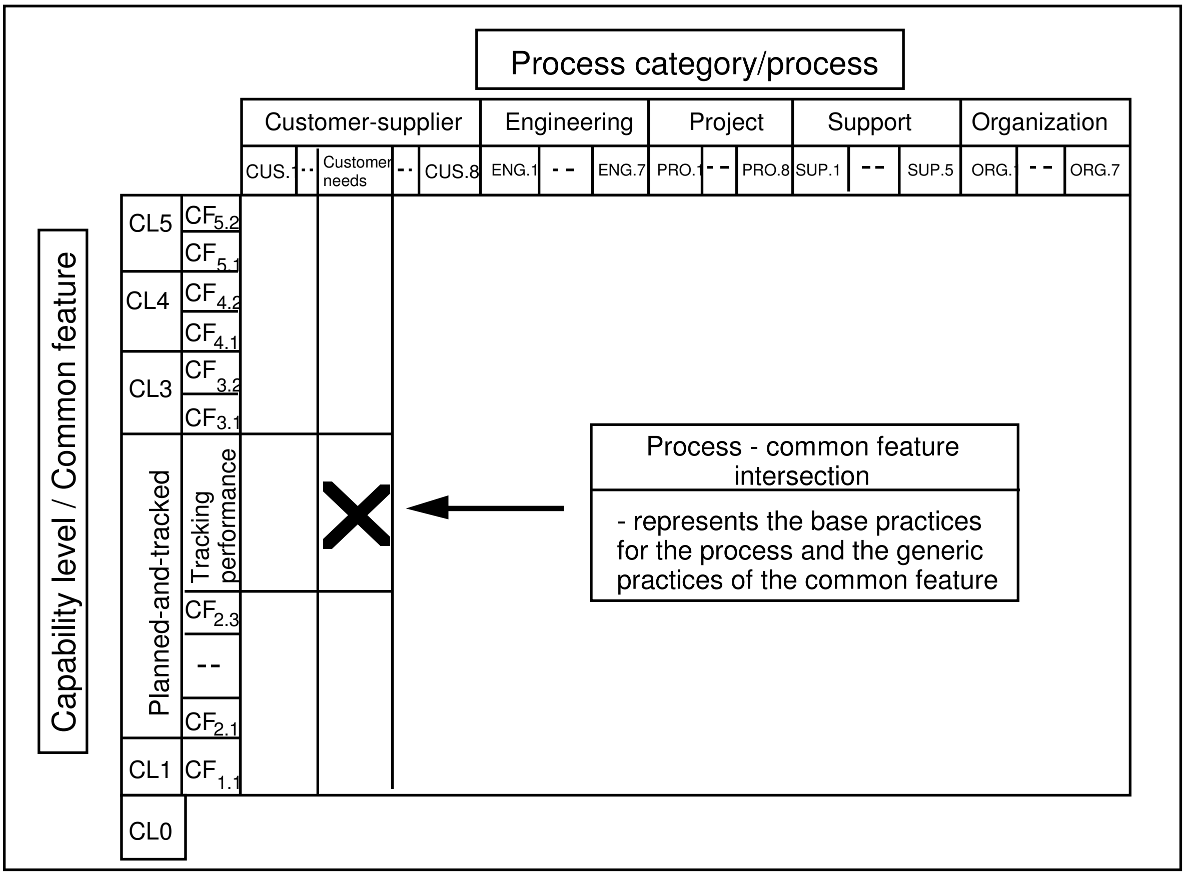


Figure 3 Architecture: 2-dimensional view

C.4 Ordering common features by capability levels

By their nature, there is more than one way to group aspects into common features and common features into capability levels. The ordering of the common features in this model stems from the observation that some implementation and institutionalization aspects benefit from the presence of others. This is especially true if institutionalization aspects are well established. For example, the provision of a well-defined, usable process for an entire organization to tailor and use should follow from some experience in managing the performance of that process at the level of individual projects. An example of this is that prior to institutionalizing a specific estimation process for an entire organization, the organization first attempts to use the estimation process on a project. Some aspects of process implementation and institutionalization should be considered together (not one ordered before the other) since they work together toward enhancing capability.

Common features and capability levels are important in performing an assessment and improving an organization's process capability. In the case of an assessment where an organization has some, but not all common features implemented at a particular capability level for a particular process, the organization usually is operating at the lowest completed capability level for that process. For example, at capability level 2, if the Tracking Performance common feature is lacking, it will be difficult to track project performance. If a common feature is in place, but not all its preceding ones (i.e.-those at lower capability levels), the organization may not reap the full benefit of having implemented that common feature. An assessment team should take this into account in assessing an organization's individual processes.

In the case of improvement, organizing the practices into capability levels provides an organization with an improvement route map should it desire to enhance its capability for a specific process. For these reasons, the generic practices are grouped into common features which are ordered by capability levels.

In either case, an assessment should determine the capability levels for all of the different processes within the assessment scope. Processes can, and probably will, exist at different levels of capability. The organization will then be able to utilize this process-specific information to focus the improvements of its processes. The priority and sequence of the improvement of the organization's software processes should take into account their business goals.

Annex D (informative)

Example process with generic practices elaborated

The following elaboration of “Establish project plan” is intended to help assessors in recognizing and understanding how to judge the adequacy of implementation of generic practices. There are many ways to implement a process, and there are interactions between processes and practices that can best be identified by discussing subtleties of implementation in a specific context. The discussion of generic practice 2.1.6, for example, identifies some of the complexities to be considered when looking at large project versus small project contexts.

PRO.2 Establish project plan

The purpose of the *Establish project plan* process is to establish reasonable plans for performing the software engineering and as a basis for managing the software project.

Project plans typically document

- project purpose and objectives;
- work products to be developed;
- software life cycle model;
- software estimates;
- project risks and mitigation plans;
- schedule;
- resources allocated to the project activities;

Inputs to this process can include the project's software life cycle model produced by process, "Plan project life cycle" PRO.1.

Project planning involves developing estimates for the work to be performed, establishing the necessary commitments, and defining the plan to perform the work. It begins with a statement of the work to be performed and other constraints and goals that define and bound the software project, those established by the practices of the “Identify customer needs” CUS.3 and “Manage requirements” PRO.4 processes. The planning process includes steps to estimate the size of the work products and the resources needed, produce a schedule, identify and assess an initial set of risks, and negotiate commitments. Iterating through these steps may be necessary to establish the plan for the project.

This plan provides the basis for performing and managing the project's activities and addresses the commitments to the software project's customer according to the resources, constraints, and capabilities of the software project.

Level 1: Performed-Informally Level

Common Feature 1.1: Performing Base Practices

1.1.1 Perform the process. Perform the base practices in implementing the process to provide work products and/or services to a customer.

Note: The customer of the process may be internal or external to the organization.

The base practices for “Establish project plan” follow.

PRO.2.1 Develop work breakdown structure. Develop a work break down structure relating project tasks and sequence with the resources required to accomplish them.

Note: This work break down structure will be critically dependent on having a statement of work for the software project that covers scope of the work, technical goals and objectives, identification of customers and end users, imposed standards, assigned responsibilities, cost and schedule constraints and goals, dependencies between the software project and other organizations, resource constraints and goals, and other constraints and goals for development and/or maintenance.

PRO.2.2 Identify project standards. Identify the software standards which will guide the project's software activities, consistent with the needs of the project.

Note: Examples of software standards and procedures include software planning, software configuration management, software quality assurance, risk management, software design, problem tracking and resolution, and software measurement.

PRO.2.3 Identify specialized facilities. Identify any specialized tools, equipment, or rooms beyond those normally available which will be needed to meet any unusual technical requirements of the project.

Note: Examples include target hardware, simulators, specialized test equipment, and test labs.

Process, "Provide software engineering environment" ORG.6 deals with providing the (normal) software engineering environment. Specialized facilities may be needed due to security requirements of the project.

Plan for the project's software engineering facilities and support tools, basing estimates of capacity requirements for these facilities and support tools on the size estimates of the software work products and other characteristics.

Examples of software engineering facilities and support tools include software development computers and peripherals, software test computers and peripherals, target computer environment software, and other support software.

PRO.2.4 Determine reuse strategy. Identify opportunities for reuse, analyze their impacts, and determine the strategy for reuse.

Note: This base practice benefits from process, "Enable reuse" ORG.5.

PRO.2.5 Develop project estimates. Develop estimates of what is needed to satisfy the software requirements for the entire software life cycle.

Note: Parameters to estimate include size, effort, cost, schedule, and resources

Examples of types of work products and activities for which size estimates are made include operational software and support software, deliverable and non deliverable work products, software and non software work products (e.g., documents), and activities for developing, verifying, and validating work products.

PRO.2.6 Identify initial project risks. Identify, assess, and document an initial (or baseline) set of project risks and their mitigation plans.

Note: Software project risks include risks to staying within budgeted software size, cost, effort, schedule; risks in availability of resources; and technical risks.

See process, "Manage risks" PRO.6.

Identify, assess, and document the software risks associated with the cost, resource, schedule, and technical aspects of the project. Analyze and prioritize risks based on their potential impact to the project. Identify contingencies for the risks. Examples of contingencies include schedule buffers, alternate staffing plans, and alternate plans for additional computing equipment.

PRO.2.7 Identify project measures. Identify the basic set of status and other measures which will be used to track project progress and help determine if the project is meeting its objectives.

Note: Examples of software size measurements include function points, feature points, lines of code, number of requirements, and number of pages.

PRO.2.8 Establish project schedule. Establish the project schedule, based on the software life cycle model, work breakdown structure, estimates, and risk mitigation plans.

Note: Examples of software life cycles include waterfall, overlapping waterfall, spiral, serial build, and single prototype/overlapping waterfall.

PRO.2.9 Establish project commitments. Establish commitments to the estimates and plans with all affected groups.

Note: Review software project commitments made to individuals and groups external to the organization with senior management.

PRO.2.10 Document project plans. Document the results of the activities in this process within the project plans.

Note: This includes documenting the project's software life cycle model.

See discussion of generic practice 2.1.6 later.

The software plan covers:

- The software project's purpose, scope, goals, and objectives.
- Selection of a software life cycle.
- Identification of the selected procedures, methods, and standards for developing and/or maintaining the software.
- Identification of software work products to be developed.
- Size estimates of the software work products and any changes to the software work products.
- Estimates of the software project's effort and costs.
- Estimated use of critical computer resources.
- The software project's schedules, including identification of milestones and reviews.
- Identification and assessment of the project's initial software risks.
- Plans for the project's software engineering facilities and support tools.

Level 2: Planned-and-Tracked Level

Common Feature 2.1: Planning Performance

2.1.1 Allocate resources. Allocate adequate resources (including people) for performing the process.

Note: Resource management is described in project process, "Manage resources and schedule" PRO.7.

Where feasible, make experienced individuals, who have expertise in the application domain of the software project being planned, available to develop the software plan.

Initiate software project planning in the early stages of, and in parallel with, the overall project planning.

2.1.2 Assign responsibilities. Assign responsibilities for developing the work products and/or providing the services of the process.

Note: Designate a project software manager to be responsible for negotiating commitments and developing the project's software plan.

Assign responsibilities for developing the software plan.

The project software manager, directly or by delegation, co-ordinates the project's software planning.

Partition the software work products and activities, and assign responsibilities to software managers in a traceable, accountable manner. Examples of software work products include work products for delivery to the external customer or end users, as appropriate; work products for use by other engineering groups; and major work products for internal use by the software engineering group.

The software engineering group participates on the project proposal team.

Involve the software engineering group in proposal preparation and submission, clarification discussions and submissions, and negotiations of changes to commitments that affect the software project.

Assign responsibilities and negotiate commitments to procure or develop software engineering facilities and support tools.

2.1.3 Document the process. Document the approach to performing the process in standards and/or procedures.

Note: Employee participation in developing standards and procedures is essential to creating a usable process definition. The people who perform the process are its owners.

See ORG2.9 “Establish policy” for the establishment of organizational policies that should be followed, specifically as instantiated in planning. Follow a written organizational policy for planning a software project.

Review the statement of work according to a documented procedure, which will typically specify reviews by the project manager, the project software manager, the other software managers, and other affected groups.

Develop the project's software plan according to a documented procedure, which will typically specify that:

– the software plan is based on and conforms to the customer's standards, as appropriate; the project's standards; the approved statement of work; and the allocated requirements.

– plans for software-related groups and other engineering groups involved in the activities of the software engineering group are negotiated with those groups, the support efforts are budgeted, and the agreements are documented.

– plans for involvement of the software engineering group in the activities of other software-related groups and other engineering groups are negotiated with those groups, the support efforts are budgeted, and the agreements are documented.

– the software plan is reviewed by the project manager, the project software manager, the other software managers, and other affected groups.

For PRO.2.5, derive estimates for the size of the software work products (or changes to the size of software work products) according to a documented procedure, which will typically specify that:

- size estimates are made for all major software work products and activities;
- software work products are decomposed to the granularity needed to meet the estimating objectives;
- historical data are used where available;
- size estimating assumptions are documented;
- size estimates are documented, reviewed, and agreed to. Examples of groups and individuals who review and agree to size estimates include the project manager, the project software manager, and the other software managers.

Similarly, derive estimates for the software project's effort and costs according to a documented procedure, which will typically specify that:

- estimates for the software project's effort and costs are related to the size estimates of the software work products (or the size of the changes);
- productivity data (historical and/or current) are used for the estimates when available, and sources and rationale for these data are documented.;
- the productivity and cost data are from the organization's projects when possible;
- the productivity and cost data take into account the effort and significant costs that go into making the software work products. Examples of significant costs that go into making the software work products include direct labour expenses, overhead expenses, travel expenses, and computer use costs;
- effort, staffing, and cost estimates are based on past experience;
- similar projects are used when possible;
- time phasing of activities is derived;
- distributions of the effort, staffing, and cost estimates over the software life cycle are prepared;
- estimates and the assumptions made in deriving the estimates are documented, reviewed, and agreed to.

Derive estimates for the project's critical computer resources according to a documented procedure, which will typically specify that:

- critical computer resources for the project (e.g. computer memory capacity, computer processor use, and communications channel capacity) are identified;
- estimates for the critical computer resources are related to the estimates of: the size of the software work products, the operational processing load, and the communications traffic;
- estimates of the critical computer resources are documented, reviewed, and agreed to.

Critical computer resources may be in the host environment, in the integration and testing environment, in the target environment, or in any combination of these.

For PRO.2.8, derive the project's software schedule according to a documented procedure, which will typically specify that:

- the software schedule is related to: the size estimate of the software work products (or the size of changes), and the software effort and costs;
- the software schedule is based on past experience;
- similar projects are used when possible;
- the software schedule accommodates the imposed milestone dates, critical dependency dates, and other constraints;
- the software schedule activities are of appropriate duration and the milestones are of appropriate time separation to support accuracy in progress measurement;
- assumptions made in deriving the schedule are documented;
- the software schedule is documented, reviewed, and agreed to.

2.1.4 Provide tools. Provide appropriate tools to support performance of the process.

Note: Examples of support tools for planning include spreadsheet programs, cost estimation models, and project planning and scheduling programs.

2.1.5 Ensure training. Ensure that the individuals performing the process are appropriately trained in how to perform the process.

Note: Train the software managers, software engineers, and other individuals involved in software planning **i**n the software estimating and planning procedures applicable to their areas of responsibility.

2.1.6 Plan the process. Plan the performance of the process.

Note: Project planning is described in this process, "Establish project plan" PRO.2, so this could be considered a “recursive” instantiation of planning. Generic practice 2.1.6 refers to the planning of the process, "Establish project plan" PRO.2 rather than the actual software project plan itself.

In large software projects, where millions of dollars/yen/pounds/etc., may be spent on a proposal or concept exploration phase, it is appropriate to “plan the plan.” In more common projects, this “plan the process” practice would probably be realized more at the organizational level in terms of establishing strategic directions for products and planning guidelines, which are then monitored by senior management.

Common Feature 2.2: Disciplined Performance

2.2.1 Use plans, standards, and procedures. Use documented plans, standards, and/or procedures in implementing the process.

Note: The standards and procedures used were documented in 2.1.3. The plans used were documented in 2.1.6.

2.2.2 Do configuration management. Place work products of the process under version control or configuration management, as appropriate.

Note: The process for configuration management is described in support process, "Perform configuration management" SUP.2.

The following planning work products would typically be considered

- statement of work;
- software plan;
- software planning data.

Common Feature 2.3: Verifying Performance

2.3.1 Verify process compliance. Verify compliance of the process with applicable standards and/or procedures.

Note: The applicable standards and procedures were documented in 2.1.3 and used in 2.2.1. The quality assurance process is described in support process, "Perform quality assurance" SUP.3.

At a minimum, the reviews verify conduct of the following activities according to the applicable procedures:

- activities for software estimating and planning;
- activities for reviewing and making project commitments;
- activities for preparing the software plan;

2.3.2 Audit work products. Verify compliance of work products with the applicable standards and/or requirements.

Note: Requirements are identified in Customer-Supplier process, "Identify customer needs" CUS.3, and managed in project process, "Manage requirements" PRO.4.

At a minimum, the audits verify the content of the software plan complies with the applicable standards and procedures.

Common Feature 2.4: Tracking Performance

2.4.1 Track with measurement. Track the status of the process against the plan using measurement.

Note: The use of measurement implies that the measures have been defined and selected, and data has been collected.

Note that the measures identified in PRO.2.7, “Identify project measures” are the measures used to manage the software project rather than the measures identified here for tracking the status of the planning process (PRO.2). As was discussed for 2.1.6, measurement of large project planning efforts may be implemented directly by a proposal team; more typical projects will probably aggregate and track planning overhead more indirectly.

Examples of measurements include completion of milestones for the software project planning activities compared to the plan; and work completed, effort expended, and funds expended in the software project planning activities compared to the plan.

Review the activities for software project planning with the project manager on both a periodic and event-driven basis ensuring that affected groups are represented. Topics for review include:

- status and current results of the software project planning activities against the software project's statement of work and customer requirements;
- dependencies between groups;
- conflicts and issues not resolvable at lower level;
- software project risks;
- assignment, review, and tracking of action items.

Prepare and distribute a summary report from each meeting to the affected groups and individuals.

Review the activities for software project planning with senior management on a periodic basis for the purpose of providing awareness of, and insight into, planning activities at an appropriate level of abstraction and in a timely manner. The time between reviews should meet the needs of the organization and may be lengthy, as long as adequate mechanisms for exception reporting are available. Review topics include:

- the technical, cost, staffing, and schedule performance;
- conflicts and issues not resolvable at lower levels;
- software project risks;
- assignment, review, and tracking of action items.

Prepare and distribute a summary report from each meeting to the affected groups and individuals.

2.4.2 Take corrective action. Take corrective action as appropriate when progress varies significantly from that planned.

Note: Progress may vary because estimates were inaccurate, performance was affected by external factors, or the requirements, on which the plan was based, have changed. Corrective action may involve changing the process or changing the plan or both.

Record software planning data for possible use in future planning/replanning as part of 2.4.1. Information recorded includes the estimates and the associated information needed to reconstruct the estimates and assess their reasonableness.

Level 3: Well-Defined Level

Common Feature 3.1: Defining a Standard Process (Organization-Level Common Feature)

3.1.1 Standardize the process. Document a standard process or family of processes for the organization, which describes how to implement the base practices for the process.

Note: The critical distinction between 2.1.3 and 3.1.1 is the scope of application of the standards and procedures. In 2.1.3, the standards and procedures may be used in only a specific instance of the process, e.g., in a particular project. In 3.1.1, standards and procedures are being established at an organizational level for common use. The process definition process is described in the "Organization" process category, "Define the process" ORG.2.

For “Establish project plan,” the standard process is for the planning of software projects. It may specify (or encourage) cost models, such as COCOMO, PRICE-S, and SLIM. It may specify (or encourage) structured brainstorming/consensus techniques for estimating, such as the Delphi method and nominal group technique. It may specify (or encourage) life cycle models, such as spiral or incremental development.

3.1.2 Tailor the standard process. Tailor the organization's standard process family to create a defined process which addresses the particular needs of a specific use.

Note: The phrase “which addresses the particular needs of a specific use” caters to the general case of organization-level, as opposed to project-level, processes. For defined processes at the project level, the tailoring addresses the particular needs of the project. The organization's standard process family is documented in 3.1.1 and ORG.2.

Select the models, techniques, and tools that are supported by the organization for planning software projects and which most directly address the application domain and customer needs (i.e., specific planning or reporting requirements may be imposed by the customer).

Common Feature 3.2: Performing the Defined Process

3.2.1 Use a well-defined process. Use a well-defined process in implementing the process.

Note: The defined process will typically be tailored from the organization’s standard process, as described in 3.1.2. A well-defined process is one with inputs, entry criteria, tasks, validation, outputs, and exit criteria that are documented, consistent, and complete.

For “Establish project plan,” this means use a well-defined planning process that results in a clear course of action with the appropriate reviews and checkpoints defined to make the planning process responsive to the changing environment.

3.2.2 Perform peer reviews. Perform peer reviews of appropriate work products of the process.

Note: The process for peer reviews is described in support process, "Perform peer reviews" SUP.5.

The software engineering group participates with other affected groups in the overall project planning throughout the project's life; specifically, they review the project-level plans.

The software engineering group reviews the project's proposed commitments. Examples of project commitments include the project's technical goals and objectives; the system and software technical solution; the software budget, schedule, and resources; and the software standards and procedures.

Estimates that are reviewed include those for size, cost, effort, critical computer resources, and schedule.

Review the risks that have been identified.

All affected groups review the plans for software engineering facilities and support tools.

3.2.3 Use well-defined data. Use data on performing the defined process to manage the defined process.

Note: This is an evolution of 2.4.2; corrective action taken here is based on a well-defined process, which has objective criteria (see 3.2.1) for determining progress.

For “Establish project plan,” this implies using measurements that are defined and used across projects in the organization and using planning data (for estimating) based on historical performance and that is maintained in an organizational database.

Level 4: Quantitatively-Controlled Level

Common Feature 4.1: Establishing Measurable Quality Goals

4.1.1 Establish quality goals. Establish measurable quality goals for the work products of the organization's standard process family.

Note: These quality goals can be tied to the strategic quality goals of the organization, the particular needs and priorities of the customer, or to the tactical needs of the project.

For “Establish project plan,” this implies planning to build products that are aligned with strategic business objectives, e.g., in a particular market or product line that the organization has chosen to compete in.

Common Feature 4.2: Objectively Managing Performance

4.2.1 Determine process capability. Determine the process capability of the defined process quantitatively.

Note: This is a quantitative process capability based on a well-defined (3.1.1) and measured process.

For “Establish project plan,” this implies understanding how good the planning process is.

4.2.2 Use process capability. Take corrective action as appropriate when the process is not performing within its process capability.

Note: Special causes of variation, identified based on an understanding of process capability, are used to understand when and what kind of corrective action is appropriate. This is an evolution of 3.2.3, with the addition of quantitative process capability to the defined process.

For “Establish project plan,” this implies identifying when the planning process is not working effectively and will be realized more in replanning than in initial planning.

Level 5: Continuously-Improving Level

Common Feature 5.1: Improving Organizational Capability (Organization-Level Common Feature)

5.1.1 Establish process effectiveness goals. Establish quantitative goals for improving process effectiveness of the standard process family, based on the business goals of the organization and the current process capability.

Note: For “Establish project plan,” this implies establishing goals such as cost/performance indices that maximize the accuracy of plans within the limits of planning capability.

5.1.2 Continuously improve the standard process. Continuously improve the process by changing the organization's standard process family to increase its effectiveness.

Note: Changes to the organization's standard process family may come from innovations in technology or incremental improvements. Innovative improvements will usually be externally driven by new technologies. Incremental improvements will usually be internally driven by improvements identified during tailoring (3.1.2) or defect prevention (5.2.2) activities.

The effect of improving the standard process is to attack common causes of variation. Special causes of variation are controlled in 4.2.2. Common causes of variation across instantiations of the process are changed in this practice.

For “Establish project plan,” this deploying new estimating techniques, cost models, etc., that have been demonstrated to be effective.

Common Feature 5.2: Improving Process Effectiveness

5.2.1 Perform causal analysis. Perform causal analysis of defects.

5.2.2 Eliminate defect causes. Eliminate the causes of defects in the defined process selectively.

Note: Defect causes are selectively eliminated because it may be impractical to perform causal analysis (5.2.1) on all defects, so some screening criteria may be used. Also note the desirability of eliminating similar, as yet undiscovered, defects in the product, as well as eliminating the cause of the defect.

5.2.3 Continuously improve the defined process. Continuously improve process performance by changing the defined process to increase its effectiveness.

Note: The improvements may be based on incremental improvements (5.2.2) or new technologies (perhaps as part of pilot testing) in ORG.3.8). Improvements will typically be driven by the goals established in 5.1.1.

Annex E (informative)

Mapping to ISO 12207

E.1 Introduction

ISO 12207 is a stand-alone document which is primarily intended for use in a contract situation involving the acquisition/supply of a system that contains software. Thus for example, it contains explicit links between processes which determine a "calling sequence" of processes, and assigns responsibilities for activities/products between parties to the contract - elements which would be inappropriate for this part of the International Standard, given its differences in scope and intended use. Therefore, it is inappropriate to try to individually map every ISO 12207 process task to a particular practice in this part of the International Standard. Instead, the mapping has been performed on the ISO 12207 views, process categories and processes. Each mapping is summarized by a table, followed by a textual explanation briefly outlining the rationale for mapping each entry.

E.2 Mapping ISO 12207 views

There are five basic views in ISO 12207 - contract, management, operating, engineering and supporting. These views depict the ISO 12207 software life cycle processes and their relationships under different views of the usage of the standard. Although there is no direct equivalent in this part of the International Standard, these ISO 12207 views can be mapped as shown in table 7.

Table 7 – ISO 12207 views mapping

12207 View	Process Categories
Contract	Customer-Supplier
Management	Project
Operating	Customer-Supplier
Engineering	Engineering
Supporting	Support

E.2.1 Contract view

The contract view defines the tasks of the acquirer and supplier from the contractual viewpoint. This part of the International Standard doesn't contain an explicit supplier process, however the acquirer contract-establishing processes are in the “Customer-Supplier” process category.

E.2.2 Management view

Under the management view, the acquirer, developer, operator etc. manage their respective processes for the software project. T part of the International Standard “Project” process category consists of processes which establish the project and manage its resources. In addition, each process in this part of the International Standard has a set of common level generic practices which focus on planning and tracking the performance of the process.

E.2.3 Operating view

Under the operating view, the operator provides software operation services for the users. This part of the International Standard “Customer-Supplier” process category consists of processes that work directly with the customer, including supporting operation of the software and providing customer support.

E.2.4 Engineering view

Under the engineering view, the developer or maintainer conduct their respective tasks to produce or modify software products. This part of the International Standard “Engineering” process category consists of processes that directly specify, implement, or maintain a system and software product.

E.2.5 Supporting view

Under the supporting view, supporting services (such as configuration management and quality assurance) are provided to others in fulfilling specific, unique tasks. This part of the International Standard “Support” process category consists of process (including configuration management and quality assurance) which enable and support the performance of the other processes on a project.

E.3 Mapping ISO 12207 process categories

There are three process categories in ISO 12207 (primary, supporting and organizational) and five in this part of the International Standard (customer-supplier, engineering, project, support, and organization). The ISO 12207 process categories correspond to the process categories as shown in table 8.

E.3.1 Primary process category

The ISO 12207 Primary process category contains processes that initiate or perform development, operation and maintenance activities. It corresponds to the “Engineering” process category, although operation activities are in the “Customer-Supplier” process category since the customer is directly involved.

E.3.2 Supporting process category

The ISO 12207 Supporting process category contains processes that support other processes as an integral part with a distinct purpose and contribute to the success and quality of the software project. It corresponds to the “Support” process category, although joint reviews with the customer are in the “Customer-Supplier” process category, since again, the customer is directly involved.

E.3.3 Organizational process category

The ISO 12207 Organizational process category contains processes which are established at the organizational level to provide an infrastructure of viable processes and competent personnel. It corresponds to the “Organization” process category, although some of the management activities are in the “Project” process category, because they co-ordinate and manage the project and its resources (in addition, each this part of the International Standard process has a set of common level generic practices which focus on planning and tracking performance of the process).

Table 8 – Process category mapping

ISO 12207 Process Category	Process Category
Primary	Engineering Customer - Supplier
Supporting	Support Customer - Supplier
Organizational	Organization + Project

E.4 Mapping ISO 12207 processes

There are seventeen processes in ISO 12207 and thirty five in this part of the International Standard. ISO 12207 processes are comprised of activities and tasks, while this part of the International Standard processes consist of practices (base practices, which address the purpose of the process, and generic practices, which help to implement and institutionalize the process and enhance its capability). Some ISO 12207 activities and tasks map to base practices, while others, particularly those concerned with implementing and planning the process are covered by generic practices.

The purpose and activities of ISO 12207 processes provide a better basis than tasks for mapping to this part of the International Standard, because, given the differences in scope and intended use of the two documents, ISO 12207 tasks are generally at a lower level of detail than this part of the International Standard practices. Using this basis, the seventeen ISO 12207 processes map to this part of the International Standard as shown in table 9.

Table 9 – Process mapping

ISO 12207 Process	Processes
Acquisition	Acquire software products and/or service Establish contract
Supply	Establish contract Perform joint audits and reviews Package, deliver, and install the software Support operation of software Plan project life cycle Establish project plan Manage Project Resources Manage quality
Development	Develop software requirements Develop software design Integrate and test software Integrate and test system Package, deliver, and install the software Perform Audits and Reviews Plan project life cycle Establish project plan
Operation	Support operation of software
Maintenance	Maintain system and software
Documentation	Develop documentation
Configuration Management	Perform configuration management
Quality Assurance	Perform quality assurance
Verification	Level 2 Generic Practices Perform peer reviews
Validation	(to be completed)
Joint Review	Perform Audits and Reviews Manage Project Resources
Audit	Perform quality assurance Perform joint audits and reviews
Problem Resolution	Problem Resolution
Management	Level 2 Generic Practices Establish project plan Manage Project Resources
Infrastructure	Level 3 Generic Practices Define the process Provide Development Environment Provide work facilities
Improvement	Level 3, 4, 5 Generic Practices Define the process Improve Process
Training	Level 2 Generic Practices Training
Tailoring	Annex A Level 3 Generic Practices

E.4.1 Acquisition process

The Acquisition process contains the tasks of the acquirer, such as identifying the need to acquire a software produce/service, considering the options for acquisition, preparing a request proposal, selecting a supplier, negotiating a contract etc. The “Acquire software product and/or service” CUS.1 process covers these activities, except contract negotiation, which is covered by the “Establish contract” CUS.2 process.

E.4.2 Supply process

The Supply process contains the activities and tasks of the supplier; preparing a proposal, negotiating a contract, developing and executing plans for managing the project, co-ordinating reviews with the supplier, and supporting the delivered product. This part of the International Standard does not contain a separate Supplier process, but all of the above ISO 12207 activities are covered by processes in the Customer-Supplier and Project Process Categories.

E.4.3 Development process

The Development process contains the activities for requirements analysis, design, coding, integration, testing, installation and acceptance of the software. In this part of the International Standard, each of these activities is a process. Requirements analysis, design, coding, integration and testing processes are all part of the “Engineering” process category. Software Installation is performed through the "Package, deliver, and install the software" CUS.5 process. Support customer acceptance review is part of the "Perform joint audits and reviews" CUS.4 process. The ISO 12207 Process Implementation activity maps to the "Plan project life cycle" PRO.1 and "Establish project plan" PRO.2 processes which deal with establishing a project and managing its resources.

E.4.4 Operation process

The Operation process covers the operation of the software and operational support to users. This part of the International Standard "Support operation of software" CUS.6 process supports correct, continuing, and effective operation of the software for the duration of its intended operation.

E.4.5 Maintenance process

The Maintenance process is concerned with modifying existing software (as a result of an error, deficiency, problem or changing need) while preserving its integrity. The "Maintain system and software" ENG.7 process has a similar orientation.

E.4.6 Documentation process

The Documentation process is for recording information produced by a life cycle process or activity. The "Develop documentation" SUP.1 process deals with designing, developing, distributing and maintaining project documentation.

E.4.7 Configuration management process

The Configuration Management process applies administrative and technical procedures to control modifications and releases of identified configuration items. The "Configuration Management" process is likewise concerned with maintaining the integrity of products of the software project.

E.4.8 Quality assurance process

The Quality Assurance process is for providing adequate assurance that the processes and software products in the project life cycle conform to their specified requirements and adhere to their established plans. The "Quality Assurance" process provides appropriate visibility into the process and products of the project.

E.4.9 Verification process

The Verification process determines whether the requirements for a system or software are complete and correct. It maps to the "Peer Review" process, and is also embedded in each process through the level 2 generic practices on reviews.

E.4.10 Validation process

The Validation process determines whether the requirements and the final, as built system or software fulfils its specific intended use. This involves developing, documenting and implementing a validation plan for the project, preparing selected test requirements, test cases and test specifications for analyzing test results and testing the software as appropriate in selected areas of the target environment.

E.4.11 Joint review process

The Joint Review process is for evaluating the status and products of an activity or phase of a project and consists of project management reviews and technical reviews. In this part of the International Standard, project management and technical reviews are performed as part of the "Manage Project Resources" process. Joint reviews involving the customer are performed as part of the "Perform Audits and Reviews" process.

E.4.12 Audit process

The audit process determines compliance with the requirements, plans and contract as appropriate. In this part of the International Standard, customer audits are covered in the “Perform joint audits and reviews” CUS.4 process while other audits are covered in the “Perform quality assurance” SUP.3 process.

E.4.13 Problem resolution process

The Problem Resolution process is for analyzing and removing problems uncovered during development, operation, maintenance etc. The "Problem Resolution" process ensures that all discovered problems are analyzed and removed and trends are identified.

E.4.14 Management process

The Management process can be used by any party to manage its respective processes. In this part of the International Standard, each process has a set of level two generic practices which are concerned with planning and tracking the performance of the process. In addition to these, Project management is specifically covered by several processes in the "Project" process category, including "Establish project plan" PRO.2, which establishes the project plans and "Manage resources and schedule" PRO.7, which co-ordinates and manages the project's resources and tracks progress against the plans.

E.4.15 Infrastructure process

The Infrastructure process is used to establish and maintain the infrastructure needed for any other processes, including software, tools, techniques, standards etc. In this part of the International Standard, each process has a set of level three generic practices which deal with defining, standardizing and tailoring the process. The process "Define the process" ORG.2 is an integral part of these practices since it is concerned with building a reusable library of process definitions. The "Provide Development Environment" process deals with providing an integrated set of software development tools for use by projects. The "Provide work facilities" ORG.7 process is concerned with providing a secure and reliable environment in which software project activities may be carried out.

E.4.16 Improvement process

The Improvement process is for establishing, assessing, measuring, controlling and improving a software life cycle process. In this part of the International Standard, each process has a common set of generic practices at capability levels three, four and five, which are concerned respectively with defining, measuring/analyzing and optimizing the process. These practices are also related to the processes "Define the process" ORG.2 and "Improve the process" ORG.3.

E.4.17 Training process

The Training process is for providing and maintaining trained personnel. In this part of the International Standard, each process has a common level two generic practice which is concerned with training individuals in how to perform the process. In addition, the "Training" process is about providing the organization and projects with individuals who possess the needed skills and knowledge to perform their roles effectively.

E.4.18 Tailoring process

The Tailoring process is for performing basic tailoring of the ISO 12207 standard. Advice on tailoring this part of the International Standard is not contained within Annex A.

Annex F (informative)

Mapping to ISO 9001

This annex maps ISO 9001:1994(E) to this part of the International Standard. The mapping is done at an overview level because the ISO 9001 requirements are typically not as detailed as the practices in this part of the International Standard. The scope and the audience of the documents also differ in many essential points, such as use in contractual situations.

The mapping in table 10 covers the general requirements of ISO 9001 versus the process categories and processes. Many of the process categories, processes and practices are much more specific to software development than the ISO 9001 requirements. Because of different architectures of the documents, many of the ISO 9001 requirements are also covered by common features or generic practices.

Table 10 – ISO 9001 mapping

ISO 9001 requirements	Process categories and processes
4.1 Management responsibility	Engineer the business Manage quality (build project teams) Assess customer satisfaction
4.2 Quality system	Manage quality Perform quality assurance Define the process (Improve the process)
4.3 Contract review	Establish contract Identify customer needs Develop system requirements and design Manage risks (Perform joint audits and reviews)
4.4 Design control	Identify customer needs Establish project plan Build project teams Manage requirements Manage resources and schedule Manage risks Develop system requirements and design Develop software requirements Develop software design (Enable reuse)
4.5 Document and data control	Develop documentation Define the process
4.6 Purchasing	Manage subcontractors
4.7 Control of customer-supplied product	Develop system requirements and design (Acquire software product and/or service)
4.8 Product Identification and traceability	Develop documentation Perform configuration management Enable reuse

Table 10 (concluded)**– ISO 9001 mapping**

ISO 9001 requirements	Process categories and processes
4.9 Process control	Implement software design Provide software engineering environment Provide work facilities (Provide customer service)
4.10 Inspection and testing	Integrate and test software Integrate and test system Perform peer reviews
4.11 Control of inspection, measuring and test equipment	Integrate and test software Integrate and test system Provide software engineering environment
4.12 Inspection and test status	Integrate and test software Integrate and test system Perform configuration management (Perform problem resolution)
4.13 Control of non conforming product	Perform configuration management
4.14 Corrective and preventive action	Perform problem resolution Improve the process
4.15 Handling, storage, packaging, preservation and delivery	Package, deliver, and install the software (Support operation of software) Perform configuration management
4.16 Control of quality records	Mostly covered by common features 2.3 and 2.4 (Assess customer satisfaction) (Develop documentation)
4.17 Internal quality audits	Perform quality assurance
4.18 Training	Perform training
4.19 Servicing	Provide customer service Maintain system and software
4.20 Statistical techniques	Covered mostly by common features 4.1 and 4.2 (Improve the process)

Annex G (informative)

Derivation and traceability

The objective of this International Standard is to encourage predictable quality products, optimum productivity, and to promote a repeatable software process. To achieve this objective, the practices in this standard were derived from the many standards and software practices that have already been defined by national and international bodies such as ISO and IEEE. Relevant standards were considered to be those that address methods for improving the quality of the software process and which were recognized nationally and/or internationally by professionals in the software industry.

The following lists the sources that were used.

Bell Communications Research, Inc., *Quality Program Specification for Surveillance Management Process(SMP) - Software (General),* QPS 88.001, Issue 1.

Institute of Electrical and Electronics Engineers, Inc., *IEEE Software Engineering Standards Collection,* April 1991.

International Organization for Standardization, *Quality systems - Model for quality assurance in design, development, production, installation, and servicing*, ISO 9001 : 1994, Revised edition, July 1994.

International Organization for Standardization, *Quality management and quality assurance standards - Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software,* ISO 9000-3 : 1991(E), First edition, June 1991.

International Organization for Standardization, *Quality management and quality system elements - Part 2: Guidelines for services,* ISO 9004-2 : 1991 (E), First edition, August 1991.

International Organization for Standardization, *Software Life Cycle Processes,* ISO 12207, Draft International Standard.

National Institute of Standards and Technology, 1993 *Award Criteria -Malcolm Baldrige National Quality Award,* December 1992.

M.C. Paulk, C.V. Weber, S.M. Garcia, M.B. Chrissis, and M. Bush, *Key Practices of the Capability Maturity Model, Version 1.1,* Technical Report CMU/SEI-93-TR-25, Software Engineering Institute, Pittsburgh, Pennsylvania, February 1993.

SDCE - Software Development Capability Evaluation, Dept. of the Air Force, Headquarters Air Force Materiel Command, AFMC Pamphlet 800-61, 24 Nov. 1993, Prepublication version.

SDCCR - Software Development Capability /Capacity Review, ASC Pamphlet 800-5, 11 September 1992, Department of the Air Force, Aeronautical Systems Center (AFMC).

SCE - Software Capability Evaluation Training Guide / SPA - Software Process Assessment Training Guide , Software Engineering Institute, Pittsburgh Pennsylvania.

Compita Ltd., *Software Technology Diagnostic Practitioner's Guideline,* Version 2.5, 1993.

TickIT: A Guide to Software Quality Management System Construction and Certification Using EN29001, Issue 2.0, U.K. Department of Trade and Industry and the British Computer Society, 28 February 1992.

F. Coallier, N. Gammage, A.W. Graydon, *Trillium - Software Process Self-assessment Capability Assessment* , Bell Canada, Bell Northern Research, Northern Telecom, PI Q008, Issue 4.0, March 1993.

ISO/IEC JTC1/SC7/WG7, *Information Technology Software Life Cycle Process,* Work Item 7.21, Committee Draft, March 1993.

ISO/IEC JTC1/SC7/WG8, *Information Technology: Integral Life Cycle Process,* Work Item 7.27.2, Working Draft, September 1992.

Total Quality Management, the European Model for Self-appraisal 1993, Guidelines for Identifying and addressing total quality issues, The European Foundation for Quality Management, Brussels, Belgium.

Annex H (Informative)

Style guide for extended processes

This annex describes the guidelines used in developing the process descriptions found in the base practices and processes in clause 6. These guidelines can be used by those wishing to create extended processes. See Annex A for the requirements for creating variant models.

A process description consists of several components. These guidelines provide guidance as to the format and content of each component. Nomenclature requirements are contained in clause 4.

A process description contains

- a section number;
- the name of process;
- an identifier for process (to aid in external reference);
- a first paragraph stating the purpose of the process;
- optionally, additional paragraphs;
- descriptions of its base practices.

Each base practice description contains

- an identifier for the base practice (to aid in external reference);
- the name of the base practice;
- a statement of what the base practice does;
- and optionally, a note.

For ease of use, the guidelines are contained in two tables. Table 11 contains guidelines covering the components of a process description, and table 12 covers the components of a base practice description. The guidelines applying to a particular component appear to the right of the name of that component.

Table 11 Guidelines for describing processes

Component	Guideline
Section number	1. Each process appears in its own section and is listed in the table of contents to help the reader quickly turn to the page which describes it. That is why process descriptions have a section number. When putting together a variant model containing extended processes, it may be desirable to likewise put each process description in its own section.
Name of process	1. The name of the process should identify what the process does in summary form. 2. In general, the name should be an action-oriented phrase, beginning with a verb which summarizes the action, e.g., "Plan," "Establish," "Build," etc. On the other hand if there is a widely-accepted phrase naming the process, e.g., "configuration management," in which the first word is not a verb, we recommend you add the word "Perform" to the beginning of the name. Thus you would construct the name, "Perform configuration management," in our example. 3. Name of process appears on same line as and immediately after section number.
Identifier for process	1. Processes are grouped into process categories. The identifier for a process identifies its process category and number within that category. When creating an extended process, you may wish to assign it to a process category and a number relative to the other processes in that category, for ease of reference. 2. Identifiers consist of two components: a process category abbreviation and a number. The process categories and their abbreviations are given in Annex B The abbreviations are the initial three letters of first word in the name of the category. If adding new categories, we recommend maintaining this same style for reasons of consistency, as long as no conflicting abbreviations are created. 3. Identifier for process is parenthesized and appears on same line as and immediately after name of process.
Purpose paragraph	1. States the purpose of performing the process (why perform it?). 2. Appears as the first paragraph following the section number, name, and identifier.
Additional paragraphs	1. Add additional paragraphs as appropriate that clarify what was said in the purpose paragraph, describe inputs/outputs to other processes, describe when the process is invoked, and/or give terminology.

Table 12 – Guidelines for describing base practices

Component	Guideline
Identifier	1. To ease external reference, each base practice is given an identifier. The identifier should consist of two components, the first giving the identifier of the process the base practice belongs to, and the second component giving its textual position in the description of the process. For example the first base practice appearing in the first process description of Section 6 is given the identifier, "CUS.1.1," because "CUS.1" identifies the process and it is the first base practice.
Name	1. The name of the base practice should identify what the process does in summary form. 2. The name should be an action-oriented phrase, beginning with a verb which summarizes the action, e.g., "Evaluate," "Select," "Identify," etc. The verb should be followed by a noun (or noun phrase) which indicates the object of that action. 3. Name of base practice appears on same line as and immediately after its identifier.
Statement of what it does	1. The key part of a base practice description is the statement of what it does. It should be a complete sentence. 2. The statement begins immediately after and on the same line as the identifier and name.
Note	1. In one or more paragraphs following the statement, additional information may be provided on the base practice to help in understanding what the base practice is saying.

Consolidated product

Software Process Assessment – Part 3 : Rating Processes

Version 1.00

(Formerly PAG 1.01)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout

Members: Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : ***Concepts and introductory guide***

Product Manager: Terry Rout

– Part 2 : ***A model for process management***

Product Managers: Al Graydon, Mark Paulk

– Part 3 : ***Rating processes***

Product Manager: Harry Barker

– Part 4 : ***Guide to conducting assessment***

Product Manager: Harry Barker

– Part 5 : ***Construction, selection and use of assessment instruments and tools***

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : ***Qualification and training of assessors***

Product Manager: Ron Meegoda

– Part 7 : ***Guide for use in process improvement***

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : ***Guide for use in determining supplier process capability***

Product Manager: John Hamilton

– Part 9 : ***Vocabulary***

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards)

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 3

Rating Processes

Contents

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialling to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being coordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

This part of this International Standard (part 3) is normative

Introduction

This part of the International Standard defines the minimum set of requirements for conducting a software process assessment to ensure that the outputs of the assessment are consistent, repeatable and representative of the process instances assessed.

Process assessment is an activity that is performed either during a process improvement initiative as described in part 7 of this International Standard, or as part of a capability determination exercise as described in part 8. In either case, the formal entry to the assessment process occurs with the compilation of the assessment input which defines the purpose of the assessment (why it is being carried out), the scope of the assessment (which processes are being assessed), what constraints, if any, apply to the assessment, and any aditional information that needs to be gathered. The assessment input also defines the responsibility for carrying out the assessment and gives definitions for any processes within the scope of the assessment that are extended processes (see part 2 of this International Standard).

Process assessment is undertaken to understand an organizational unit's current processes. An assessment may be conducted as a self-assessment, an assisted self-assessment, a self-assessment with external verification, or an independent assessment. A team or an individual approach can be used to perform the assessment. This International Standard does not define one methodology for the performance of an assessment but rather a framework and key elements that an assessment methodology should incorporate.

An assessment is carried out by assessing selected processes against the process model defined in part 2 of this International Standard. This consists of a set of process-specific base practices on one hand and a set of generic practices on the other hand. The generic practices apply across all processes. The generic practices are grouped into five process capability levels that define how well the process is managed. The assessment output includes a set of process capability level ratings for each process instance assessed.

An assessment is implemented with the aid of an assessment instrument, or set of instruments, constructed according to the requirements and guidance contained in part 5 of this International Standard. The process assessment may be carried out by a team with at least one qualified assessor who has the competencies described in part 6 of this International Standard, or on a continuous basis using suitable tools for data collection. Part 4 of this International Standard provides guidance for interpreting the requirements for a team-based assessment.

This part of the International Standard assumes familiarity with the relevant guidance parts of the standard. It is primarily addressed to the qualified assessor and other people, such as the sponsor of the assessment, who need to assure themselves that the requirements have been met. It will also be of value to developers of assessment methods and of tools to support an assessment.

1 Scope

As part of the Software Process Assessment Standard this document establishes the requirements for a software process assessment, for rating, analysing and profiling an assessment, and defines the circumstances under which assessment results are comparable.

Process Assessment is applicable in the following circumstances:

a) by or on behalf of an organization with the objective of understanding the state of its own processes for process improvement;

b) by or on behalf of an organization with the objective of determining the suitability of its own processes for a particular requirement or class of requirements;

c) by or on behalf of one organization with the objective of determining the suitability of another organization's processes for a particular contract or class of contracts.

This document describes a process assessment framework which:

a) encourages self-assessment;

b) takes into account the context in which the assessed processes operate;

c) produces a set of process ratings (a process profile) rather than a pass/fail result;

d) through the generic practices, addresses the adequacy of the management of the assessed processes;

e) is appropriate across all application domains and sizes of organization.

2 Normative references

There are no normative references in this part of the International Standard.

3 Definitions

For the purposes of this part of this International Standard, the definitions in *Software Process Assessment - Part 9 : Vocabulary* apply.

4 Requirements

4.1 General

This clause sets out the requirements placed on an assessment in order to ensure that the assessment outputs are consistent, repeatable and representative of the process instances assessed.

4.2 Defining the assessment input

The assessment input shall be defined prior to an assessment. At a minimum, the assessment input shall define:

– the assessment purpose;

– the assessment scope;

– the assessment constraints;

– the identity of the qualified assessor and any other specific responsibilities for the assessment;

– the definition of any extended processes identified in the assessment scope;

– the identification of any additional information to be collected to support process improvement or process capability determination.

4.3 Responsibilities

The qualified assessor named in the assessment input shall be a member of the assessment team in a team based assessment or shall oversee an assessment conducted using a continuous or tool-based approach.

The qualified assessor shall ensure that the assessment is conducted in accordance with the requirements of this International Standard.

The qualified assessor shall ensure that the organizational unit's processes to be assessed, as defined in the assessment scope, are mapped to the corresponding processes in part 2 of this International Standard or are defined as extended processes.

The qualified assessor shall ensure that the set of process instances selected for assessment is adequate to meet the assessment purpose and will provide outputs that are representative of the assessment scope.

The qualified assessor shall ensure that all of the information required in the assessment output is recorded in a suitable format to fulfil the assessment purpose and that it meets the requirements of this International Standard (see 4.5).

4.4 Assessing and rating processes

4.4.1 Assessing processes

The assessment shall include at least one process instance of each process identified in the assessment scope.

4.4.2 Rating components

4.4.2.1 Base practice rating

A base practice adequacy rating (see 4.4.3.1) or a base practice existence rating (see 4.4.3.2) shall be determined and validated for every base practice within each selected process instance for each process and/or extended process identified within the assessment scope.

4.4.2.2 Generic practice adequacy rating

A generic practice adequacy rating (see 4.4.3.3) shall be determined and validated for every generic practice within each selected process instance of each process and/or each extended process identified within the assessment scope.

4.4.2.3 Process capability level rating

An actual process capability level rating shall be determined for each process instance assessed by aggregating the generic practice adequacy ratings within each capability level.

For each process instance, the actual process capability level ratings shall describe, for each capability level, the proportion of generic practices that were rated at each point on the generic practice adequacy scale in a clear and unambiguous way.

A set of derived process capability level ratings shall be determined for each process identified in the assessment scope by aggregating the actual process capability ratings of the process instances. These derived ratings shall be sufficiently representative of the process capability levels of each process assessed to satisfy the assessment purpose.

For each process identified in the assessment scope, the derived process capability level ratings shall describe, for each capability level, the proportion of generic practices that were rated at each point on the generic practice adequacy scale in a clear and unambiguous way.

NOTE 1 - If only one process instance was identified and assessed then the derived ratings will be the same as the actual ratings of that process instance.

4.4.3 Rating scales

4.4.3.1 Base practice adequacy rating scale

Base practice adequacy shall be rated using the base practice adequacy rating scale defined below.

N; Not adequate: The base practice is either not implemented or does not to any degree contribute to satisfying the process purpose;

P; Partially adequate: The implemented base practice does little to contribute to satisfying the process purpose;

L; Largely adequate: The implemented base practice largely contributes to satisfying the process purpose;

F; Fully adequate: The implemented base practice fully contributes to satisfying the process purpose.

4.4.3.2 Base practice existence rating scale

Base practice existence shall be rated using the base practice existence rating scale defined below:

N; Non-Existent: The base practice is either not implemented or does not produce any identifiable work products;

Y; Existent: The implemented base practice produces identifiable work products.

4.4.3.3 Generic practice adequacy rating scale

Generic practice adequacy shall be rated using the generic practice adequacy rating scale defined below.

N; Not adequate: The generic practice is either not implemented or does not to any degree satisfy its purpose;

P; Partially adequate: The implemented generic practice does little to satisfy its purpose;

L; Largely adequate: The implemented generic practice largely satisfies its purpose;

F; Fully adequate: The implemented generic practice fully satisfies its purpose.

4.4.4 Weighting

Equal weighting shall be applied to each generic practice adequacy rating when aggregating or deriving ratings.

NOTE 2 - Within a given process, each generic practice in the process model in part 2 of this International Standard is regarded of equal importance, both within a common feature, within a capability level and across multiple instances of the process.

4.4.5 Rating references

NOTE 3 - For definitions of the terms used within these subclauses see part 2 of this International Standard.

4.4.5.1 Base practice rating references

A unique reference shall be generated for each base practice rating that includes the process category, the process within the process category, the base practice of the process, and a process instance reference.

4.4.5.2 Generic practice rating references

A unique reference shall be generated for each generic practice rating that includes the process category, the process within that process category, the capability level, the common feature within that capability level, the generic practice within that common feature, and a process instance reference.

4.4.6 Basis for Comparison

In some circumstances it may be desirable to compare the outputs of the assessment of two or more organizational units, or for the same organizational unit at different times. Comparisons of assessment outputs shall be valid only if their process contexts are similar.

NOTE 4 - The sample size used to generate the ratings will influence the precision with which results may be compared.

4.4.7 Assessment Instrument

An assessment instrument that conforms to the requirements set out in part 5 of this International Standard shall be used to support the assessment.

4.5 Recording the assessment output

4.5.1 The process profile

The ratings for the assessed process instances within the assessment scope shall be recorded as the process profile consisting of:

- the actual generic practice ratings and process capability level ratings for each process instance;
- derived generic practice ratings and process capability level ratings for each process within the scope of the assessment;

4.5.2 The assessment record

Any other information which is pertinent to the assessment and which may be helpful in understanding the output of the assessment shall be compiled and recorded as the assessment record . At a minimum, the assessment record shall contain:

- the assessment input;
- the assessment approach that was used;
- the assessment instrument used;
- the base practice ratings for each process instance assessed;
- the date of the assessment;
- the names of team who conducted the assessment;
- any additional information collected during the assessment that was identified in the assessment input to support process improvement or process capability determination;
- any assessment assumptions and limitations.

Consolidated product

Software Process Assessment –
Part 4 : Guide to conducting assessments
Version 1.00

(Formerly PAG 1.01)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

These terms and conditions apply to the set of documents developed by the SPICE Project, and published within the Project as Version 1.0, with the following titles:

– Part 1 : *Concepts and introductory guide*
– Part 2 : *A model for process management*
– Part 3 : *Rating processes*
– Part 4 : *Guide to conducting assessment*
– Part 5 : *Construction, selection and use of assessment instruments and tools*
– Part 6 : *Qualification and training of assessors*
– Part 7 : *Guide for use in process improvement*
– Part 8 : *Guide for use in determining supplier process capability*
– Part 9 : *Vocabulary*

1. You may copy and distribute verbatim copies of any or all of the Documents as you receive them, in any medium, provided that you conspicuously and appropriately publish with each copy a copy of these Terms and Conditions. You may charge a fee for the physical act of transferring a copy.
2. You may copy extracts from these documents in materials for internal or public use, providing you provide clear acknowledgment of the source of the material, by citation or other appropriate means.
3. You may not copy, modify, sub-license, or distribute the Documents except as expressly provided under these Terms and Conditions.

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout

Members: Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : ***Concepts and introductory guide***

Product Manager: Terry Rout

– Part 2 : ***A model for process management***

Product Managers: Al Graydon, Mark Paulk

– Part 3 : ***Rating processes***

Product Manager: Harry Barker

– Part 4 : ***Guide to conducting assessment***

Product Manager: Harry Barker

– Part 5 : ***Construction, selection and use of assessment instruments and tools***

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : ***Qualification and training of assessors***

Product Manager: Ron Meegoda

– Part 7 : ***Guide for use in process improvement***

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : ***Guide for use in determining supplier process capability***

Product Manager: John Hamilton

– Part 9 : ***Vocabulary***

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards)

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 4

Guide to conducting assessments

Contents

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialling to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being coordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

This part of this International Standard (part 4) is not normative: it provides guidance on interpreting the requirements contained in part 3 within a team-based assessment. To aid readability, the requirements from part 3 are embedded at appropriate points in the text of part 4.

Introduction

Process assessment is a means of capturing information describing the current capability of an organization's processes and is initiated as a result of a desire to determine and/or improve the capability of these processes.

This part of the International Standard provides guidance on interpreting the requirements set out in part 3 primarily for use in a team-based assessment. As an aid to understanding, the requirements from part 3 are embedded verbatim in italics at appropriate points within the text of this guide.

Although the guidance in this part of the International Standard is directed at conducting a team-based assessment, the principles for rating processes are the same for a continuous, tool-based assessment. In a continuous assessment, however, the means of collecting data is different.

This document is primarily aimed at:

- the assessment team, who use the document to prepare for the assessment;
- the participants in the assessment, who use the document to help understand the assessment and interpret the results;
- all staff within organizations who need to understand the details and benefits of performing process assessment;
- tool and method developers who wish to develop tools or methods supporting the process assessment model.

1 Scope

The purpose of this document is to provide guidance on meeting the requirements contained in part 3 of this International Standard during a team-based assessment.

This document describes a process assessment framework that

- encourages self-assessment;
- takes account of the process context;
- produces a process rating profile rather than a pass/fail result;
- addresses adequacy of generic practices relative to purpose;
- is appropriate across all application domains and sizes of organization.

Process assessment is applicable in the following circumstances:

a) by or on behalf of an organization with the objective of understanding the state of its own processes for process improvement

b) by or on behalf of an organization with the objective of determining the suitability of its own processes for a particular requirement or class of requirements;

c) by or on behalf of one organization with the objective of determining the suitability of another organization's processes for a particular contract or class of contracts.

2 Normative references

There are no normative references in this part of the International Standard.

3 Definitions

For the purposes of this part of this International Standard, the definitions in *Software Process Assessment - Part 9 : Vocabulary* apply.

4 Overview of process assessment

4.1 Context of process assessment

Process assessment may be invoked either by process improvement or by process capability determination which provide the inputs to assessment and use the output as illustrated in figure 1.

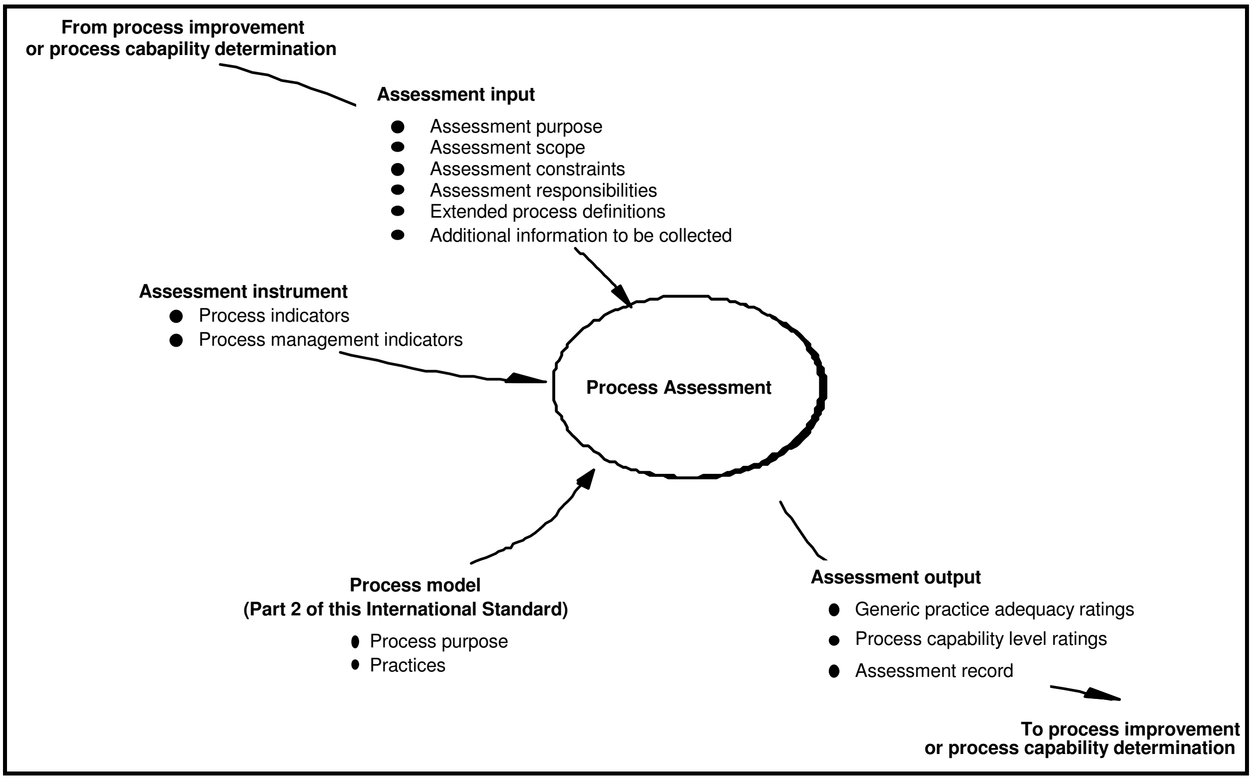

From process improvement
or process cabapility determination
Assessment input
● Assessment purpose
● Assessment scope
● Assessment constraints
● Assessment responsibilities
● Extended process definitions
● Additional information to be collected
Assessment instrument
● Process indicators
● Process management indicators
Process Assessment
Process model
(Part 2 of this International Standard)
● Process purpose
● Practices
Assessment output
● Generic practice adequacy ratings
● Process capability level ratings
● Assessment record
To process improvement
or process capability determination

Figure 1 – Process assessment context

4.1.1 Process assessment

Process assessment is undertaken to understand an organizational unit's current processes. Process assessment deals potentially with all the software related processes (e.g. management, development, maintenance, support) used by an organization. This is accomplished by assessing the organizational unit's processes against the process model described in part 2 of this International Standard. The process model defines, for each process, a set of base practices essential to good software engineering, and a set of generic practices grouped into capability levels. The assessment output consists of a set of generic practice adequacy ratings and process capability level ratings for each process instance assessed together with the assessment record.

Although part 2 of this International Standard covers a range of processes applicable to the software process, in many instances of process assessment a subset of these processes may be selected. For instance the sponsor may wish to focus attention on particular critical processes or processes which are candidates for improvement actions. In process capability determination mode, an acquirer may wish to evaluate the capabilities of suppliers only for the processes related to the tender or contract requirements.

The sponsor may wish a process to include additional base practices to those defined in part 2 of this International Standard or to define an entirely new process - for example to meet industry specific requirements. These are defined as extended processes (see Annex A of part 2 of this International Standard).

The sophistication and complexity of the implemented generic practices for each process utilized within an organizational unit will be dependant upon the context of that process within the organizational unit. For instance, the planning required for a five person project team will be much less than for a fifty person team. This context, recorded in the process context, influences how a qualified assessor should judge an implemented generic practice when assessing its adequacy. The process context also influences the degree of comparability between process capability level ratings.

4.1.2 Process improvement

The assessment output identifies the current process capability level ratings of an organizational unit's processes and forms the basis to plan, prepare, implement and evaluate specific improvement actions, as described in part 7 of this International Standard.

4.1.3 Process capability determination

The assessment output allows an organizational unit to identify, analyse and quantify its strengths, weaknesses and risks, as described in part 8 of this International Standard.

4.2 Process rating scheme

The process rating scheme links process instances, capability levels and generic practices to the defined process purpose.

The process assessment framework is based on assessing a specific process instance. A process instance is a singular instantiation of a process that is uniquely identifiable and about which information can be gathered in a repeatable manner. The actual ratings that are determined for the process instance are therefore repeatable by different qualified assessors.

Each process instance has a set of five process capability level ratings (one at each capability level), each of which is an aggregation of the generic practice adequacy ratings of the one or more generic practices that belong to that level. Hence the generic practice adequacy ratings are the foundation for the rating system.

Generic practice adequacy determines the adequacy of the implemented generic practice at meeting its purpose. It is therefore as much an assessment of the 'effectiveness' of the implemented generic practice within the process context as it is an assessment of its 'conformance' to the process model defined in part 2 of this International Standard.

From the actual ratings of process instances, a number of derived ratings can be determined which consist of aggregations of one or more actual ratings. These derived ratings take account of the variability of individual process instances and may provide better insight into the capability of a process within an organizational unit as a whole.

In order to support the rating of the 'Performed-Informally' Capability Level, part 2 of this International Standard defines base practices for each process. These base practices are rated using the base practice adequacy or base practice existence ratings and the ratings are recorded as part of the assessment record.

4.3 Assessment approaches

4.3.1 Self-assessment

A self-assessment is used by an organization to assess the capability of its software process. The sponsor of a self-assessment is always internal to the organization. A self-assessment may be either team-based or continuous.

4.3.1.1 Team-based assessment

This approach establishes an assessment team from within the organization. An external expert may be brought into an organization:

- to assist the assessment team with the assessment;
- to help the organizational unit understand the concepts expressed by the standard;
- to explain how to use an assessment instrument.

4.3.1.2 Continuous assessment

This approach involves the use of an assessment instrument that supports automated or semi-automated collection of data in the assessment of an organizational unit's process capability. An assessment instrument could be used continuously throughout the software development life cycle: at defined milestones to measure adherence to the process; to measure process improvement progress; or to gather data to facilitate a future assessment. When using the continuous assessment approach there may not be an assessment team as there is for a team-based assessment, but there is still a need for an identified qualified assessor to ensure conformance to the requirements for assessment.

4.3.2 Independent assessment

An independent assessment is an assessment conducted by an assessor who is independent of the organizational unit being assessed. An independent assessment may be conducted, for example, by an organization on its own behalf as independent verification that its assessment programme is functioning properly or by an acquirer who wishes to have an independent assessment output.

In general, the sponsor of an independent assessment will be external to the organizational unit being assessed. The degree of independence, however, may vary according to the purpose and circumstances of the assessment. When the independent assessment is conducted for an acquirer, the sponsor is external to the organization being assessed. If the assessment is being conducted by the organization on its own behalf, however, the sponsor will belong to the same organization as the organizational unit being assessed.

4.4 Assessment stages

Process assessment consists of eight stages (as described below)

- reviewing the assessment input;
- selecting the process instances;
- preparing for assessment;
- collecting and verifying information on practices;
- determining the actual ratings for process instances;
- determining derived ratings;
- validating the ratings;
- presenting the assessment output.

4.5 Success factors for process assessment

The following factors are essential to a successful process assessment.

4.5.1 Commitment

Both the sponsor and owner should commit themselves to the objectives established for an assessment to provide the authority to undertake the assessment within an organization. This commitment requires that the necessary resources, time and personnel are available to undertake the assessment. The commitment of the assessment team is fundamentally important to ensuring that the objectives are met.

4.5.2 Motivation

The attitude of the organization's management, and the method by which the information is collected, has a significant influence on the outcome of an assessment. The organization's management, therefore, needs to motivate participants to be open and constructive. An assessment should be focused on the process, not on the organizational unit members implementing the process. The intent is to make the outcome more effective in an effort to support the defined business goals, not to allocate blame to individuals.

Providing feedback and maintaining an atmosphere that encourages open discussion about preliminary findings during the assessment helps to ensure that the assessment output is meaningful to the organizational unit. The organization needs to recognize that the participants are a principal source of knowledge and experience about the process and that they are in a good position to identify potential weaknesses.

4.5.3 Confidentiality

Respect for the confidentiality of the sources of information and documentation gathered during assessment is essential in order to secure that information. If discussion techniques are utilized, consideration should be given to ensuring that participants do not feel threatened or have any concerns regarding confidentiality. Some of the information provided might be proprietary to the organization. It is therefore important that adequate controls are in place to handle such information.

4.5.4 Relevance

The organizational unit members should believe that the assessment will result in some benefits that are relevant to their needs.

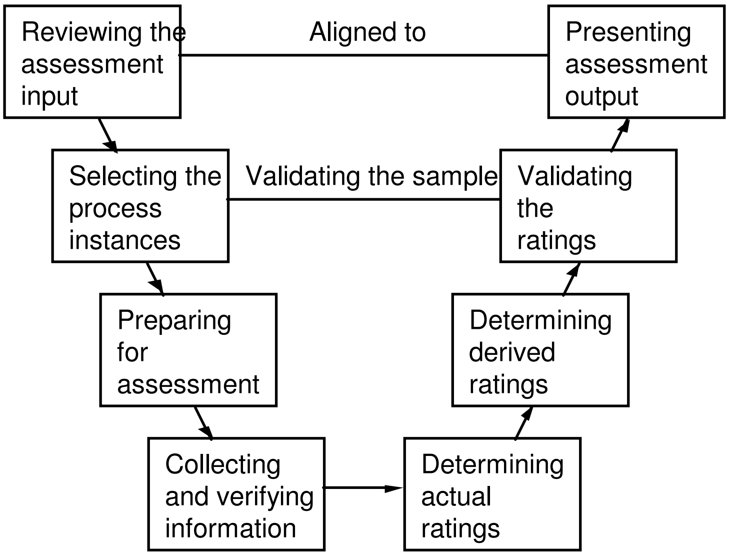
4.5.5 Credibility

The sponsor, and the management and staff of the organizational unit must all believe that the assessment will deliver a result which is objective and is representative of the assessment scope. It is important that all parties can be confident that the team selected to conduct the assessment has

- adequate experience in assessment;
- adequate understanding of the organizational unit and its business;
- sufficient impartiality.

5 Guidance on conducting assessments

The assessment consists of the eight stages shown in figure 2.


Reviewing the assessment input
Aligned to
Presenting assessment output
Selecting the process instances
Validating the sample
Validating the ratings
Preparing for assessment
Determining derived ratings
Collecting and verifying information
Determining actual ratings

Assessment stages

Figure 2 – Assessment stages

5.1 Reviewing the assessment inputs

The assessment input shall be defined prior to an assessment. At a minimum, the assessment input shall define:

- *the assessment purpose;*
- *the assessment scope;*
- *the assessment constraints;*
- *the identity of the qualified assessor and any other specific responsibilities for the assessment;*
- *the definition of any extended processes identified in the assessment scope;*
- *the identification of any additional information to be collected to support process improvement or process capability determination.*

[Software Process Assessment - Part 3: Rating processes, 4.2]

The qualified assessor named in the assessment input shall be a member of the assessment team in a team-based assessment or shall oversee an assessment conducted using a continuous or tool-based approach.

The qualified assessor shall ensure that the assessment is conducted in accordance with the requirements of this International Standard.

[Software Process Assessment - Part 3: Rating processes, 4.3]

The qualified assessor should review the defined assessment purpose, scope and constraints to ensure that they are consistent and that the assessment purpose can be fulfilled. The qualified assessor should seek clarification from the sponsor as appropriate.

5.2 Selecting the process instances

5.2.1 Mapping the organizational unit processes to the process model

> *The qualified assessor shall ensure that the organizational unit's processes to be assessed, as defined in the assessment scope, are mapped to the corresponding processes in part 2 of this International Standard or are defined as extended processes.*
> **[Software Process Assessment - Part 3: Rating processes, 4.3]**

In order to provide a consistent basis for assessment, part 2 of this International Standard establishes a process model that is representative of the software process as a whole. However, the processes within the model need not necessarily have a one-to-one mapping with the processes as performed by the organizational unit. The assessment scope details the organizational unit processes to be assessed and the mapping to the process model. If this mapping has not been performed as part of the assessment input then the qualified assessor will have to perform this mapping before it is possible to start to select specific process instances for assessment. This mapping should be agreed with the sponsor.

NOTE 1: The mapping performed will also have to be taken into account in order to fully comprehend the assessment output which is in the form of the process model, not necessarily processes as performed by the organizational unit.

5.2.2 Process Instance Selection

> *The qualified assessor shall ensure that the set of process instances selected for assessment is adequate to meet the assessment purpose and will provide outputs that are representative of the assessment scope.*
> **[Software Process Assessment - Part 3: Rating processes, 4.3]**
>
> *The assessment shall include at least one process instance of each process identified in the assessment scope.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.1]**

Process assessment determines the capability of processes as defined and implemented by the organizational unit. There may be a number of instances of the implementation of a process. The assessment purpose and scope will have been defined to support either process improvement or process capability determination (or both), and should give clear indication as to the expected depth and coverage that is required.

The process instances assessed within the organizational unit will affect the circumstances under which the derived ratings can represent current capability or predict future capability. The larger the sample size of process instances and the more recent the information, the more representative the assessment outputs.

NOTE 2: The assessment constraints may require particular process instances to be included or excluded.

5.3 Preparing for a team-based assessment

5.3.1 Selecting and preparing the assessment team

5.3.1.1 Choosing the assessment team size

An assessment team should be appointed. In order to assist in maintaining balanced judgement, it is recommended that the team should consist of at least two members. The optimum size of the assessment team, however, will depend on many factors including

- the assessment scope;
- the size of the organizational unit involved in the assessment;
- the skills and experience of the available resources;
- the cost/benefit trade-offs.

5.3.1.2 Defining assessment team roles

The qualified assessor should appoint the assessment team leader and assessment team co-ordinator. The assessment team leader will be responsible for the overall conduct of the assessment. The assessment team co-ordinator will be responsible for the assessment logistics and interfacing with the organizational unit. Depending on the needs of the assessment, the qualified assessor may fulfil one or both of the roles described above. An assessment team leader who is not a qualified assessor may need to seek advice and guidance on aspects of the assessment from the nominated qualified assessor, who has overall responsibility for ensuring that the assessment meets the requirements of this International Standard.

The assessment team may also have as many other members as appropriate. All assessment team members should have experience in software engineering and one or more should have specific experience in the processes under assessment and in the technologies used to support the processes.

In choosing the assessment team, the aim is to ensure a suitable level of objectiveness and to minimize the risk of misunderstandings. The following should be considered:

- assessment team members should be sensitive to people and able to collect information in a clear and non-threatening way;

  NOTE 3: If any of the assessment team are managers of one or more of the participants then this might be difficult to achieve.
- at least one assessment team member should be from the organizational unit;
- the composition of the assessment team should ensure a balanced set of skills to meet the assessment purpose and scope.

The assessment team members can be drawn from a variety of sources including

- organizational unit members;
- elsewhere within the organization;
- internal or external experts in specific processes;
- internal or external experts in process assessment;
- customers;
- sponsors.

5.3.1.3 Preparing the assessment team

Prior to the assessment, the qualified assessor should ensure that the assessment team has an understanding of

- the assessment inputs, purpose, constraints, output and process as described in this part of this International Standard;
- this International Standard, its requirements and relevant guidance.

5.3.2 Planning the assessment

5.3.2.1 Identifying risk factors

The assessment team should identify to the sponsor any significant risk factors that could lead to a failure of the assessment. Factors that should be considered include

- changes in the commitment of the sponsor;
- unplanned changes to the structure of the assessment team (e.g. the qualified assessor becomes unavailable);
- organization changes;
- implementation or changeover to new standard processes;
- changes in the assessment purpose or scope (e.g. because of some sudden change in the business factors driving the motivation for the assessment);
- resistance or unwillingness to participate by organizational unit members;
- lack of financial or other resources required for the assessment;
- lack of confidentiality, either internal or external to the organization.

5.3.2.2 Selecting the assessment techniques

The assessment team should select the appropriate assessment techniques to suit the assessment purpose and scope, the skills of the assessment team and the level of understanding of the organizational unit being assessed. These techniques may include

- on-line expert system;
- interviews;

- individual discussions;
- group discussions;
- closed team sessions;
- documentation reviews;
- feedback sessions.

Feedback sessions should be used as appropriate during the assessment. They can be particularly useful to present preliminary findings.

5.3.2.3 Selecting the assessment instrument

> *An assessment instrument that conforms to the requirements set out in part 5 of this International Standard shall be used to support the assessment.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.7]**

The assessment team should decide which assessment instrument will be used to assist the team in performing the assessment. The following aspects should be considered when defining the requirements for an appropriate assessment instrument

- the type of assessment instrument required;
- support for security and confidentiality;
- the level and detail of reporting;
- support for rating and analysis.

It is helpful if at least one assessment team member has experience of the particular assessment techniques and assessment instrument to be used.

For more guidance on selecting or developing an assessment instrument see part 5 of this International Standard.

5.3.2.4 Developing the assessment plan

The assessment team should develop an assessment plan detailing

- the assessment inputs;
- the role and responsibilities of all involved in the assessment activity;
- estimates for schedule, costs and resources;
- control mechanisms and checkpoints;
- the interface between the assessment team and the organizational unit;
- outputs expected;
- risk factors to be taken into account and appropriate contingent and preventive actions;
- the logistics which may include the rooms for discussions, presentations, appropriate audio-visual equipment, word processing facilities, escort requirements, access to facilities.

The assessment team should ensure that the assessment plan is able to meet the assessment purpose. The assessment plan should be formally accepted by the sponsor and owner. The assessment team should endeavour to ensure that the plan is acceptable to the participants and that it is realistic in terms of its impact on existing projects.

5.3.3 Preparing the organizational unit

5.3.3.1 Selecting the organizational unit co-ordinator

An organizational unit may appoint an organizational unit co-ordinator to represent it in the assessment.

The organizational unit co-ordinator is responsible for:

- supporting all assessment logistics for the organizational unit;
- interfacing with the assessment team;
- establishing the environment needed for the assessment activities.

5.3.3.2 Briefing of the Organizational Unit

The organizational unit should be briefed on

- the assessment purpose, scope and constraints;
- the conduct of the assessment;
- how the assessment outputs can be used to provide the most benefit to the organization;
- what arrangements exist for confidentiality and ownership of the assessment outputs.

The briefing should be performed in co-operation with the assessment team.

5.3.3.3 Selecting the Participants

The assessment team needs to capture information on every base practice and generic practice for each process instance to be assessed. The organizational unit should, in co-operation with the assessment team, select the participants that adequately represent the process instances chosen to ensure that the appropriate expertise will be available to allow for a satisfactory assessment.

5.3.4 Meeting confidentiality agreements

The assessment team should ensure that any discussions held with participants and the use of the assessment outputs are subject to any confidentiality agreement defined in the assessment constraints.

The assessment team should ensure that all participants fully understand the confidentiality agreement.

5.3.5 Gathering support documentation and records

The assessment team may need access to support documentation and records (e.g. project plan, progress meeting minutes, deliverable review notes) on the process instances to be examined, either in advance of the assessment or to provide support during the assessment. This may be particularly important for an independent assessment.

As part of the organizational unit briefings, the participants involved in the assessment will be aware of the processes to be assessed. Consequently, either they or the organizational unit co-ordinator should ensure that the support documentation and necessary records are made available for review.

Much of this material will be local to the organizational unit being assessed but some may be shared with other organizational units or be held centrally within the organization. If necessary the organizational unit co-ordinator should ensure that access to such material is considered in the assessment plan.

It helps the progress of the assessment if adequate time is provided for this information to be collated off-line from any ongoing discussions.

5.4 Collecting and verifying information

Although an assessment may be a self-assessment or an independent assessment, the principles behind the involvement of participants are the same; they are a primary source of information to be provided to the assessment team about the process instances being assessed. They may participate in informal, unstructured discussions that allow them to express their professional views about the processes in place, and any issues or problems facing the organization. They may also be involved in providing validation materials to the assessment team.

During an assessment, it is typical to perform a series of information collection and analysis stages, where the scope of the information is refined and more detailed information is collected along the way. For example, the first stage of information collection and analysis may help the assessment team determine which processes should be investigated more thoroughly. Subsequent stages should then collect and analyse more detailed information regarding how individual practices are being applied and how the process purpose is being achieved.

5.4.1 Collecting information

Collecting the information will rely on the assessment instrument and assessment techniques selected. Information has to be collected for each base practice and each generic practice for each process instance to be assessed.

Certain base practices or generic practices may be implemented more than once within a single process instance e.g. multiple reviews. In this case the assessment team must use the selected assessment instrument and their judgement to choose a representative sample.

If, during information gathering, it becomes obvious to the assessment team that a particular process has no implemented generic practices within a particular common feature or capability level, the assessment team may choose not to attempt to gather any additional information for that common feature or capability level. The absence of any capability can often be determined by probing for information either at the capability level generally, or more specifically regarding a particular common feature.

5.4.2 Categories of information

The categories of information that should be collected during an assessment include

- the degree of adequacy or existence of base practices;
- the adequacy of generic practices;
- the experiences of the participants where they observed problems associated with the current processes used, e.g. ideas for process improvement.

Typically, the assessment instrument is used to collect information for all the categories of information outlined above. Some or all of the detailed information collected will usually be recorded as part of the assessment record to be used to support process improvement or process capability determination.

The assessment team should take adequate steps to protect any information collected that may be covered by a confidentiality agreement.

5.4.3 Verifying information

Support documentation and records should be used as appropriate to verify the information collected during an assessment. The amount of support documentation and records examined depends upon the assessment team's knowledge of the organizational unit, the assessment purpose and the level of trust and confidentiality established for the assessment.

Due to lack of familiarity with the process being examined, the assessment team in an independent assessment may need to collect more support documentation and records to establish the basis for determining ratings than is needed in a self-assessment.

5.5 Determining the Actual Ratings for Process Instances

> *A base practice adequacy rating or a base practice existence rating (see 4.4.3.2) shall be determined and validated for every base practice within each selected process instance for each process and/or extended process identified within the assessment scope.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.2.1]**
>
> *A generic practice adequacy rating shall be determined and validated for every generic practice within each selected process instance of each process and/or each extended process identified within the assessment scope.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.2.2]**

These ratings, collected for every process instance assessed, are the actual ratings determined from the information collected about the process instance by the assessment team.

5.5.1 Determining the actual ratings for base practices

5.5.1.1 Base practice adequacy and existence

> *Base practice adequacy shall be rated using the base practice adequacy rating scale defined below.*
>
> ***N; Not adequate:*** *The base practice is either not implemented or does not to any degree contribute to satisfying the process purpose;*
> ***P; Partially adequate:*** *The implemented base practice does little to contribute to satisfying the process purpose;*
> ***L; Largely adequate:*** *The implemented base practice largely contributes to satisfying the process purpose;*
> ***F; Fully adequate:*** *The implemented base practice fully contributes to satisfying the process purpose.*
>
> **[Software Process Assessment - Part 3: Rating processes, 4.4.3.1]**

> *Base practice existence shall be rated using the base practice existence rating scale defined below:*
>
> ***N; Non-Existent:*** *The base practice is either not implemented or does not produce any identifiable work products;*
> ***Y; Existent:*** *The implemented base practice produces identifiable work products.*
>
> **[Software Process Assessment - Part 3: Rating processes, 4.4.3.2]**

Base practices may be rated using either the base practice adequacy rating scale or the base practice existence scale. The same scale should be used for all base practices for a given assessment. The base practice ratings are determined for each process instance assessed from the information collected.

The base practice ratings do not constitute a part of the process profile but rather are recorded as part of the assessment record. Their purpose is to provide a clear understanding of the extent to which the process is performed.

The work product indicators that are defined in the assessment instrument are provided to indicate which points to consider to help to make consistent rating judgements.

NOTE 4: There is no implied relationship between base practice adequacy and base practice existence ratings.

5.5.1.2 Base practice rating reference

> *A unique reference shall be generated for each base practice rating that includes the process category, the process within the process category, the base practice of the process, and a process instance reference.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.5.1]**

In the examples that follow, the rating reference is of the form:

PC.PR.BP[instance reference]

where:

- PC is a process category;
- PR is a process within that process category;
- BP is a base practice of the process;
- [instance reference] is either the number of process instances in the rating or a complete list of the process instance references.

For example, for a single instance of process PRO.5 (Manage Quality) which we have given a process instance reference of 'A', then if base practice adequacy ratings are determined for each base practice we might have:

PRO.5.1[A] = L
PRO.5.2[A] = F
PRO.5.3[A] = F
PRO.5.4[A] = P
PRO.5.5[A] = F
PRO.5.6[A] = P

5.5.2 Determining the actual ratings for generic practices

5.5.2.1 Generic practice adequacy

> *Generic practice adequacy shall be rated using the generic practice adequacy rating scale defined below.*
>
> ***N; Not adequate:*** *The generic practice is either not implemented or does not to any degree satisfy its purpose;*
> ***P; Partially adequate:*** *The implemented generic practice does little to satisfy its purpose;*
> ***L; Largely adequate:*** *The implemented generic practice largely satisfies its purpose;*
> ***F; Fully adequate:*** *The implemented generic practice fully satisfies its purpose.*
>
> **[Software Process Assessment - Part 3: Rating processes, 4.4.3.3]**

The generic practice adequacy ratings are determined for each process instance assessed from the information collected.

Generic practice 1.1.1 (*Perform the process*) is concerned with the performed process fulfilling its process purpose as described in the in Part 2 of this International Standard. The ratings for the individual base practices provide an indication of the extent to which the practices contribute to meeting the process purpose. The overall rating for generic practice 1.1.1, however, is a judgement of the extent to which the base practices, and possibly other practices that may be required for a given process context, perform together to achieve the overall process purpose.

It is therefore possible, even though the ratings for each individual base practice are fully adequate or existent, that the process is not satisfying its process purpose . This may be the result of an inability of the base practices to operate effectively as a whole, or that key base practices not included in the process model are required to achieve the process purpose in a particular context.

All of the other generic practices fulfil purposes that provide a process management infrastructure to support the performance of the base practices. The purpose of some of these generic practices is the process purpose of an associated process within the process model. For example, the purpose of generic practice 3.2.2 is the same as its associated process SUP.5 - "Perform Peer Reviews".

The process management indicators that are included in the assessment instrument (see part 5 of this International Standard) are provided to indicate points to consider in making consistent judgements for the generic practices.

5.5.2.2 Generic practice rating reference

> *A unique reference shall be generated for each generic practice rating that includes the process category, the process within that process category, the capability level, the common feature within that capability level, the generic practice within that common feature, and a process instance reference.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.5.2]**

In the examples that follow, the rating reference is of the form:

PC.PR[instance reference];CL.CF.GP

where:

- PC is a process category;
- PR is a process within that process category;
- [instance reference] is either the number of process instances in the rating or a complete list of the process instance references;
- CL is a capability level;
- CF is a common feature within that capability level;
- GP is a generic practice within that common feature.

For example, for a single instance of process PRO.5 ('Manage Quality') which we have given a process instance reference of 'A', then if a rating is determined for each generic practice at capability level 3, the well-defined level, we might have:

PRO.5[A];3.1.1 = L
PRO.5[A];3.1.2 = P
PRO.5[A];3.2.1 = L
PRO.5[A];3.2.2 = N
PRO.5[A];3.2.3 = P

5.5.2.3 Process capability level

> *An actual process capability level rating shall be determined for each process instance assessed by aggregating the generic practice adequacy ratings within each capability level.*
>
> *For each process instance, the actual process capability level ratings shall describe, for each capability level, the proportion of generic practices that were rated at each point on the generic practice adequacy scale in a clear and unambiguous way.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.2.3]**
>
> *Equal weighting shall be applied to each generic practice adequacy rating when aggregating or deriving ratings.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.4]**

This actual process capability level rating can be represented in many different ways; within this part of this International Standard the following vector representation will be used:

[% Fully, % Largely, % Partially, % Not Adequate]

Within process instance 'A' of Process PRO.5, a rating can be determined for each capability level. The process capability level rating is determined by aggregating all the generic practice ratings for the generic practices associated with that capability level.

From the above example it can be seen that none of the generic practices were fully adequate i.e. 0%, 2 of the 5 generic practices were largely adequate - 40%, 2 of the 5 generic practices were partially adequate - 40% and 1 of the 5 generic practices were not adequate - 20%. These generic practice adequacy ratings can be represented as a vector in the form:

PRO.5[A];3 = [0,40,40,20]

The process instance is represented by the five process capability level ratings determined for that process instance. For example, for the process instance 'A' of process PRO.5 we might have:

PRO.5[A];1 = [0,100,0,0]
PRO.5[A];2 = [50,25,17,8]
PRO.5[A];3 = [0,40,40,20]
PRO.5[A];4 = [0,0,33,67]
PRO.5[A];5 = [0,0,20,80]

It should be noted that capability level 1 is represented by the single generic practice 1.1.1. Hence the actual rating will always be 100% of one of fully, largely, partially or not adequate, and 0% for the other three.

5.6 Determining derived ratings

> *Equal weighting shall be applied to each generic practice adequacy rating when aggregating or deriving ratings.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.4]**

From the actual ratings, derived ratings may be determined which can help to gain further insight into the processes within the organizational unit as a whole. Derived ratings are based on sampling and are therefore subject to all the restrictions that apply to sampled ratings.

The assessment team should decide which of the following derived ratings are useful in helping to ensure that the assessment purpose can best be fulfilled.

Since any derived ratings are based on an aggregation of actual ratings for process instances, the assessment team has to ensure that traceability is provided from the derived ratings to the actual ratings.

5.6.1 Aggregation between process instances

The ratings for generic practices and process capability levels for two or more process instances may be aggregated to determine a derived rating for the process.

5.6.1.1 Generic practice adequacy

We may aggregate actual generic practice adequacy ratings between two or more process instances of a specific process to derive an aggregated rating for the generic practice.

For example, if generic practice adequacy is rated for generic practice 3.3.1 of process instances 'A', 'B' and 'C' of process PRO.5 we might have:

PRO.5[A];3.1.1 = L
PRO.5[B];3.1.1 = P
PRO.5[C];3.1.1 = L

We can aggregate these three generic practice adequacy ratings to calculate a derived generic practice adequacy rating for the generic practice across of process instances:

PRO.5[A,B,C];3.1.1 = [0,67,33,0]

From this aggregated rating we can see that

- the derived rating consists of three process instances (A, B and C);
- for the sample taken of generic practice 3.1.1, it was largely adequate 67% of the time and partially adequate 33% of the time.

Optionally, rather than listing the process instance references for each process instance, if the references are not required for future analysis purposes then the number of process instances may be substituted and would be represented as:

PRO.5;3.1.1 = L,P,L
or
PRO.5[3];3.1.1 = [0,67,33,0]

5.6.1.2 Process capability level rating

> *A set of derived process capability level ratings shall be determined for each process identified in the assessment scope by aggregating the actual process capability ratings of the process instances. These derived ratings shall be sufficiently representative of the process capability levels of each process assessed to satisfy the assessment purpose.*
>
> *For each process identified in the assessment scope, the derived process capability level ratings shall describe, for each capability level, the proportion of generic practices that were rated at each point on the generic practice adequacy scale in a clear and unambiguous way.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.2.3]**

By aggregating process capability level ratings between a sample of process instances of a specific process within an organizational unit, derived ratings for the capability levels are obtained for the process.

For example, if the 'Well-Defined' capability level (level 3) is rated for process instances 'A', 'B' and 'C' of process PRO.5 then we might have:

PRO.5[A];3 = [0,40,40,20]
PRO.5[B];3 = [0,20,40,40]
PRO.5[C];3 = [0,20,20,60]

We can aggregate these three process capability level ratings to calculate an aggregated process capability level rating:

PRO.5[A,B,C];3 = [0,27,33,40]
or
PRO.5[3];3 = [0,27,33,40]

From this aggregated rating we can see that

- the derived process capability level rating consists of three instances of process PRO.5 (A, B and C);
- within the sample taken of capability level 3 of process PRO.5, 27% of the generic practices were largely adequate, 33% of the generic practices were partially adequate and 40% of the generic practices were not adequate.

5.6.2 Aggregation across processes

We may aggregate any generic practice ratings between process instances across different processes.

For example, for process category PRO, the 'Project Process Category', and for generic practice 3.1.1 we may have:

PRO.1[A,B,C];3.1.1 = [0,33,33,33]
PRO.2[A,B,C];3.1.1 = [0,33,67,0]
PRO.3[A,B,C];3.1.1 = [33,33,33,0]
PRO.4[A,B,C];3.1.1 = [0,0,67,33]
PRO.5[A,B,C];3.1.1 = [0,67,33,0]
PRO.6[A,B,C];3.1.1 = [0,33,33,33]
PRO.7[A,B,C];3.1.1 = [0,67,33,0]
PRO.8[A,B,C];3.1.1 = [33,33,33,0]

We can aggregate these eight generic practice adequacy ratings to calculate an aggregated generic practice adequacy rating:

PRO[A,B,C];3.1.1 = [8,38,42,12]

From this rating for generic practice 3.1.1 of process instances 'A', 'B' and 'C' for all processes within process category PRO we can see that 8% of the generic practices were fully adequate, 38% of the generic practices were largely adequate, 42% of the generic practices were partially adequate and 12% of the generic practices were not adequate.

This mechanism may be used to infer ratings for the generic practices within a specific capability level of a group of processes where a derived rating from a subset of those processes suggests that the implementation of that capability level is identical across the entire group e.g. because there is an organization-wide measurement programme.

5.7 Validating the ratings

> *A base practice adequacy rating or a base practice existence rating (see 4.4.3.2) shall be determined and* validated *for every base practice within each selected process instance for each process and/or extended process identified within the assessment scope.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.2.1]**
>
> *A generic practice adequacy rating shall be determined and* validated *for every generic practice within each selected process instance of each process and/or each extended process identified within the assessment scope.*
> **[Software Process Assessment - Part 3: Rating processes, 4.4.2.2]**

The ratings should be validated to ensure that they are an accurate representation of the processes assessed. The validation should include assessing whether the sample size chosen is representative of the processes assessed and that it is capable of fulfilling the assessment purpose.

The following mechanisms are useful in supporting validation:

- comparing results to those from previous assessments for the same organizational unit;
- looking for consistencies between connected or related processes;
- looking for proportional ratings across the capability levels e.g. higher ratings for higher levels than for lower ones;
- taking an independent sample of ratings and comparing them to the assessment team ratings;
- feedback sessions of preliminary findings to the organizational unit.

5.8 Presenting the assessment output

5.8.1 Preparing the assessment output

Having determined the base practice, generic practice and process capability level ratings, the assessment outputs need to be prepared.

> *The qualified assessor shall ensure that all of the information required in the assessment output is recorded in a suitable format to fulfil the assessment purpose and that it meets the requirements of this International Standard.*
> **[Software Process Assessment - Part 3: Rating processes, 4.3]**
>
> **The process profile**
>
> *The ratings for the assessed process instances within the assessment scope shall be recorded as the process profile consisting of:*
> – *the actual generic practice ratings and process capability level ratings for each process instance;*
> – *derived generic practice ratings and process capability level ratings for each process within the scope of the assessment;*
> **[Software Process Assessment - Part 3: Rating processes, 4.5.1]**
>
> **The assessment record**
>
> *Any other information which is pertinent to the assessment and which may be helpful in understanding the output of the assessment shall be compiled and recorded as the assessment record. At a minimum, the assessment record shall contain:*
> – *the assessment input;*
> – *the assessment approach that was used;*
> – *the assessment instrument used;*
> – *the base practice ratings for each process instance assessed;*
> – *the date of the assessment;*
> – *the names of team who conducted the assessment;*
> – *any additional information collected during the assessment that was identified in the assessment input to support process improvement or process capability determination;*
> – *any assessment assumptions and limitations.*
> **[Software Process Assessment - Part 3: Rating processes, 4.5.2]**

The assessment output will normally be used as a basis for developing an agreed improvement plan or determining capability and associated risk as appropriate. The guidance on how to perform this is provided in Part 7 and Part 8 of this International Standard.

5.8.2 Reporting the assessment output

> *In some circumstances it may be desirable to compare the outputs of the assessment of two or more organizational units, or for the same organizational unit at different times. Comparisons of assessment outputs shall be valid only if their process contexts are similar.*
>
> **[Software Process Assessment - Part 3: Rating processes, 4.4.6]**

The presentation of the assessment output might simply be in the form of a simple presentation for an internal assessment or might be in the form of a detailed report for an independent assessment. In addition, other findings and proposed action plans may be prepared for presentation, depending upon the assessment purpose and whether this additional analysis is performed at the same time as the assessment.

The ratings determined for generic practice adequacy provide the input for generation of the process capability level ratings. The use of summary ratings, for example for a process category, in addition to the detailed findings may be used to help to understand the findings.

In some circumstances it may be useful to assign a weighting to the four points on the generic practice adequacy scale e.g. 100% for F, 75% for L, 25% for P and 0% for N. Any derived rating may then be represented as a single value rather than as a vector. This may assist with presentation of ratings at a summary level. These ratings should be used for summary purposes only and should not be used for comparison instead of the generic practice adequacy ratings.

The presentation of the ratings may be in absolute terms (numbers, absolute scales) or relative terms (since last time, compared to benchmarks, compared to contract requirements, compared to business needs).

Consolidated product

Software Process Assessment – Part 5 : Construction, selection and use of assessment instruments and tools

Version 1.00

(Formerly AI 1.00)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

– assist the standardisation project in its preparatory stage by developing initial working drafts;

– undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;

– create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

These terms and conditions apply to the set of documents developed by the SPICE Project, and published within the Project as Version 1.0, with the following titles:

– Part 1 : *Concepts and introductory guide*
– Part 2 : *A model for process management*
– Part 3 : *Rating processes*
– Part 4 : *Guide to conducting assessment*
– Part 5 : *Construction, selection and use of assessment instruments and tools*
– Part 6 : *Qualification and training of assessors*
– Part 7 : *Guide for use in process improvement*
– Part 8 : *Guide for use in determining supplier process capability*
– Part 9 : *Vocabulary*

1. You may copy and distribute verbatim copies of any or all of the Documents as you receive them, in any medium, provided that you conspicuously and appropriately publish with each copy a copy of these Terms and Conditions. You may charge a fee for the physical act of transferring a copy.
2. You may copy extracts from these documents in materials for internal or public use, providing you provide clear acknowledgment of the source of the material, by citation or other appropriate means.
3. You may not copy, modify, sub-license, or distribute the Documents except as expressly provided under these Terms and Conditions.

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout

Members: Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : ***Concepts and introductory guide***

Product Manager: Terry Rout

– Part 2 : ***A model for process management***

Product Managers: Al Graydon, Mark Paulk

– Part 3 : ***Rating processes***

Product Manager: Harry Barker

– Part 4 : ***Guide to conducting assessment***

Product Manager: Harry Barker

– Part 5 : ***Construction, selection and use of assessment instruments and tools***

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : ***Qualification and training of assessors***

Product Manager: Ron Meegoda

– Part 7 : ***Guide for use in process improvement***

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : ***Guide for use in determining supplier process capability***

Product Manager: John Hamilton

– Part 9 : ***Vocabulary***

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards)

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

The document was composed in Helvetica 10 pt for A4 paper. There may be minor variations in the pagination of some of the big tables for font Arial 10 pt and/or use of US sized paper.

SOFTWARE PROCESS ASSESSMENT Part 5

Construction, selection and use of assessment instruments and tools

Contents

Annexes

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialing to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being co-ordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

In this part of the standard (Part 5) Annexes A, B, C and D are normative. Annexes E to H are informative.

Introduction

This document establishes the requirements for constructing an assessment instrument. In addition, it provides guidance on selection and usability characteristics associated with various types of assessment instrument. The other components of the standard, their relationships and interdependencies, are described in part 1 of this International Standard.

When an assessment is performed, the organization's implemented processes are compared with the process model defined in part 2 of this International Standard. Typically, during an assessment it is not realistic to build a complete process model of the entire organization. Hence, to determine whether a process has been sufficiently implemented, the assessor probes for evidence of the actual capability of the process. Information is collected about a representative sample of process attributes that is evaluated against the expected attributes. Based upon a review of this information, an assessor makes a judgement about the process capability of the organizational unit.

An assessment instrument (AI) is a tool (or set of tools) used throughout an assessment to support the evaluation of the adequacy or existence of practices. An assessment instrument aids the assessor by providing a consistent set of indicators as discriminators to help judge how well the practices have been implemented in the organizational unit's processes. An assessment instrument provides a mechanism to record the collected information from an assessment. Storage and retrieval capabilities provide the ability to maintain the results and supporting information for post-assessment analysis and improvement. Sophisticated assessment instruments may help the assessor to process the data and generate the results, thereby improving the efficiency and effectiveness of the assessment.

This part of the International Standard describes a framework for an assessment instrument. An important aspect of the framework is a set of assessment indicators that are the source input data to an assessment instruments. Other elements of the framework incorporate the ability to capture and process assessment data to produce repeatable results. Different types of assessment instrument support specific assessment techniques, objectives or modes of use. This document does not prescribe a particular format for an assessment instrument (e.g., questionnaire, checklist, computer input screen): the requirements for an assessment instrument are independent of a particular design, instrument style or mode of use. Assessment tool designers and methodology providers should evaluate the intended approach to gathering data and build an assessment instrument that supports the assessment approach.

It is important that each assessor performs the assessment in a consistent and repeatable manner to ensure the validity, usability and comparability of the assessment results. The common set of assessment criteria available in an assessment instrument through the indicators help to provide consistent and repeatable assessment results.

1. Scope

This part of the International Standard defines the required elements of an assessment instrument to support an assessment conducted according to this International Standard. It also provides guidance on the construction or selection of different types of assessment instruments. This document:

- sets out the minimum requirements to be met in the construction of an assessment instrument;
- defines a set of indicators to be included in an assessment instrument;
- provides guidance on the selection, construction and usability of assessment instruments.

Different types of assessment instrument support specific assessment techniques, objectives or modes of use. This document does not prescribe a particular format for an assessment instrument (e.g., questionnaire, checklist, computer -based tool); the requirements for an assessment instrument are independent of a particular design, instrument style or mode of use.

The set of indicators included in this part of the International Standard is not intended to be an all inclusive set, but rather provide the key characteristics of an instantiation of an assessed process that may be useful to judge adequacy. Requirements for tailoring the standard set of indicators are provided in Clause 4.

This part of the International Standard is directed to:

- those responsible for the design and construction of assessment instruments, e.g. methodology providers, tool suppliers, assessors;
- assessors and assessment teams with responsibility for the selection and procurement of appropriate assessment instruments;
- assessors, sponsors or other parties responsible for assessing conformance of an assessment instrument to these requirements.

2. Normative References

There are no normative references in this part of the International Standard.

3. Definitions

For the purposes of this part of this International Standard, the definitions in *Software Process Assessment - Part 9 : Vocabulary* apply.

4. Construction of an assessment instrument

4.1 Form and purpose of an assessment instrument

For the purposes of this International Standard, an assessment instrument is a tool or set of tools that is used throughout an assessment to support the evaluation of the existence or adequacy of practices within the scope of the assessment. It may provide assistance in collecting, recording, formalizing, processing, using, storing or retrieving information gathered during an assessment.

This International Standard does not require an assessment instrument to take any particular form or format. It may be constructed to be, for example, a paper-based instrument containing elements such as forms, questionnaires or checklists, or it may take the form of a computer-based instrument such as a spreadsheet, a data base system, an expert system or an integrated CASE tool.

4.2 Implementation of standard indicators

Regardless of the form of the assessment instrument, its main objective is to help an assessor to perform an assessment in a consistent and repeatable manner, reducing assessor subjectivity and ensuring the validity, usability and comparability of the assessment results. As a primary means of achieving this objective, an assessment instrument shall incorporate the standard set of assessment indicators defined in Annexes A, B, C and D, as appropriate to the scope and context of the assessment.

All indicators incorporated into an assessment instrument shall be traceable to a corresponding process, generic practice, or base practice in the process model in part 2 of this International Standard, or to a practice in an extended process.

4.3 Tailoring of indicators contained in an assessment instrument

4.3.1 General

Within an assessment instrument, the standard set of indicators and the form of the instrument may be tailored to meet the needs of the assessment team or sponsor in the following aspects:

- the modification of indicator format to accommodate presentation style preferences (i.e., questions, sentences, tables, on-line input screens, etc.);
- the modification of indicator wording to accommodate synonym names or meaning for cultural differences;

- the addition of scoping characteristics to help select the set of indicators used by process area, user, job function, application domain, software product, or other pre-defined organizational unit or tool characteristics;
- the addition of new indicators to support new work products, new technology and specific extended processes;
- the adaptation of the assessment instrument to accommodate extended processes, limited scope modularity, or intended distribution of tools to collect the assessment data incrementally;
- the user interface (i.e., format for data input, method of recording data, etc.);
- the format of the results (presentation format and output record format, etc.);
- the overall design and format of the assessment instrument;

Tailoring the indicators shall not impair the availability of the standard set of indicators appropriate to the scope and context of the assessment.

All practices within the assessment scope shall be covered by the tailored indicators.

4.3.2 Tailoring indicators for extended processes

This International Standard allows for the creation of extended processes containing additional practices to supplement those in the process model in part 2. When extended processes are defined, the following shall apply:

- corresponding indicators shall be defined and included in the assessment instrument for each additional practice in the extended process;
- a reference shall be recorded in the assessment record identifying the indictors related to the practices in extended processes;
- indicators for the practices in extended processes shall be maintained and made available to the sponsor or the assessed organization on request.

4.4 Modular assessment instruments

A modular assessment instrument is an instrument constructed or tailored from a collection of components, each of which provides only partial coverage of the full scope of the process model. A modular assessment instrument, at a minimum, shall incorporate all standard indicators related to the processes to be assessed and all of the process management indicators.

Assessors using a modular assessment instrument shall record any limitation of the coverage of the instruments used in the assessment record.

The use of a modular assessment instrument shall not negate the rules for coverage of the practices contained within this International Standard.

A supplier of a modular assessment instrument should clearly identify the applicability of the instrument and the extent of its coverage of the process and practices of the process model in part 2 or of extended processes.

4.5. Capturing and processing assessment data

An assessment instrument shall have the ability to capture the data required to be used in the production of ratings as defined in part 3 of this International Standard.

NOTE 1: In paper-based instrument, for example, this could be met simply by providing a place to write the results.

An assessment instrument shall have the ability to capture and maintain supporting information as required by the assessment sponsor and defined in the assessment input.

An assessment instrument shall support the rating of the practices being assessed, including those contained in extended processes, according to the rating scheme defined part 3 of this International Standard.

When an extended process is included in the scope of an assessment, the assessment instrument should enable the assessor to segregate the rating of the base practices contained in the process model from the additional base practices in an extended process.

An assessment instrument should provide a mechanism to aid the segregation of data and results between the assessment output as defined in part 3 of this International Standard.

An assessment instruments should, whenever possible, provide automated support to the assessor for the processing and aggregation of results across multiple organizational units or process instances.

4.6 Using an assessment instrument

The assessment instrument should be appropriate to the scope and purpose of assessment.

Assessors should record the existence, absence, or non-applicability of the indicators used in the assessment.

The assessment instrument records of the existence, absence or non-applicability of the indicators should be provided to the assessed organizational unit upon request to allow the use of the information in subsequent process improvement planning.

Assessment instrument records should be maintained by the assessor's organization as a record of the assessment.

An assessment instrument should be capable of loading, storing and comparing process profiles.

Assessors should use all the data captured in an assessment instrument about indicators, the context of the assessment, and the organizational unit characteristics to support their judgements of practice adequacy or existence.

Annex A (normative)

Process management indicators

Introduction

Process Management Indicators provide guidance to the assessor on what to probe for in the organization to determine whether the generic practices, defined in part 2 of this International Standard, have be adequately implemented. Generic practices are applicable to every process. They provide the assessor with a view of the organization's ability to manage its processes. The Process Management Indicators should be used in conjunction with the practices in part 2 of this International Standard (which are not duplicated in this document) and the Work Products Characteristics contained in Annex D. The practices, together with the work product characteristics form the set of Process Indicators. These indicators help the assessor judge the adequacy of the generic practice 1.1.1 "Perform the process".

The information in Annex B and Annex C provide a way to map the appropriate process and practices to the information contained in Annex D. The information in all these tables provides guidance to the assessor in how to judge "practice adequacy". This information may be tailored by the assessment instrument tool designer or assessor according to the rules defined in clause 4 of this part of the standard.

The Process Management Indicators table contains the following fields:

Associated Processes / Practices: Defines other processes and/or practices which may be used to support the assessment of this practice

Potential Sources for Existence Evidence: Defines potential artefacts where an assessor might look to find evidence that this practice was implemented in the organization. This field lists equivalent artefacts which could be used to demonstrate that the practice was implemented. The list is not inclusive.

Process Management Indicators: The phrases or key words that provide guidance to the assessor in what to probe for during an assessment in making a judgement of "adequacy". An assessor probes for all of the things listed. Then their judgement is based on the information gathered supported by the context information defined for the assessment.

Process Management Indicator Table

Level 1 Performed-Informally

Common Feature 1.1: Base Practices are Performed

Practice:	**1.1.1**
Perform the process.	**Perform the base practices in implementing the process to provide work products and/or services to a customer.**
Associated Processes/Practices:	This practice applies to each process within the scope of the assessment. Note: To help evaluate this generic practice use the Process Indicators defined in Annexes B - D and the base practices.
Potential Sources for Existence Evidence	– Any input and output work product associated with the process – Discussions with process representatives
Process Indicators	The process is performed in the organization. – Evaluate each base practice for each instance of the process being reviewed. – each base practice is represented in the process defined in the organization. – the process representatives can demonstrate that the base practices for process are used (even though the process may not be documented). – In each organizational unit assessed, evidence exists that each base practice is actually performed. – samples of the input and output work products similar to those specified in the process in part 2 of this International Standard exist and have the characteristics to indicate an adequate implementation (see Annex D). – a mechanism exists to distribute the work products associated with the process.

Note 1: At Level 1 a process may not be documented, just performed informally. An organization that has the capability to perform at a higher level of maturity will have a more well defined process. To assess this, the assessor should use the Process Indicators along with the Process Management Indicators. The attributes observed in the process will help define the organization's level of capability.

Note 2: If using an *automated* self-assessment tool the assessor may just indicate in the tool that the practices for process are performed, and record the information about work products characteristics, and practice adequacy scores, capturing any evidence that might substantiate the self-assessment results if required at some later point.

Process Management Indicator Table

Level 2 Planned-and-Tracked

Common Feature 2.1: Planning Performance

Practice:	**2.1.1**
Allocate resources.	**Allocate adequate resources (including people) for performing the process.**
Associated Processes/Practices:	**PRO.7.1** - Acquire Resources.
Potential Sources for Existence Evidence	– Current project specification or plan (17) – Current progress status (20, 89) – Process performance data (18) – Estimates (tools/records) (11) – Historical records of similar projects (18, 20) – Process measures (38) – Schedule (5) – Commitments/agreements (50) – Process representative – Tour of facilities – Overtime records
Process Management Indicators	– Evidence of resource allocation exists. Resource(s) may include: – funding – staff – equipment – workspace – tools – Process representative(s) indicate that resources are sufficient to perform tasks assigned – Records/plan indicate resources are allocated to perform job tasks – Project tracking shows resource utilization consistent with current project plan – Where historical records exist: – allocation of resources are consistent with the historical records of projects with similar scope. – resources estimates are based on historical data (when it exists).

Process Management Indicator Table

Level 2	**Planned-and-Tracked**

Practice:	**2.1.2**
Assign responsibilities.	**Assign responsibilities for developing the work products and/or providing the services of the process**
Associated Processes/Practices:	PRO.3 - Build Project Teams
Potential Sources for Existence Evidence	– Process description (3) – Job procedures/practices (4) – Project plans (17) – Work breakdown structure (6) – Training records (89) – Review strategy/plan (30) – Process representative
Process Management Indicator	– Job responsibilities correspond to tasks attributes defined in the practices. – Representative understands the process and tasks they are responsible for. – Staff assigned either have the skills or are planned to have the skills needed to perform the task. – Assigned responsibilities are recorded

Process Management Indicator Table

Level 2 Planned-and-Tracked

Practice:	**2.1.3**
Document the process.	**Document the approach to performing the process in standards and/or procedures.**
Associated Processes/Practices:	ORG.2 Define the Process. SUP.1 Develop Documentation PRO.1.3 Describe Activities and Tasks PRO.2.2 Identify Project Standards
Potential Sources for Existence Evidence	– Process descriptions (3) – Standards (9) – Coding standards (10) – Job procedures or practices (4) – Quality strategy (25) – Review strategy (30) – Customer support procedures (82) – Installation guide (75) – Configuration management plan (91) – Project management tools
Process Management Indicators	– Process approach and tasks to be performed for the organizational unit are documented. – Procedures documented are the procedures used by the organizational unit. – Process documented contains elements defined in Org.2, such as: – tasks to be performed – inputs and outputs – entry / exit criteria – control points – internal and external interfaces – process measurements – In defining tasks consideration is given to: – sequencing of tasks – task dependencies – good practices to follow in performing the tasks – Standard and procedure documents for the organizational unit are developed consistent with SUP.1: – requirements are identified – documents are checked – documents are maintained – Process owners/users have input to defining the documented process / procedures.

Process Management Indicator Table

Level 2	**Planned-and-Tracked**

Practice:	**2.1.4**
Provide tools	**Provide appropriate tools to support performance of the process.**
Associated Processes/Practices:	ORG.6 Provide Software Engineering Environment. ORG.7 Provide Work Facilities
Potential Sources for Existence Evidence	– Development environment (104) (see the associated process indicators)
Process Management Indicators	– Tools are used support the process activities defined in the organizational unit. – Practitioners verify that the tools in use meet their needs. – Tools defined are available to those who perform the task(s). – Adequate number of tools are available to support the activities defined – Tools used add value to the required tasks – Personnel who use the tools receive adequate training in the operation of the tool – Documentation and/or instructions is available for the tool – Support for the tool is available

Process Management Indicator Table

Level 2	**Planned-and-Tracked**

Practice:	**2.1.5**
Ensure training.	**Ensure that the individuals performing the process are appropriately trained in how to perform the process.**
Associated Processes/Practices:	ORG.4 - Perform Training * This is applied to each process in the scope of the assessment.
Potential Sources for Existence Evidence	– Training strategy/plan (88) – Mentoring plan – Training records (89) – Training materials (90) – Formal course materials – Written project materials – On-line job aids – Video library – Training curricula – Project plan (17) – Personnel records (109) – Estimates (11)
Process Management Indicators	– Training needs for the staff performing the tasks are identified – Practitioners verified that training was sufficient and adequate to perform the tasks assigned: – training is available for tools used in the process tasks performed – training curricula covers tasks in the defined process. – The organizational unit allocated resources for training: – resources cover training costs – time is allocated in project plan for staff training when required – training materials for the process exist – personnel records indicate staff had sufficient training in the process tasks assigned. – Training could take the form of: – internal training (training classes, self-instruction tools) – external training (ex. degrees, courses taken outside company, certification) – prior expertise / experience – mentoring related to process tasks they are assigned to

Process Management Indicator Table

Level 2	**Planned-and-Tracked**

Practice:	**2.1.6**
Plan the process.	**Plan the performance of the process.**
Associated Processes/Practices:	PRO.2 Establish Project Plan. PRO.5 Manage Quality PRO.6 Manage Risks
Potential Sources for Existence Evidence	– Project plan(s) (17) – Work breakdown structure (6) – Review strategy/plan (30) – Reuse strategy (33) – Risk management strategy/plan (23) – Risk analysis record/report (22) – Quality strategy (25) – Estimates (tools/records) (11) – Quality records (28) – Measures (36) – Project management tools – CASE tools
Process Management Indicators	– The **plan** contains the key elements: – **Work breakdown structure** is defined including at least the tasks specified in part 2 if this International Standard for the process being reviewed. – **Project standards** to be used are identified and available. – **Special needs** (facilities, tools, personnel) are identified, along with resources to obtain the special need. – **Reuse strategy** is defined and identifies the: – key elements to be reused – the objectives for reuse within the project – the mechanism used to implement reuse – **Project resource estimations** are: – based on historical information when available – consistent with resources available to the project – based on project measurements (see 2.4.1) – Project measures used in estimation are identified – **Project risks** are identified and reflect the: – resources utilization – availability of resources – schedule constraints – cost constraints – technical risks – **Schedule** is defined which – reflects the constraints considered (resources, time, personnel skills) – contains appropriate contingency time – meets customers needs and objectives

Process Management Indicator Table

Level 2	**Planned-and-Tracked**
Process Management Indictors (continued)	– The **plan** defined is: – complete – accurate – easy to understand – realistic – available to those performing the task – consistent with schedule needs and project objectives – and – contains project commitments – covers the strategy (approach / methodology / life cycle)

Process Management Indicator Table

Level 2	**Planned-and-Tracked**

Common Feature 2.2:	**Disciplined Performance**

Practice:	**2.2.1**
Use plans, standards, and procedures.	**Use documented plans, standards, and/or procedures in implementing the process.**
Associated Processes/Practices:	ORG.2.13 Deploy the Process PRO.5 Manage Quality PRO.6 Manage Risks PRO.7 Manage Resources and Schedules PRO.8 Manage Sub-Contractors
Potential Sources for Existence Evidence	– Project plans (16,17) – Quality plan (25) – Process performance data (18) – Process status record (20) – Meeting minutes (19) – Estimates (tools/records) (11) – Quality records (28) – Risk analysis record/report (22) – Assessment audit record (29) – Review record (31) – Measures (36-42) – Training records (89) – Process representative
Process Management Indicators	– Organizational unit representative understands the documented process / standards and / or procedures – Evidence exists that the documented process is used: – input/output work products exist – tasks to be performed have been assigned – Process defined is achievable with the project constraints – Evidence exists that the plans defined are used by the organization: – plan milestones are achieved or replanning is performed – schedule is consistent with the plan defined or replanning performed – resources used are in line with those specified in the plan or replanning performed – potential risks identified in the plan are tracked

Process Management Indicator Table

Level 2 Planned-and-Tracked

Practice:	**2.2.2**
Do Configuration Management.	**Place work products of the process under version control or configuration management, as appropriate.**
Associated Processes/Practices:	SUP.2 Perform Configuration Management
Potential Sources for Existence Evidence	– Listing of work products associated with the process – Configuration management plan (91) – Configuration management (file, library, system)(92) – Change requests (94) – Change control records (95) – Change history (96) – Progress status record / report (20) – Build list(57) – Release package (70)
Process Management Indicators	– All appropriate work product are maintained under configuration management – A storage mechanism for configured items exists, such as: – paper document library – project files or binders – on-line configuration management system library – The configuration management (CM) mechanism: – has archival / retrieval capabilities – has an index of items under CM – has controlled access procedures – indicates the status of items under CM – has a version indicator scheme – Work Products identified for the process have version indicators identified, for example: – on-line documents have ability to generate version information. – printed documents have version identifiers on them – Baselined copies of the work product for the process correspond to the project's current development status. – current status of the work product can be readily ascertained – Work products are accessible to organizational unit personnel with a "need to know":

Process Management Indicator Table

Level 2	**Planned-and-Tracked**

Process Management Indicators	– Change control is established for items baselined under CM: – change control procedure requires approval for change to baselined products – a mechanism to track changes made is established – process revisions are proactively made available to those who need them – a mechanism to inform project personnel of changes made to baselined documents exist

Process Management Indicator Table

Level 2 Planned-and-Tracked

Common Feature 2.3: Verifying Performance

Practice:	**2.3.1**
Verify process compliance.	**Verify compliance of the process with applicable standards and/or procedures**
Associated Processes/Practices:	SUP.3 Perform Quality Assurance. CUS.4.3 Conduct joint management reviews.
Potential Sources for Existence Evidence	– Standards (9) – Coding standards (10) – Product needs assessment(44) – Review records(31) – Assessment / audit record (29) – Progress status record / report(20) – Meeting minutes(19) – Corrective actions (97) – Process quality records (28) – Process measures (39)
Process Management Indicators	– Reviews, self-assessments and / or audits of the process are performed on a regular scheduled basis – Review, self-assessment and / or audit results exist: – are documented and/or maintained – indicate verification of appropriate standards and/or procedures – identify adherence to appropriate standards and or procedures – indicate corrective action plans for non- conformance to standards and or procedures – are used in process improvement planning – Milestones and/or quality criteria defined for the process tasks include verification of the usage of the appropriate standard and procedures

Process Management Indicator Table

Level 2 Planned-and-Tracked

Practice:	**2.3.2**
Audit work products.	**Verify compliance of Work Products with the applicable standards and/or requirements.**
Associated Processes/Practices:	CUS.3 Identify Customer Needs PRO.4. Manage Requirements PRO.5 Manage Quality SUP.3 Audit work products* (duplicate requirement).
Potential Sources for Existence Evidence	– Customer request (83) – Customer requirements(52) – Standards (9) – Review records (31) – Assessment audit records (29) – Meeting minutes (19) – Progress status records (20) – Corrective actions (97) – Work product quality records (28) – Quality measures (29) – Requirements specification.(52) – Customer contracts (51) – Process input and output work products
Process Management **Indicators**	– Work product are reviewed: – review criteria for work products include the verification of standards and/or requirements – review records indicate usage of project standards and requirements – Quality criteria for work product completion verifies usage of the standards and/or procedures. – Requirement traceability is established for the work products: – work product(s) are traceable to associated requirements or standards – when CASE tools are used to store work product outputs, they have requirements traceability capabilities

Process Management Indicator Table

Level 2 Planned-and-Tracked

Common Feature 2.4: Tracking Performance

Practice:	**2.4.1**
Track with measurement.	**Track the status of the process against the plan using measurement.**
Associated Processes/Practices:	PRO.2.7 Identify project measures PRO.3.4 Manage inter-team issues PRO.7 Manage Resources and Schedule ORG.2.9 Define Process Measures CUS.4.3 Conduct joint management reviews.
Potential Sources for Existence Evidence	– Progress status(20) – Review records (31) – Assessment / audit record (29) – Corrective actions (97) – Process description (3) – Measures (36-42) – Project measures (37) – Tracking system (98) – Schedule (5) – Meeting minutes(19) – Project plan (17) – Project management tools
Process Management Indicators	– Measurements to track status of the process identify: – key process attributes to be tracked – status of deliverables – quality of deliverables – Project measures cover key elements of the project plan: – process / critical task status – project performance against plan – resource utilization against plan – time schedule against plan – process quality measures – product quality measures – Process owners and customers participate in defining the measures / goals. – Milestones / quality objectives for each process are established.

Process Management Indicator Table

Level 2 Planned-and-Tracked

Process Management Indicators (Continued)	– The measures used are indicative of the process' performance – progress deviations are identified. – measures show planned vs. actual. – process defects are identified . – process quality vs. objectives/criteria. – The measures defined are: – usable – understood by those expected to utilize them – provide value to users in the organization – non-interruptive to the work flow – The reporting interval is appropriate for the life cycle model used. – Measurement reports are available to those with a need to know – managers – process owners – interface groups – quality representatives – customers

Process Management Indicator Table

Level 2 Planned-and-Tracked

Practice:	**2.4.2**
Take corrective action.	**Take corrective action as appropriate when progress varies significantly from that planned.**
Associated Processes/Practices:	PRO.5.6 Take corrective action PRO.6.8 Take corrective action PRO.7.2 Track progress ORG.3.7 Change the process
Potential Sources for Existence Evidence	– Corrective actions (97) – Meeting minutes (19) – Project measures (37) – Process measures (38) – Progress status(20) – Schedules (5) – Project plans (17) – Trouble reporting system
Process Management Indicators	– A mechanism is defined to facilitate monitoring of the process, project, and product to identify when corrective actions are required – When projects performance is deviating from planned activities or performance goals: – problems area are identified which have associated corrective actions – established plans and schedules are adjusted – notification is given to dependent task owners and customers – New requirements to a project result in a project analysis and potential replanning activities when required

Process Management Indicator Table

Level 3   Well-Defined

Common Feature 3.1:   Defining a Standard Process

Practice:	**3.1.1**
Standardize the process.	**Document a standard process or family of processes for the organization, which describes how to implement the base practices for the process.**
Associated Processes/Practices:	ORG.2 Define the Process SUP.1 Develop Documentation PRO.1.3 Describe activities and tasks **ORG.2.10* *Document the standard process (duplicate base practice)*
Potential Sources for Existence Evidence	– Process description(3) – Job practices, procedure(4) – Work breakdown structure (6) – Standards (9) – Software development methodology (1) – Quality criteria (27)

Process Management Indicator Table

Level 3 Well-Defined

Process Management Indicators

- The organization's standard process documentation exists and includes:
  - expected input and output work products
  - work break down structure:
    - tasks to be performed
    - task ownership
    - objective criteria for demonstrate the task completeness
    - objective criteria to demonstrate the sufficiency of input and output work products
  - definition of internal and external interfaces
  - quality controls:
    - process entry and exit criteria
    - process decision control points
    - process measures
    - process performance characteristics and expectations
  - performance characteristics for the standard or tailored process:
    - productivity expectations
    - quality expectations
    - process adherence objectives
    - estimated development resources:
      - time
      - cost
      - personnel
- The standard process is documented and provides coverage for the associated base practices
- When defining measures consideration is given to ensure:
  - usability of measures
  - applicability of measures to the project
  - availability of measures to those with a "need to know"
  - completeness of source data used to generate the results
  - validation of the accuracy of the source data
- The standard process documented reflects the current practices performed throughout the organization
  - documentation is validated (reviewed, tested)
  - documentation is approved
- Standard process is available to all with a *need-to-know* in the organization.
  - paper documentation distributed to key process users
  - on-line documentation is accessible to key process users

Process Management Indicator Table

Level 3 Well-Defined

Practice:	**3.1.2**
Tailor the standard process.	**Tailor the organization's standard process family to create a defined process which addresses the particular needs of a specific use.**
Associated Processes/Practices:	PRO.1 Plan Project Life Cycle, CUS.3 Identify Customer Needs, ORG.2 Define the Process, SUP.1 Develop documentation
Potential Sources for Existence Evidence	– Process description(3) – Job practices, procedure(4) – Work breakdown structure (6) – Standards (9) – Software development methodology (1) – Quality criteria (27) – Process management tools – Configuration management library / system(92)
Process Management Indicators	– Guidelines on how to tailor the standard process exist. – Tailoring guideline contain: – criteria on what may be tailored – approval process for tailoring – usage criteria for the tailored process – The tailored process includes: – tasks to be performed (e.g. work break down structure) – objective criteria for demonstration of task completeness – objective criteria for demonstration of inputs and output sufficiency for the next dependent task – Standard process documentation is adapted to include the tailored process. – Process documentation related to the tailored process is available to those who need it. – The tailored process is understood by organizational representatives using it. – Training is adapted to the tailored process and available to those who need it

Process Management Indicator Table

Level 3	Well-Defined

Common Feature 3.2:	Perform the Defined Process

Practice:	**3.2.1**
Use a well-defined process.	**Use a well-defined process in implementing the process.**
Associated Processes/Practices:	ORG.2.13 Deploy the Process PRO.5 Manage Quality PRO.6 Manage Risks PRO.7 Manage Resources and Schedules PRO.8 Manage Sub-Contractors
Potential Sources for Existence Evidence	– Process description (3) – Job procedures, practices (4) – Work breakdown structure (6) – Work products (7) – Project plans (16,17) – Process performance data (18) – Process status record (20) – Meeting minutes (19) – Estimates (tools/records) (11) – Quality records (28) – Risk analysis record/report (22) – Assessment audit record (29) – Measures (36-42) – Training records (89)
Process Management Indicators	– The organization implements a standard process throughout the organization in a consistent way – all projects use the same standard process or a tailored version of it – Organizational representatives: – understand the standard / tailored process – are trained in the standard / tailored process – verify the performance of defined tasks – Quality criteria for proceeding from task to task exists – entry criteria are met prior to the start of the process – exit criteria are met prior to the completion of the process – deviations from entry / exit criteria have documented, approved corrective actions defined. – entry / exit criteria demonstrate the sufficiency of input and output work products to perform the scheduled tasks – all defined exit criteria and / or corrective actions are tracked until completed.

Process Management Indicator Table

Level 3	Well-Defined

Process Management Indicators (Continued)	– The organization demonstrates the performance of all tasks defined in the standard or tailored process – task completion verification mechanisms exists – deviations from the defined process are documented, and officially approved – quality criteria are evaluated at key milestones in the defined process. – Input/output work products are monitored for: – adherence to defined standards and requirements – accuracy – sufficiency to perform the next task (or process). – completeness at the start of the next task. – availability to those who need them in a time frame to support activities of the next task

Process Management Indicator Table

Level 3	**Well-Defined**

Practice:	**3.2.2**
Perform peer reviews.	**Perform peer reviews of appropriate work products of the process.**
Associated Processes/Practices:	SUP.5 Perform Peer Reviews
Potential Sources for Existence Evidence	– Review strategy/plan (30) – Review records (31) – Corrective actions (97) – Standards and procedures (5) – Project plans (17) – Work breakdown structure (6) – Meeting minutes (19) – Problem tracking systems (98) – Distribution list (77) – Work products list(7)
Process Management Indicators	– Project plans / schedules indicates adequate resources for the reviews are allocated (example: time, appropriate expertise, materials) – Peer reviews are performed for all key work products – list of work products to be reviewed for the process corresponds to those identified in project plan/specification. – Records of peer reviews exist which show: – that appropriate expertise participated in the peer review. – the time spent for the review – statistics about the number of faults found – problems which were identified have – corrective actions plans with target closure dates – status indicators – person responsible for closure – the status of the work product after the review – Quality and coverage criteria is available for the work product reviewed which assess: – the completeness of the work product – the adherence to standards – the coverage of requirements – if the information is understandable – usability for the subsequent task – accuracy and validity – Evidence of corrective action closure exists.

Process Management Indicator Table

Level 3	Well-Defined

Practice:	**3.2.3**
Use well-defined data.	**Use data on performing the defined process to manage the defined process.**
Associated Processes/Practices:	PRO.7.2 Track Progress ORG.2 Define the Process
Potential Sources for Existence Evidence	– Process description (3) – Work breakdown structures (6) – Quality criteria (27) – Process measures (38) – Quality records (28) – Corrective action (97) – Change control records (95)
Process Management Indicators	– Process measures are collected which monitor: – timeliness planned tasks – completeness of planned tasks – sufficiency of the deliverable for the next task. – the quality of the end customer deliverable – usability of the deliverable – Measurement results are used in managing the process: – project management plans reflect the use of the defined process measures – change management criteria reflect the use of process trends – corrective action are defined when data indicates deviations from established processes – correct action results are monitored through the use of process measurement trend data

Process Management Indicator Table

Level 4 Quantitatively-Controlled

Common Feature 4.1: Establishing Measurable Quality Goals

Practice:	**4.1.1**
Establish quality goals.	**Establish measurable quality goals for the work products of the organization's standard process family.**
Associated Processes/Practices:	PRO.5.1 Establish Quality Goals PRO.5.2 Define Quality Metrics ORG.2.1 Define goals
Potential Sources for Existence Evidence	– Quality criteria (27) – Goals (12) – Quality strategy / plan (25) – Quality measures (39) – Process description (3) – Work products (7)
Process Management Indicators	– The organization has defined the desired work product quality characteristics and goals – the quality goals for the work products assesses if the work product is sufficient to satisfy the objectives for its intended use. – internal work product quality goals support the goals for the quality of the end customer product – thresholds are established as part of the quality goals – customer (internal and external) needs and expectations are considered when establishing quality goals – quality goals and measures established are auditable, verifiable, repeatable – cost /benefit analysis is performed to optimize quality goals – Standards exist for the organization's work products of the standard process family which: – define the expected characteristics of the work products – establish what is to be measured – define the source data coverage – define the applicability of the measures – define the usability of measurements – define availability of measurements – specify the source data validation procedures – define key points in the process where work product quality is to be measured

Process Management Indicator Table

Level 4 Quantitatively-Controlled

Common Feature 4.2: Objectively Managing Performance

Practice:	**4.2.1**
Determine process capability.	**Determine the process capability of the defined process quantitatively.**
Associated Processes/Practices:	PRO.5.2 Define quality metrics SUP.3 Perform Quality Assurance ORG.3.3 Understand the process
Potential Sources for Existence Evidence	– Goals (12) – Quality strategy/plan (25) – Quality measures(39) – Quality records (28) – Process performance data(18) – Corrective actions (97) – Assessment / audits records (29) – Process management tools – Assessment instrument repository – Historical records
Process Management Indicators	– Process assessment results are available: – results identify capabilities of the defined process – results are stored for future use – results are measured against available benchmarks, target profiles – Process measures are used to monitor the process performance at key points in the defined process – quality thresholds established are evaluated against actual performance – measurement trend analysis data is used to determine the process capability results – The organization unit measures compliance with the established process tasks activities and the established work product quality characteristics – process adherence is monitored against established criteria and goals – measurement data is available – deviations from established specifications are measured

Process Management Indicator Table

Level 4 Quantitatively-Controlled

Practice:	**4.2.2**
Use process capability.	**Take corrective action as appropriate when the process is not performing within its process capability.**
Associated Processes/Practices:	SUP.4 Perform Problem Resolution. ORG.3 Improve the process
Potential Sources for Existence Evidence	– Corrective action records (97) – Job procedure (4) – Quality goals (12) – Quality strategy/plan(25) – Improvement opportunities (26) – Process measures (38) – Process performance data (16) – Progress status records/report (20) – Meeting minutes (19) – Problem report(84) – Tracking system (98) – Analysis results(21) – Change request(94) – Assessment / audit record(29) – Benchmarking data(43)
Process Management Indicators	– Historical data about the performance of the process is used to identify variation/deviation from defined capability – The organizational unit can show evidence that when established goals are not achieved corrective actions are defined: – when goal thresholds are out of bounds, measures / statistical controls show corrective actions are implemented, and effective – when process' capability results do not reach established targets corrective actions are implemented and effective.

Process Management Indicator Table

Level 5 Continuously-Improving

Common Feature 5.1: Improving Organizational Capability

Practice:	**5.1.1**
Establish process effectiveness goals.	**Establish quantitative goals for improving process effectiveness of the standard process family, based on the business goals of the organization and the current process capability.**
Associated Processes/Practices:	CUS.3.2 Understand the customer expectations ORG.1.1 Establish a strategic vision ORG.3 Improve the Process ORG.2.1 Define Goals
Potential Sources for Existence Evidence	– Process measures (38) – Capability assessment results (29) – Business/organizational plans/goals (12) – Quality goals (12) – Improvement opportunities (26) – Corrective action records (97) – Progress status records/report (20) – Process performance data (16) – Meeting minutes (19) – Problem report (84) – Tracking system (98) – Change request (94) – Quality strategy/plan (25) – Benchmarking data (43)
Process Management Indicators	– Current business/organizational goals have been defined – Current capability assessment results and/or target profiles are available for the process being reviewed – Process capability results and profiles are benchmarked against: – other available data and profiles (industry, internal organizations, historical, etc.) – defined goals (organizational, business, customer expectation) – established target profiles

Process Management Indicator Table

Level 5 Continuously-Improving

Process Management Indicators (continued)

- The goals established reflect:
  - the known process capability and target capability desired
  - establishes a target date for when the desired capability will be achieved
  - identify the potential cost/benefits of planned improvement activities
- The software process effectiveness goals established:
  - optimize the relationship between business needs and customer expectations
  - are achievable within the constraints of the project and allocated resources
  - are measurable
- In establishing quantitative goals for improving effectiveness of the software process consideration was given to:
  - the strategic business goals of the company
  - the customers expectations/needs
  - historical process performance measurement results
  - the factors that impact effectiveness, such as:
    - *economic factors* (productivity, profit, growth, efficiency, quality, competition, resources, and capacity)
    - *human factors:* (job satisfaction, motivation, morale, conflict/cohesion, goal consensus, participation, training, span of control)
    - *management factors:* (skills, commitment, leadership, adaptation, knowledge, ability)
    - *technology factors:* (sophistication of system, technical expertise, development methodology; organizational process capability, adherence to process)

Process Management Indicator Table

Level 5 Continuously-Improving

Practice:	**5.1.2**
Continuously improve the standard process.	**Continuously improve process by changing the organization's standard process family to increase its effectiveness.**
Associated Processes/Practices:	ORG.3 Improve the Process CUS.8 Assess Customer Satisfaction ORG.1.1 Establish a strategic vision CUS.3.2 Understand the customer expectations
Potential Sources for Existence Evidence	– Process measures (38) – Job procedures, practices (4) – Assessment results (29) – Meeting minutes (19) – Goals/objectives: *(effectiveness criteria, business goals)* (12) – Process performance data (18) – Quality measurements (39) – Field measures (41) – Service level measures (42) – Benchmarking data (43) – Process description (3) – Software life cycle model (2) – Waivers to the standard process – Process improvement historical records
Process Management Indicators	– Evidence of potential improvement to the standard processes exists – A Process Improvement plan for the organization's standard process exist which: – identifies the scope of the improvement effort – defines the improvement tasks to be performed – defines the ownership for improvement activities – establishes target dates for completion of improvements – Change procedures for the standard process exist: – analyzes common causes in variations of the processes used by different organizations – analyzes process waivers and the amount of tailoring required – defines factors to consider in prioritizing changes – contains controls for orderly change and transition

Process Management Indicator Table

Level 5	Continuously-Improving

Process Management Indicators (Continued)	– Process improvement history shows: – on-going changes to the organization's standard process – a decreased need for waivers/tailoring of the standard process – confirmation about the effectiveness of the process changes which were performed – Results of corrective actions are monitored against established process measures, and established quality goals to determine if they were effective. – Changes initiated are orderly and controlled: – impacts to organization using the standard process family are assessed before changes are implemented – potential changes are evaluated against the defined process effectiveness criteria – pilot testing of change(s) is performed – potential changes are benchmarked against existing process performance and improvement goals desired – effect of potential process change on current development is considered – goals/objectives to be achieved by process change defined – process change results are monitored for effectiveness – the staff is training of on the new process prior to implementation of the change

Process Management Indicator Table

Level 5 Continuously-Improving

Common Feature 5.2: Improving Process Effectiveness

Practice:	**5.2.1**
Perform Causal Analysis.	**Perform causal analysis of defects**
Associated Processes/Practices:	SUP.4 Perform Problem Resolution ORG.3 Improve the Process PRO.5.4 Perform quality activities PRO.5.5 Assess quality ENG.7.2 Analyze user problems and enhancements
Potential Sources for Existence Evidence	– Analysis results (21) – Problem report (84) – Tracking system (98) – Process descriptions (3) – Work breakdown structure(6) – Review plan (16) – Improvement opportunities (26) – Quality records (28) – Corrective action records (97)
Process Management Indicators	– A mechanism(s) to record defects is available – customer found field failures – defects found in testing product – defects in internal work products – process deficiencies – A causal analysis process is defined which establishes the criteria for: – the resources needed – the expertise needed – the events that would trigger an analysis – grouping, prioritizing and removing defects – the approach to performing an analysis

Process Management Indicator Table

Level 5	Continuously-Improving

Process Management Indicators (continued)	– Causal analysis results: – identified the place in the life cycle where defects were first introduced – identified process deficiencies associated with common product defects – identified product design deficiencies associated with common product defects – identified corrective actions related to product and process deficiencies – Project records show that: – the appropriate expertise was involved in the causal analysis activities – project time was committed to performing the causal analysis – the results of the causal analysis were utilized in process improvement planning

Process Management Indicator Table

Level 5	**Continuously-Improving**

Practice:	**5.2.2**
Eliminate defect causes.	**Eliminate the causes of defects in the defined process selectively.**
Associated Processes/Practices	SUP.4 Perform problem resolution
Potential Sources for Existence Evidence	– Process descriptions (3) – Work breakdown structure (6) – Corrective action records (97) – Change control records (95)* – Quality improvement plan (25) – Quality records (28) – Review records(31) – Measurement reports (37-43)
Process Management Indicators	– Corrective actions from causal analysis: – are prioritized by established criteria – are implemented in a timely manner – Corrective actions implemented are reviewed for effectiveness in solving – process deficiency (inadequate processes, incorrect process) – product defects (faults) – project management deficiencies (staff, resources, time) – A mechanism exists to track and measure the improvement associated with process change – Corrective actions implemented result in a reduction of defects – Corrective actions have associated process improvement plans

Process Management Indicator Table

Level 5 Continuously-Improving

Practice:	**5.2.3**
Continuously improve the defined process.	**Continuously improve process performance by changing the defined process to increase its effectiveness.**
Associated Processes/Practices:	CUS.8 Assess Customer Satisfaction ORG.1.1 Establish a strategic vision ORG.2 Define the Process ORG.3 Improve the Process CUS.3 Identify Customer needs
Potential Sources for Existence Evidence	– Process descriptions (3) – Work breakdown structure (6) – Corrective action records (97) – Change control records (95)* – Quality improvement plan (25) – Improvement opportunities (26) – Quality records (28) – Review records(31) – Measurement reports (37-43) – New process technology analysis – Causal /defect analysis (21) – Process assessment results (29) – process change management procedures (4) – Plans (strategic, improvement, project, etc.) (16) – Process improvement record (28) – Goals, objectives(12) – Effectiveness criteria (27)
Process Management Indicators	– Potential improvement opportunities are identified – Process change procedures exist which: – defines the method for identification of potential changes – prioritize changes – contains controls for orderly change and transition – Process improvement history shows: – on-going changes to the organization's defined process – confirmation about the effectiveness of the changes performed

Process Management Indicator Table

Level 5	Continuously-Improving

Process Management Indicators (continued)	– Changes initiated are orderly and controlled: – the scope of the improvement effort is defined along with targets dates for completion – ownership for improvement efforts are defined – impacts to organization using the standard process family are assessed – potential changes are evaluated against the defined process effectiveness criteria – pilot testing of change(s) is performed – bench marking potential changes against existing process performance and improvement goals desired – effect of potential process change on current development is considered – goals/objectives to be achieved by process change defined – monitoring of process change results – training of the organization on the new process prior to implementation of the change

Annex B (normative)

Process to work product mapping table

Introduction

The purpose of this table is to help the assessor relate the work products found in an organization to the process in part 2 of this International Standard that impact its creation or subsequent use. This information is helpful when reviewing sample work products of an organizational unit, and assessing the adequacy of the process/practices that created the work product.

Work products are produced by the execution of a series of practices defined in a process. Either this table or the table provided in Annex C can be used to help the assessor or tool builder understand this mapping.

The information in this table is similar to the information provided in the Practice Mapping table provided in Annex C, which maps each practice to the work product. This table just provides a process view of the mapping.

The following information describes the fields in the Process Mapping table which is provided in this Annex.

The process identifier — Provides a direct mapping from the process in part 2 of this International Standard to the associated work product(s).

Potential input work product type — Lists a Work Product identifier, followed by the name of the associated work product which would be *input* to the process.

Note: *Each potential input work product has associated characteristics defined in Annex D*

Potential output work product type — Lists a Work Product identifier, followed by the name of the associated work product which would be *output* from the process, updated or created by the practices contained within the process.

Note: *Each potential output work product has associated characteristics defined in Annex D*

NOTE: Within this table the symbol "*reference" is used to show the outputs from of another process, as referenced, may be required to understand if this process is adequately implemented. Rather than duplicate the information in that process, a reference is given to the associated process.

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
CUS.1	83) Customer Request 52) *Internal* Requirements 48) Supplier Proposal Response 49) *Supplier* History record 29) Assessment / Audit record	44) Product Needs Assessment 52) *Product / Service* Requirements 45) Acquisition strategy/plan 47) Request for Proposal 21) Analysis Results
CUS.2	51) Contract 21) Analysis Results 45) Acquisition Strategy	31) Review Records 51) Contract
CUS.3	83) Customer Request 52) *Customer* Requirements	83) Customer Request 46) Market Analysis 87) Communication Mechanism
CUS.4	51) Contract 83) Customer Request 12) Business Goals 24) Quality statement or policy 20) Progress Status Report 84) Problem Report 62) Test Results 52) *Customer* Requirements 59) *Acceptance* Test Plan 3) Process Description 12) Business Goals 24) Quality statement or policy 18) Process Performance Data 38) Process Measures	25) Quality strategy/plan 29) Assessment / Audit record 26) Improvement Opportunity 31) Review Records 30) Review strategy/plan
CUS.5	52) *Customer* Requirements 51) Contract 106) Customer Documentation 73) System	74) Installation Plan 75) Installation Guide 71) Release Notes 78) Delivery instructions 107) Installation Record 76) Packaging Record 70) Release Package 79) Delivery Record 81) Acceptance Record 80) Handling and Storage Guide
CUS.6	84) Problem Report 42) Service Level Measures 41) Field Measures 59) Test Plan 60) Test Script 61) Test Case 73) System 4) Job Procedures 83) Customer Request	22) Risk Analysis 62) Test Results 94) Change Request 42) Service Level Measures 99) Work-around 84) Problem Report 87) Communication Mechanism

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
CUS.7	51) Contract 52) Requirements Specification 52) *Customer* Requirements 41) Field Measures 84) Problem Reports	89) Training Records 87) Communication Mechanism 42) Service Level Measures
CUS.8	85) Customer Satisfaction Survey 41) Field Measures 31) Review records 82) Competitor Information	86) Customer Satisfaction Data 43) Benchmarking Data 87) Communication Mechanism
ENG.1	52) *Customer* Requirements 52) *Maintenance* Requirements 44) Product Needs Assessment 83) Customer Request 94) Change Request 46) Market Analysis	52) *System* Requirements 100) Product Configuration _ 53) System Design / Architecture _ 101) Database Design 58) Traceability record/mapping _ 69) Release strategy /plan
ENG.2	52) *Customer* Requirements 52) *Maintenance* Requirements 44) Product Needs Assessment 83) Customer Request 94) Change Request 53) System Design / Architecture 84) Problem Reports 87) Communication Mechanism	52) *Software* Requirements _ 21) Analysis Results 52) *System* Requirements _ 31) Review records 19) Meeting Minutes 87) Communication Mechanism
ENG.3	52) *Software* Requirements 53) System Design / Architecture	54) High Level Software Design 55) Low Level Software Design 58) Traceability record/mapping
ENG.4	55) Low Level Software Design 101) Database Design 35) Reuse Repository 10) Coding Standards 52) *Software* Requirements 52) *System* Requirements	56) Software units (code) 59) Test Plan 60) *Unit* Test Script 61) Test Case 62) Test Results
ENG.5	52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 95) Change Control 54) High Level Software Design 55) Low Level Software Design 53) System Design / Architecture 56) Software units (code) 69) Release strategy /plan	67) Regression Test Strategy 58) Traceability record/mapping 57) Build Lists 65) Integration Test strategy/plan 60) *Integration* Test Script 64) *Software* Test plan 60) *Software* Test Script 61) Test Case 62) Test Results 72) Integrated Software

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
ENG.6	52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 53) System Design / Architecture 54) High Level Software Design 55) Low Level Software Design 69) Release strategy /plan 108) System Components	57) Build Lists 65) Integration Test strategy/plan 58) Traceability record/mapping 60) *Integration* Test Script 61) Test Case 62) Test Results 66) *System* Test plan 60) *System* Test Script 73) System
ENG.7	52) *Customer* Requirements 83) Customer Request 84) Problem Reports 53) System Design / Architecture 94) Change Request 34) Testing Strategy 67) Regression Test Strategy 95) Change Control	52) *Maintenance* Requirements 21) Analysis Results 95) Change Control record 69) Release strategy /plan _ *Reference. ENG.1, ENG.2, ENG.3, ENG.4, ENG.5, ENG.6, SUP.1, CUS.5, CUS.6, CUS.7, CUS.8
PRO.1	52) *Customer* Requirements 12) Business Goals 22) Risk Analysis 24) Quality Statement / Policy 51) Contract	21) Analysis Results 2) Life Cycle Models 3) Process Description 6) Work Breakdown Structures 25) Quality strategy/plan 30) Review strategy/plan 1) Software Development Methodology
PRO.2	1) Software Development Methodology 52) *Customer* Requirements 12) Business Goals 24) Quality Statement / Policy 33) Reuse Strategy 32) Reuse Plan 35) Reuse Repository 49) Subcontractor or supplier database 25) Quality strategy/plan 2) Life Cycle Models 30) Review strategy/plan	6) Work Breakdown Structure 17) Project Plan _ 9) Standards 10) Coding Standards 104) Development Environment 33) *Projects* Reuse Strategy 11) Estimates 22) Risk Analysis 23) Risk Management Plan 37) Project Measures 5) Schedule 50) Commitment / Agreements
PRO.3	17) Project Plan 89) Training records	4) Job Procedure 14) Policies 87) Communication Mechanism 90) Training Material 50) Commitment / Agreements 119) Meeting Minutes

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
PRO.4	83) Customer Request 21) Analysis Results 22) Risk Analysis record 96) Change History 51) Contract 52) Customer Requirements 6) Work breakdown structure	50) Commitment / Agreements 52) Customer Requirements 51) Contract 95) Change Control 96) Change History 87) Communication Mechanism 58) Traceability record/mapping *Reference: PRO.2, ENG.1, ENG.2, ENG.5, ENG.6, ENG.7, PRO.1
PRO.5	52) Customer Requirements 17) Project Plan 16) Business Plan 24) Quality Statement/Policy 25) Quality Plan 6) Work Break Down Structure	12) Goals (Quality) 25) Quality Plan _ 17) Project Plan _ 25) Quality strategy/plan 39) Quality Measures 6) Work Break Down Structure 18) Process Performance data 97) Corrective Actions
PRO.6	17) Project Plan 12) *Business* Goals 12) *Quality* Goals 1) Software Development Methodology 23) Risk Management strategy/plan 52) *Customer* Requirements 49) Subcontractor or supplier database 11) Estimates 25) Quality strategy/plan 59) Test Plan 74) Installation Plan 37) Project Measures 38) Process Measures 39) Quality Measures 41) Field Measures 42) Service Level Measures 86) Customer Satisfaction Data 31) Review Records 29) Assessment / Audit record 5) Schedule	22) Risk Analysis 40) Risk Measures 23) Risk Management strategy/plan 18) Process Performance data 29) Assessment / Audit Records 97) Corrective Actions *Note: Risk Management strategy / plan could be included as a part of any of the following:* 17) Project Plan 25) Quality strategy/plan 16) Business Plan 45) Acquisition strategy/plan 59) Test Plan (updated as appropriate)

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
PRO.7	17) Project plan 6) Work Breakdown Structure 11) Estimates 5) Schedule 50) Commitment /Agreements 51) Contract 37) Project Measures 30) Review plan 23) Risk Management Plan 25) Quality plan 16) Business plan 45) Acquisition strategy/plan 59) Test plan 52) Requirements	20) Progress Status 31) Review records 29) Assessment / audit record 97) Corrective Actions
PRO.8	17) Project plan 6) Work Breakdown Structure 45) Acquisition strategy/plan 52) *Subcontractor* Requirements 49) *Subcontractor* History record 48) Supplier Proposal Response 25) Quality strategy/plan 39) Quality Measures 27) Quality Criteria 30) Review plan 37) Project Measures 23) Risk Management strategy/plan* 40) Risk Measures 87) Communication Mechanism 18) Process Performance data 59) *Acceptance* Test Plan 51) Contract	*Reference CUS.1 47) Request for Proposal 29) Assessment / Audit record 21) Analysis Results 51) Contract 50) Commitments / Agreements 31) Review records 97) Corrective Actions 26) Improvement Opportunity 87) Communication Mechanism 81) Acceptance Record
SUP.1	52) *Customer* Requirements 83) Customer Request 53) System Design / Architecture 44) Product Needs Assessment 3) Process Description 30) Review plan 59) Test Plan 60) Test Script 61) Test Case 77) Distribution List 78) Delivery Instructions 84) Problem Reports 52) *Maintenance* Requirements 94) Change Request	52) *Documentation* Requirements 18) Process Performance data 106) Customer Documentation 31) Review records 62) Test Results 79) Delivery record 81) Acceptance record 95) Change Control 96) Change History

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
SUP.2	93) Configuration Item 94) Change Request 69) Release strategy /plan	92) Configuration Management (file, library, system) 93) Configuration Item 95) Change Control 57) Build Lists 72) Integrated Software 73) System 70) Release Package 96) Change History 20) Progress Status record / report
SUP.3	52) Requirements 1) Software Development Methodology 38) Process Measures 39) Quality Measures 17) Project plan 25) Quality strategy/plan 30) Review strategy/plan 37) Project Measures 27) Quality Criteria	9) Standards 10) Coding Standards 44) Product Needs Assessment 31) Review records 29) Assessment / Audit record 20) Progress Status record / report 19) Meeting minutes 97) Corrective Actions
SUP.4	83) Customer Request 31) Review Records 84) Problem Report * References: CUS.4.3, CUS.4.5, CUS.6.5, CUS.6.6, CUS,7.3 CUS.7.4, CUS.7.5, ENG5.4, ENG5.6, ENG.6.3, ENG.6.5, ENG.7.2, SUP.3.6, SUP.5.7	84) Problem Report 98) Tracking system 21) Analysis Results 97) Corrective Actions 94) Change Request 69) Release strategy /plan 52) *Maintenance* Requirements 95) Change Control 6) Work Breakdown Structure 17) Project Plan * References:ORG.3, ENG.7, CUS.5
SUP.5	17) Project plan 6) Work Breakdown Structures 25) Quality strategy/plan 1) Software Development Methodology 9) Standards 10) Coding Standards 52) Requirements 27) Quality Criteria	30) Review strategy/plan 77) Distribution list 31) Review Records 97) Corrective Actions
ORG.1	12) *Business* Goals 17) Project Plan	13) Vision 14) Policies 87) Communication Mechanism 19) Meeting Minutes 17) Project Plan 4) Job Procedure 86) Customer Satisfaction Data 25) Quality strategy/plan _ 15) Personnel Policies 89) Training records

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
ORG.2	12) *Business* Goals 13) Vision 14) Policies 1) Software Development Methodology 2) Life Cycle Models 24) Quality Statement or Policy 52) *Product / Service* Requirements 9) Standards 39) Quality Measures	12) *Process* Goals 3) Process Description 6) Work Breakdown Structures 4) Job Procedure 7) Work Products 27) Quality Criteria 30) Review strategy/plan 8) Interfaces 38) Process Measures 17) Project plan 87) Communication Mechanism 77) Distribution List 89) Training records 28) Quality records
ORG.3	52) Requirements 9) Standards 44) Product Needs Assessment 51) Contract 29) Assessment / Audit record 97) Corrective Actions 21) Analysis Results 31) Review Records 83) Customer Request 46) Market Analysis 12) Business Goals 84) Problem Report 18) Process Performance Data 86) Customer Satisfaction Data 22) Risk Analysis 43) Benchmarking Data 38) Process Measures 39) Quality Measures 42) Service Level Measures 3) Process Description 12) *Process* Goals 27) Quality Criteria	26) Improvement Opportunity 25) Quality strategy/plan 29) Assessment / Audit record 38) Process Measures 43) Benchmarking data 3) Process Description 17) Project plan 87) Communication mechanism 77) Distribution list 89) Training records
ORG.4	89) Training records 44) Product Needs Assessment 6) Work Breakdown Structures 23) Risk Management 17) Project Plan 16) Business Plan 4) Job Procedures, Practices 9) Standards 15) Personnel Policies	88) Training strategy/plan 89) Training records 9) Training Material

Process to Work Product Mapping Table

Process	Input work product type	Output work product type
ORG.5	1) Software Development Methodology 2) Life Cycle Models 12) Business Goals 7) any Work Products (i.e., designs, code, architecture, tests, etc.) 14) Policies 9) Standards 23) Risk Management 17) Project Plan 16) Business Plan 94) Change Request 84) Problem Reports 95) Change Control 52) *Maintenance* Requirements	33) Reuse Strategy 32) Reuse Plan 9) Standards 34) Reusable Object 35) Reuse Repository 95) Change Control 31) Review Records 1) Software Development Methodology 2) Life Cycle Models 3) Process Description *Reference: ENG.3, ENG.4, ENG.5, ENG.6, ENG.7, SUP.1
ORG.6	44) Product Needs Assessment 94) Change Request 84) Problem Reports 95) Change Control 52) *Maintenance* Requirements	52) *Environment* Requirements 17) Project Plan 104) Development Environment * References: CUS.6, CUS.7, ENG.7
ORG.7	52) *Environment* Requirements 17) Project plan 25) Quality strategy/plan	104) Development Environment 103) Recovery plan 102) Backup/recovery records * Reference: ORG.6

Annex C (normative)

Base practice to work product mapping table

Introduction

The purpose of this base practice to work product mapping table is to help the assessor or tool builder relate work products and their defined characteristics to the practices in part 2 of this International Standard which impact their creation or subsequent use.

The fields in the base practice to work product mapping table contain the following information.

Base practice identifier:	Provides a direct mapping from the base practice in part 3 of this International Standard to the associated work product(s).
Potential input work product type:	Lists a work product identifier, followed by the name of the associated work product which would be *input* to the process or practice. Each *input work product has associated characteristics defined in Annex D*
Potential output work product type	Lists a work product identifier, followed by the name of the associated work product which would be *output* from the practice. *Each output work product has associated characteristics defined in Annex D*

NOTE: Within this table several symbols are used as follows:

(1) The symbol Δ is used to imply that the work product is built incrementally from the execution of several practices, or processes. Each practice may add a piece of information to the eventual output work product. When looking at these work products the assessor will need to judge the adequacy of the practices that produced the work product. Sometimes the adequacy of these practices can not be viewed in isolation, but rather should be viewed in how they contribute to achieving the purpose of the process. The work products produced and their usefulness to achieving the process purpose help determine this judgement.

(2) The symbol "*reference" is used to show that one actually may need to look at the outputs from a whole process as referenced to understand if this practice is adequately implemented. Rather than duplicate the information in that process, a reference is given to the associated process.

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
CUS.1.1	83) Customer Request 52) *Internal* Requirements	44) Product Needs Assessment
CUS.1.2	44) Product Needs Assessment	52) *Product / Service* Requirements
CUS.1.3	52) *Product / Service* Requirements	45) Acquisition strategy/plan
CUS.1.4	52) *Product / Service* Requirements 45) Acquisition strategy/plan	47) Request for Proposal
CUS.1.5	48) Supplier Proposal Response 49) *Supplier* History record 29) Assessment / Audit record	21) Analysis Results
CUS.2.1	51) Contract	31) Review Records
CUS.2.2	51) Contract 21) Analysis Results	51) Contract
CUS.2.3	45) Acquisition Strategy 21) Analysis Results	51) Contract
CUS.2.4	45) Acquisition Strategy 21) Analysis Results	51) Contract
CUS.3.1	83) Customer Request	83) Customer Request 46) Market Analysis
CUS.3.2	83) Customer Request 46) Market Analysis 52) *Customer* Requirements	83) Customer Request 46) Market Analysis
CUS.3.3	83) Customer Request	87) Communication Mechanism
CUS.4.1	51) Contract 83) Customer Request 12) Business Goals 24) Quality statement or policy	25) Quality strategy/plan
CUS.4.2	25) Quality strategy/plan	30) Review strategy/plan
CUS.4.3	20) Progress Status Report 30) Review strategy/plan	31) Review Records
CUS.4.4	84) Problem Report 62) Test Results 83) Customer Request 30) Review strategy/plan	31) Review Records

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
CUS.4.5	52) *Customer* Requirements 59) *Acceptance* Test Plan	31) Review Records
CUS.4.6	3) Process Description 12) Business Goals 24) Quality statement or policy 18) Process Performance Data 38) Process Measures	29) Assessment / Audit record 26) Improvement Opportunity
CUS.5.1	52) *Customer* Requirements 51) Contract	74) Installation Plan 75) Installation Guide 71) Release Notes 78) Delivery instructions
CUS.5.2	74) Installation Plan	107) Installation Record
CUS.5.3	74) Installation Plan 106) Customer Documentation 71) Release Notes 73) System	76) Packaging Record 70) Release Package
CUS.5.4	74) Installation Plan 78) Delivery instructions 70) Release Package	79) Delivery Record
CUS.5.5	70) Release Package 79) Delivery Record 78) Delivery instructions	81) Acceptance Record
CUS.5.6	74) Installation Plan 75) Installation Guide 70) Release Package	81) Acceptance Record
CUS.5.7	74) Installation Plan	80) Handling and Storage Guide
CUS.6.1	84) Problem Report 42) Service Level Measures 41) Field Measures	22) Risk Analysis
CUS.6.2	59) Test Plan 60) Test Script 61) Test Case	62) Test Results
CUS.6.3	4) Job Procedures	none
CUS.6.4	84) Problem Report 73) System	94) Change Request * Reference SUP.4
CUS.6.5	83) Customer Request	84) Problem Report 87) Communication Mechanism

Practice to Work Product Mapping Table

Practice	**Input work product type**	**Output work product type**
CUS.6.6	84) Problem Report	99) Work-around
CUS.6.7	73) System	42) Service Level Measures
CUS.7.1	51) Contract 52) Requirements Specification	89) Training Records
CUS.7.2	51) Contract 52) *Customer* Requirements	87) Communication Mechanism
CUS.7.3	41) Field Measures 84) Problem Reports	42) Service Level Measures
CUS.7.4	51) Contract 52) *Customer* Requirements	*Reference: ENG.7 and CUS.5
CUS.8.1	85) Customer Satisfaction Survey 41) Field Measures 31) Review records	86) Customer Satisfaction Data
CUS.8.2	86) Customer Satisfaction Data 82) Competitor Information	43) Benchmarking Data
CUS.8.3	86) Customer Satisfaction Data 43) Benchmarking Data	87) Communication Mechanism
ENG.1.1	52) *Customer* Requirements 52) *Maintenance* Requirements 44) Product Needs Assessment 83) Customer Request 94) Change Request	52) *System* Requirements
ENG.1.2	52) *System* Requirements	100) Product Configuration Δ 53) System Design / Architecture Δ
ENG.1.3	52) *System* Requirements 53) System Design / Architecture	100) Product Configuration 101) Database Design 53) System Design / Architecture Δ 58) Traceability record/mapping Δ
ENG.1.4	52) *System* Requirements 44) Product Needs Assessment 83) Customer Request 94) Change Request 46) Market Analysis	69) Release strategy /plan

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
ENG.2.1	52) *Customer* Requirements 52) *Maintenance* Requirements 44) Product Needs Assessment 83) Customer Request 94) Change Request 53) System Design / Architecture	52) *Software* Requirements
ENG.2.2	52) *Software* Requirements 44) Product Needs Assessment 83) Customer Request 53) System Design / Architecture	21) Analysis Results 52) *Software* Requirements Δ
ENG.2.3	52) *Software* Requirements 53) System Design / Architecture	21) Analysis Results 52) *System* Requirements Δ 52) *Software* Requirements Δ
ENG.2.4	52) *Software* Requirements	31) Review records 19) Meeting Minutes 87) Communication Mechanism 52) *Software* Requirements Δ
ENG.2.5	52) *Software* Requirements 84) Problem Reports 83) Customer Request 87) Communication Mechanism	52) *Software* Requirements Δ
ENG.3.1	52) *Software* Requirements 53) System Design / Architecture	54) High Level Software Design Δ
ENG.3.2	54) High Level Software Design 52) *Software* Requirements 53) System Design / Architecture	54) High Level Software Design Δ
ENG.3.3	54) High Level Software Design	55) Low Level Software Design
ENG.3.4	52) *Software* Requirements 54) High Level Software Design 55) Low Level Software Design	58) Traceability record/mapping
ENG.4.1	55) Low Level Software Design 101) Database Design 35) Reuse Repository 10) Coding Standards	56) Software units (code)
ENG.4.2	55) Low Level Software Design 52) *Software* Requirements 52) *System* Requirements	59) Test Plan 60) *Unit* Test Script 61) Test Case

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
ENG.4.3	59) Test Plan 60) *Unit* Test Script 61) Test Case 56) Software units (code)	62) Test Results
ENG.5.1	52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 95) Change Control	67) Regression Test Strategy 58) Traceability record/mapping
ENG.5.2	52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 53) System Design / Architecture 54) High Level Software Design 55) Low Level Software Design 69) Release strategy /plan	57) Build Lists 65) Integration Test strategy/plan 58) Traceability record/mapping
ENG.5.3	57) Build Lists 65) Integration Test strategy/plan 52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 53) System Design / Architecture 54) High Level Software Design 55) Low Level Software Design	60) *Integration* Test Script 61) Test Case
ENG.5.4	65) Integration Test strategy/plan 60) *Integration* Test Script 61) Test Case 56) Software units (code)	62) Test Results
ENG.5.5	52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 53) System Design / Architecture 54) High Level Software Design 55) Low Level Software Design	58) Traceability record/mapping 64) *Software* Test plan 60) *Software* Test Script 61) Test Case
ENG.5.6	64) *Software* Test plan 60) *Software* Test Script 61) Test Case 72) Integrated Software	62) Test Results
ENG.6.1	52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 53) System Design / Architecture 54) High Level Software Design 55) Low Level Software Design 69) Release strategy /plan 108) System Components	57) Build Lists 65) Integration Test strategy/plan 58) Traceability record/mapping

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
ENG.6.2	57) Build Lists 65) Integration Test strategy/plan 52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 53) System Design / Architecture 54) High Level Software Design 55) Low Level Software Design	60) *Integration* Test Script 61) Test Case
ENG.6.3	60) *Integration* Test Script 61) Test Case	62) Test Results
ENG.6.4	52) *System* Requirements 52) *Software* Requirements 52) *Maintenance* Requirements 53) System Design / Architecture 54) High Level Software Design 55) Low Level Software Design	66) *System* Test plan 60) *System* Test Script 61) Test Case
ENG.6.5	60) *System* Test Script 61) Test Case 73) System	62) Test Results
ENG.7.1	52) *Customer* Requirements 83) Customer Request 84) Problem Reports 53) System Design / Architecture	52) *Maintenance* Requirements
ENG.7.2	52) *Maintenance* Requirements 83) Customer Request 84) Problem Reports 53) System Design / Architecture	21) Analysis Results
ENG.7.3	21) Analysis Results 83) Customer Request 94) Change Request 84) Problem Reports 34) Testing Strategy 67) Regression Test Strategy	95) Change Control record 69) Release strategy /plan Δ 52) *Maintenance* Requirements
ENG.7.4	95) Change Control 52) *Maintenance* Requirements	*Reference: ENG.1, ENG.2, ENG.3, ENG.4, ENG.5, ENG.6, SUP.1
ENG.7.5	95) Change Control 52) *Maintenance* Requirements	*Reference: CUS.5, CUS.6, CUS.7, CUS.8, SUP.1
PRO.1.1	51) Contract 52) *Customer* Requirements 12) Business Goals 22) Risk Analysis 24) Quality Statement / Policy	21) Analysis Results

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
PRO.1.2	52) *Customer* Requirements 12) Business Goals 22) Risk Analysis 24) Quality Statement / Policy	2) Life Cycle Models
PRO1.3	2) Life Cycle Models 21) Analysis Results 52) *Customer* Requirements 12) Business Goals 24) Quality Statement / Policy 22) Risk Analysis	3) Process Description 6) Work Breakdown Structures
PRO.1.4	2) Life Cycle Models 3) Process Description 6) Work Breakdown Structure 21) Analysis Results 52) *Customer* Requirements 12) Business Goals 24) Quality Statement / Policy 22) Risk Analysis 24) Quality Statement / Policy	3) Process Description 6) Work Breakdown Structure 25) Quality strategy/plan 30) Review strategy/plan
PRO.1.5	2) Life Cycle Models 3) Process Description 6) Work Breakdown Structure 21) Analysis Results 52) *Customer* Requirements 12) Business Goals 22) Risk Analysis	1) Software Development Methodology
PRO.2.1	1) Software Development Methodology 52) *Customer* Requirements 12) Business Goals 24) Quality Statement / Policy	6) Work Breakdown Structure 17) Project Plan Δ
PRO.2.2	52) *Customer* Requirements 12) Business Goals 24) Quality Statement / Policy	9) Standards 10) Coding Standards 17) Project Plan Δ
PRO.2.3	52) *Customer* Requirements 12) Business Goals	104) Development Environment 17) Project Plan Δ
PRO.2.4	33) Reuse Strategy 32) Reuse Plan 35) Reuse Repository 1) Software Development Methodology 52) *Customer* Requirements	33) *Projects* Reuse Strategy 17) Project Plan Δ
PRO.2.5	52) *Customer* Requirements 12) Business Goals	11) Estimates 17) Project Plan Δ

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
PRO.2.6	52) *Customer* Requirements 12) Business Goals 49) Subcontractor or supplier database	22) Risk Analysis 23) Risk Management 17) Project Plan Δ
PRO.2.7	6) Work Breakdown Structure 17) Project Plan 1) Software Development Methodology 52) *Customer* Requirements 52) *Customer* Requirements 12) Business Goals 49) Subcontractor or supplier database 11) Estimates 23) Risk Management Plan 25) Quality strategy/plan	37) Project Measures 17) Project Plan Δ
PRO.2.8	6) Work Breakdown Structure 17) Project Plan Δ 1) Software Development Methodology 2) Life Cycle Models 23) Risk Management Plan 52) *Customer* Requirements 11) Estimates	5) Schedule 17) Project Plan Δ
PRO.2.9	52) *Customer* Requirements 17) Project Plan 11) Estimates 5) Schedule 6) Work Breakdown Structure 25) Quality strategy/plan 30) Review strategy/plan	50) Commitment / Agreements 17) Project Plan Δ
PRO2.10	6) Work Breakdown Structure 17) Project Plan Δ 1) Software Development Methodology 2) Life Cycle Models 23) Risk Management Plan 52) *Customer* Requirements 11) Estimates	17) Project Plan
PRO.3.1	17) Project Plan 89) Training records	4) Job Procedure 14) Policies
PRO.3.2	89) Training records	87) Communication Mechanism 90) Training Material 14) Policies
PRO.3.3	17) Project Plan	50) Commitment / Agreements 19) Meeting Minutes

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
PRO.3.4	17) Project Plan	17) Project Plan 19) Meeting Minutes
PRO.4.1	83) Customer Request 21) Analysis Results 22) Risk Analysis record	50) Commitment / Agreements / Agreements
PRO.4.2	83) Customer Request 21) Analysis Results 22) Risk Analysis record	52) Customer Requirements 51) Contract
PRO.4.3	51) Contract 52) Customer Requirements 83) Customer Request 96) Change History 21) Analysis Results 22) Risk Analysis record	95) Change Control 96) Change History 52) Customer Requirements 87) Communication Mechanism
PRO.4.4	52) Customer Requirements	*Reference: PRO.2,ENG.1,ENG.2,ENG.5 ENG.6, ENG.7, PRO.1
PRO.4.5	52) Customer Requirements 6) Work breakdown structure	58) Traceability record/mapping
PRO.5.1	52) Customer Requirements 17) Project Plan 16) Business Plan 24) Quality Statement/Policy 25) Quality Plan	12) Quality Goals 25) Quality Plan ∆ 17) Project Plan ∆
PRO.5.2	52) Customer Requirements 17) Project Plan 16) Business Plan 6) Work Break Down Structure 25) Quality Plan 24) Quality Statement/Policy	25) Quality strategy/plan ∆ 39) Quality Measures 17) Project Plan ∆
PRO.5.3	52) Customer Requirements 17) Project Plan 16) Business Plan 6) Work Break Down Structure 25) Quality Plan 24) Quality Statement/Policy	25) Quality Plan ∆ 17) Project Plan ∆ 6) Work Break Down Structure ∆
PRO.5.4	25) Quality Plan 24) Quality Statement/Policy 17) Project Plan ∆	18) Process Performance data 17) Project Plan ∆

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
PRO.5.5	39) Quality Measures 25) Quality Plan Δ 24) Quality Statement/Policy 12) *Quality* Goals 17) Project Plan Δ	18) Process Performance data 17) Project Plan Δ
PRO.5.6	25) Quality Plan Δ 24) Quality Statement/Policy	97) Corrective Actions (update as appropriate) 25) Quality Plan Δ 17) Project Plan Δ
PRO.6.1	17) Project Plan 12) *Business* Goals 12) *Quality* Goals 1) Software Development Methodology 23) Risk Management strategy/plan 52) *Customer* Requirements 49) Subcontractor or supplier database 11) Estimates 25) Quality strategy/plan 59) Test Plan 74) Installation Plan	23) Risk Management strategy/plan Δ *Note: Risk Management strategy / plan could be included as a part of any of the following:* 17) Project Plan Δ 25) Quality strategy/plan Δ 16) Business Plan Δ 45) Acquisition strategy/plan Δ 59) Test Plan Δ
PRO.6.2	23) Risk Management strategy/plan 37) Project Measures 38) Process Measures 39) Quality Measures 41) Field Measures 42) Service Level Measures 86) Customer Satisfaction Data 31) Review Records 29) Assessment / Audit record 5) Schedule	22) Risk Analysis Δ 23) Risk Management Plan* Δ
PRO.6.3	37) Project Measures 38) Process Measures 5) Schedule 17) Project Plan 23) Risk Management strategy/plan 39) Quality Measures 12) Goals (Quality) 25) Quality strategy/plan	22) Risk Analysis Δ 23) Risk Management strategy/plan Δ
PRO.6.4	22) Risk Analysis	23) Risk Management strategy/plan Δ
PRO.6.5	23) Risk Management strategy/plan	40) Risk Measures Δ
PRO.6.6	23) Risk Management strategy/plan	40) Risk Measures Δ 18) Process Performance data
PRO.6.7	23) Risk Management strategy/plan 40) Risk Measures	40) Risk Measures Δ 29) Assessment / Audit Records

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
PRO.6.8	29) Assessment / Audit Records	97) Corrective Actions (updated as appropriate) 17) Project Plan Δ 25) Quality strategy/plan Δ 16) Business Plan Δ 45) Acquisition strategy/plan Δ 59) Test Plan Δ
PRO.7.1	17) Project Plan 6) Work Breakdown Structure 11) Estimates 5) Schedule 50) Commitment /Agreements 51) Contract	20) Progress Status
PRO.7.2	37) Project Measures	20) Progress Status
PRO.7.3	30) Review plan 23) Risk Management Plan 5) Schedule 17) Project plan 25) Quality plan 16) Business plan 45) Acquisition strategy/plan 59) Test plan 6) Work Breakdown Structure 52) Requirements	31) Review records 29) Assessment / Audit record 97) Corrective Actions
PRO.7.4	30) Review plan 23) Risk Management Plan 5) Schedule 17) Project plan 25) Quality plan 45) Acquisition strategy/plan 59) Test plan 6) Work Breakdown Structure 52) Requirements	31) Review records 29) Assessment / Audit record 97) Corrective Actions
PRO.7.5	51) Contract 50) Commitment / Agreements	31) Review records 29) Assessment / Audit record 97) Corrective Actions
PRO.8.1	17) Project plan 6) Work Breakdown Structure 52) Requirements 45) Acquisition strategy/plan	*Reference CUS.1 47) Request for Proposal
PRO.8.2	52) *Subcontractor* Requirements 47) Request for Proposal	29) Assessment / Audit record

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
PRO.8.3	49) *Subcontractor* History record 29) Assessment / Audit record 48) Supplier Proposal Response	* Reference CUS.1 21) Analysis Results 51) Contract 50) Commitments / Agreements
PRO.8.4	50) Commitments / Agreements 51) Contract 25) Quality strategy/plan 39) Quality Measures 30) Review plan 17) Project Plan 37) Project Measures 23) Risk Management strategy/plan* 40) Risk Measures	50) Commitments / Agreements 51) Contract 31) Review records 29) Assessment / Audit record 97) Corrective Actions 26) Improvement Opportunity
PRO.8.5	87) Communication Mechanism	87) Communication Mechanism
PRO.8.6	51) Contract 52) *Subcontractor* Requirements 17) Project Plan 25) Quality strategy/plan 27) Quality Criteria 39) Quality Measures 18) Process Performance data 38) Process Measures	31) Review records 29) Assessment / audit record 97) Corrective Actions
PRO.8.7	59) *Acceptance* Test Plan 17) Project Plan 30) Review strategy/plan 25) Quality strategy/plan 27) Quality Criteria 39) Quality Measures 52) *Subcontractor* Requirements 51) Contract	81) Acceptance Record 31) Review records 29) Assessment / Audit record 97) Corrective Actions
SUP.1.1	52) *Customer* Requirements 83) Customer Request 53) System Design / Architecture 44) Product Needs Assessment 3) Process Description ∆	52) *Documentation* Requirements
SUP.1.2	52) *Documentation* Requirements	18) Process Performance data 106) Customer Documentation
SUP.1.3	52) *Customer* Requirements 52) *Documentation* Requirements 53) System Design / Architecture 3) Process Description ∆ 30) Review plan 59) Test Plan 60) Test Script 61) Test Case	31) Review records 62) Test Results

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
SUP.1.4	106) Customer Documentation 77) Distribution List 78) Delivery Instructions	*Reference CUS.5 79) Delivery Record 81) Acceptance Record
SUP.1.5	83) Customer Request 84) Problem Reports 52) *Documentation* Requirements 52) *Maintenance* Requirements 94) Change Request	*Reference: SUP.1, SUP.2, ENG.7 106) Customer Documentation Δ 95) Change Control 96) Change History
SUP.2.1	93) Configuration Item	92) Configuration Management (file, library, system)
SUP.2.2	94) Change Request 69) Release strategy /plan	92) Configuration Management (file, library, system) 93) Configuration Item
SUP.2.3	93) Configuration Item	92) Configuration Management (file, library, system)
SUP.2.4	94) Change Request	95) Change Control
SUP.2.5	94) Change Request 69) Release strategy /plan	95) Change Control 57) Build Lists
SUP.2.6	95) Change Control 57) Build Lists 69) Release strategy /plan 92) Configuration Management (file, library, system)	72) Integrated Software 73) System 70) Release Package
SUP.2.7	95) Change Control 92) Configuration Management (file, library, system)	96) Change History
SUP.2.8	95) Change Control	20) Progress Status record / report
SUP.3.1	52) Requirements 1) Software Development Methodology	9) Standards 10) Coding Standards 44) Product Needs Assessment
SUP.3.2	1) Software Development Methodology 9) Standards 10) Coding Standards 52) Requirements 38) Process Measures 39) Quality Measures 17) Project Plan 25) Quality strategy/plan 30) Review strategy/plan	31) Review Records 29) Assessment / Audit record

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
SUP.3.3	1) Software Development Methodology 9) Standards 10) Coding Standards 52) Requirements 37) Project Measures 38) Process Measures 39) Quality Measures 25) Quality strategy/plan 27) Quality Criteria 17) Project Plan 30) Review strategy/plan	31) Review Records 29) Assessment / Audit record
SUP.3.4	17) Project Plan 25) Quality strategy/plan	20) Progress Status record / report 19) Meeting Minutes
SUP.3.5	31) Review Records 29) Assessment / Audit record	97) Corrective Actions

SUP.4.1	83) Customer Request 31) Review Records * References: CUS.4.3, CUS.4.5, CUS.6.5, CUS.6.6, CUS.7.3, CUS.7.4, CUS.7.5, ENG5.4, ENG5.6, ENG.6.3, ENG.6.5, ENG.7.2, SUP.3.6, SUP.5.7	84) Problem Report ∆
SUP.4.2	84) Problem Report	98) Tracking system
SUP.4.3	84) Problem Report	21) Analysis Results 84) Problem Report ∆
SUP.4.4	21) Analysis Results	84) Problem Report ∆ 97) Corrective Actions 94) Change Request 69) Release strategy /plan 52) *Maintenance* Requirements
SUP.4.5	94) Change Request 69) Release strategy /plan 52) *Maintenance* Requirements 97) Corrective Actions	95) Change Control 6) Work Breakdown Structure 17) Project Plan ∆ * Reference: ORG.3, ENG.7
SUP.4.6	95) Change Control	* Reference: ORG.3, ENG.7, CUS.5

SUP.5.1	17) Project Plan 6) Work Breakdown Structures 25) Quality strategy/plan	30) Review strategy/plan ∆

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
SUP.5.2	1) Software Development Methodology 9) Standards 10) Coding Standards 52) Requirements	30) Review strategy/plan Δ
SUP.5.3	27) Quality Criteria	30) Review strategy/plan Δ
SUP.5.4	27) Quality Criteria	30) Review strategy/plan Δ
SUP.5.5	30) Review strategy/plan	77) Distribution list
SUP.5.6	30) Review strategy/plan	31) Review Records Δ
SUP.5.7	31) Review Records Δ	31) Review Records Δ 97) Corrective Actions Δ
SUP.5.8	31) Review Records 97) Corrective Actions	31) Review Records Δ 97) Corrective Actions Δ
ORG.1.1	12) *Business* Goals	13) Vision 14) Policies
ORG.1.2	13) Vision	87) Communication Mechanism 19) Meeting Minutes 17) Project Plan 4) Job Procedure
ORG.1.3	12) *Business* Goals	86) Customer Satisfaction Data 25) Quality strategy/plan Δ 4) Job Procedure
ORG.1.4	17) Project Plan	* Reference: PRO.3
ORG.1.5	17) Project Plan	14) Policies 15) Personnel Policies
ORG.1.6	17) Project Plan	15) Personnel Policies 89) Training records
ORG.2.1	12) *Business* Goals 13) Vision 14) Policies 1) Software Development Methodology 2) Life Cycle Models	12) *Process* Goals
ORG.2.2	12) *Process* Goals 1) Software Development Methodology 2) Life Cycle Models	3) Process Description Δ 6) Work Breakdown Structure 4) Job Procedure

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
ORG.2.3	12) *Process* Goals 1) Software Development Methodology 2) Life Cycle Models	3) Process Description Δ 7) Work Products
ORG.2.4	12) *Process* Goals 24) Quality statement or policy 1) Software Development Methodology 2) Life Cycle Models	3) Process Description Δ 27) Quality Criteria
ORG.2.5	12) *Process* Goals 24) Quality statement or policy 1) Software Development Methodology 2) Life Cycle Models	3) Process Description Δ 30) Review strategy/plan 27) Quality Criteria
ORG.2.6	3) Process Description 1) Software Development Methodology 2) Life Cycle Models	3) Process Description Δ 8) Interfaces
ORG.2.7	3) Process Description 2) Life Cycle Models 3) Process Description	3) Process Description Δ 8) Interfaces
ORG.2.8	3) Process Description 2) Life Cycle Models 3) Process Description 52) *Product / Service* Requirements 9) Standards 24) Quality statement or policy	3) Process Description Δ 28) Quality records 38) Process Measures
ORG.2.9	3) Process Description 2) Life Cycle Models 3) Process Description 52) *Product / Service* Requirements 9) Standards 24) Quality statement or policy	3) Process Description Δ 38) Process Measures
ORG.2.10	3) Process Description 6) Work Breakdown Structures 4) Job Procedure 27) Quality Criteria 39) Quality Measures 7) Work Products 30) Review strategy/plan 8) Interfaces	3) Process Description * Reference SUP.1
ORG.2.11	3) Process Description	14) Policies 9) Standards
ORG.2.12	3) Process Description	27) Quality Criteria 38) Process Measures

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
ORG.2.13	3) Process Description	17) Project Plan 87) Communication Mechanism 77) Distribution List 89) Training records
ORG.3.1	52) Requirements 9) Standards 44) Product Needs Assessment 51) Contract 29) Assessment / Audit record 97) Corrective Actions ∆ 21) Analysis Results 31) Review Records 83) Customer Request 46) Market Analysis 12) Business Goals 84) Problem Report 18) Process Performance Data 86) Customer Satisfaction Data 22) Risk Analysis 43) Benchmarking Data 38) Process Measures 39) Quality Measures 42) Service Level Measures	26) Improvement Opportunity
ORG.3.2	26) Improvement Opportunity	25) Quality strategy/plan ∆
ORG.3.3	3) Process Description	29) Assessment / Audit record
ORG.3.4	26) Improvement Opportunity 97) Corrective Actions ∆ 12) *Process* Goals 3) Process Description 29) Assessment / Audit record 38) Process Measures	25) Quality strategy/plan ∆
ORG.3.5	26) Improvement Opportunity 97) Corrective Actions ∆ 12) *Process* Goals 3) Process Description	25) Quality strategy/plan ∆
ORG.3.6	26) Improvement Opportunity 97) Corrective Actions ∆ 12) *Process* Goals 3) Process Description	38) Process Measures 43) Benchmarking Data
ORG.3.7	26) Improvement Opportunity 97) Corrective Actions ∆ 12) *Process* Goals 3) Process Description 25) Quality strategy/plan ∆	3) Process Description ∆ * Reference: ORG.2

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
ORG.3.8	26) Improvement Opportunity 97) Corrective Actions 3) Process Description 38) Process Measures 43) Benchmarking Data 18) Process Performance Data 86) Customer Satisfaction Data 25) Quality strategy/plan Δ 12) *Process* Goals 27) Quality Criteria	38) Process Measures Δ 43) Benchmarking Data
ORG.3.9	3) Process Description	17) Project Plan 87) Communication Mechanism 77) Distribution List 89) Training records
ORG.4.1	89) Training records 44) Product Needs Assessment 6) Work Breakdown Structures 23) Risk Management 17) Project Plan 16) Business Plan 4) Job Procedures, Practices 9) Standards 15) Personnel Policies	88) Training strategy/plan
ORG.4.2	88) Training strategy/plan	89) Training records 90) Training Material
ORG.4.3	88) Training strategy/plan 17) Project Plan Δ 44) Product Needs Assessment 6) Work Breakdown Structures	89) Training records
ORG.4.4	88) Training strategy/plan	89) Training records
ORG.5.1	1) Software Development Methodology 2) Life Cycle Models 12) Business Goals 7) any Work Product (i.e., designs, code, architecture, tests, etc.) 14) Policies 9) Standards 23) Risk Management 17) Project Plan 16) Business Plan	33) Reuse Strategy 84) Reuse Plan Δ
ORG.5.2	33) Reuse Strategy 84) Reuse Plan	84) Reuse Plan Δ 9) Standards

Practice to Work Product Mapping Table

Practice	Input work product type	Output work product type
ORG.5.3	84) Reuse Plan	34) Reusable Object *Reference. ENG.3, ENG.4, ENG.5, ENG.6, SUP.1
ORG.5.4	34) Reusable Object	35) Reuse Repository
ORG.5.5	9) Standards 34) Reusable Object	95) Change Control 31) Review Records ∆ 35) Reuse Repository
ORG.5.6	17) Project Plan 84) Reuse Plan 35) Reuse Repository	1) Software Development Methodology ∆ 2) Life Cycle Models ∆ 3) Process Description ∆
ORG.5.7	94) Change Request 84) Problem Reports 95) Change Control 52) *Maintenance* Requirements	*Reference ENG.7
ORG.6.1	44) Product Needs Assessment	52) *Environment* Requirements 17) Project Plan ∆
ORG.6.2	17) Project Plan	104) Development Environment
ORG.6.3	104) Development Environment	* Reference CUS.6, CUS.7
ORG.6.4	94) Change Request 84) Problem Reports 95) Change Control 52) *Maintenance* Requirements	* Reference ENG.7
ORG.7.1	52) *Environment* Requirements 17) Project Plan	104) Development Environment ∆ * Reference ORG.6
ORG.7.2	* Reference ORG.6 104) Development Environment 17) Project Plan 25) Quality strategy/plan	104) Development Environment ∆ 103) Recovery plan
ORG.7.3	103) Recovery plan	102) Backup/recovery records
ORG.7.4	52) *Environment* Requirements 17) Project Plan	104) Development Environment ∆ * Reference ORG.6
ORG.7.5	52) *Environment* Requirements 17) Project Plan	104) Development Environment ∆ * Reference ORG.6

Annex D (normative)

Process indicators

Introduction

The work product characteristics listed in this Annex can be used when reviewing the potential inputs and outputs of an organization's process or practice implementation. The characteristics are provided as guidance for what attributes to look for in a particular sample work product to help assess if the process/practice which created the work product is adequate. Assessor judgement is needed to use this information to ensure that the application domain, business purpose, development methodology, size of the organization, etc. are taken into consideration as well as the characteristics of the work products. This table is not a checklist of what every organization must have, but rather it is a starting point for considering whether the work products are contributing to the intended purpose of the process.

The fields in the work product characteristics table contain the following information.

Work product identifier #

An identifier number for the work product which is used to reference the work product from the process management indicator table, and the process and practice mapping tables.

Work product type:

Provides a typical name associated with the work product characteristics. This name is provided as an identifier of the type of work product the practice or process might produce. Organizations may call these artefacts by different names. The name of the work product in the organization is not significant. Similarly, organizations may have several equivalent work products which contain the characteristics defined in one work product type. The formats for the work products can vary. It is up to the assessor and the organizational unit co-ordinator to map the actual work products produced in the organization to this idealized table.

Work product characteristics:

Provides the potential attributes associated with the work product types that the assessor should probe for in the samples provided by the organizational unit.

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
1)	Software Development Methodology	– Identification of the approach / method used to develop software – Identification of the life cycle model (waterfall, spiral, serial build, etc.) used to develop software – Provides a high level description of the process, activities, and controls
2)	Software Development Life Cycle Model	– High level description of activities performed at each life cycle phase – Sequencing of the life cycle phases – Identification of critical life cycle phase dependencies – Identification of required inputs, outputs to each life cycle phase – Identification of the key decision points (milestones) model – Identification of the quality control points in the model
3)	Process Description	– A detailed description of the process which includes: – purpose of the process – task and activities to be performed and ordering of tasks – critical dependencies between task activities – expected time required to execute task – input/ outputs work products – Identifies process entry and exit criteria – Identifies internal and external interfaces to the process – Identifies process measures – Identifies quality expectations – Identifies functional roles and responsibilities
4)	Job Procedures, Practices	– Each tasks to be performed uniquely identified – Each task sequenced by execution order – Coverage of support information (i.e., commands and parameter settings, etc.) when required for operations – Establishes rules by which staff is expected to operate
5)	Schedule	– Identifies the tasks to be performed – Identifies the start and completion date for required tasks – Allows for the identification of critical tasks and task dependencies – Identifies task completion status, vs. planned date – Has a mapping to scheduled resource data
6)	Work Breakdown Structure	– Defines tasks to be performed – Documents ownership for tasks – Documents critical dependencies between tasks – Documents inputs and output work products – Documents the critical dependencies between defined work products
7)	Work Product	– Defines the attributes associated with an artefact from a process execution: – key elements to be represented in the work product – expected form, style – expected media (paper, electronic) and storage attributes defined

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
8)	Interface	– Defines relationships between two products, process or process tasks – Defines criteria and format for what is common to both – Defines criteria critical timing dependencies or sequence ordering
9)	Standards	– Identification of who/what they apply to – Each requirement unique – Each requirement tagged with an identifier
10)	Coding Standards	*Coverage for software includes (but is not limited to) (as appropriate to the application):* – Data naming conventions – Defines required languages, compilers, data base management systems, etc. – format of code, structure, comments required – standard data structures, types, classes – best practices – Required usage of tools: data dictionaries, associated CASE tools – Compatibility requirement for existing software and / or hardware – Security considerations – Performance considerations – Standard error messages, codes – Interface standards: – human man-machine interfaces – external system interfaces – peripheral equipment, hardware – Storage and retrieval of source code and object modules – Quality and reliability standards
11)	Estimates	– Coverage (as appropriate to the application) for things like: – size – effort – cost – schedule – resources – Estimates are realistic and achievable – in line with resources allocated – in line with historical records (where they exist) – Source data needed to make estimates was available and complete – Source data was validated
12)	Goals (Business, Quality, Organizational, Training, Performance)	– Identifies the objective to be achieved – Identifies who is expected to achieve the goal – Identifies any incremental supporting goals – Identifies any conditions/ constraints – Identifies the time frame for achievement – Are reasonable and achievable within the resources allocated – Are current, established for current project, organization – Used to monitor progress – Are optimized to support known performance criteria, plans

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
13)	Vision	– Provides information on the overall strategy for the organizational unit, organization, or business – Is authorized at the highest level – Defines the main objectives to be achieved
14)	Policies	– Authorized – Available to all personnel impacted by the policy – Establishes practices / rules to be adhered to
15)	Personnel policies	– Defines career opportunities for individuals in the organization – Defines team building strategy – Defines reward and recognition – Covers performance appraisal
16)	Plan (General attributes applies to all plans) (i.e., Business, Organization, Project, Quality, Review, Test)	(as appropriate to the application and purpose): – Identification of the plan owner – Includes the objective of what is to be accomplished – Includes assumptions made – Includes constraints – Includes risks – Includes tasks to be accomplished – Method/approach to accomplish plan – Identifies task ownership – Includes schedules, milestones and target dates – Includes critical dependencies – Identifies quality criteria – Identifies required work products – Includes resources to accomplish plan objectives – time – staff – materials/equipment – budget – Includes contingency plan for non-completed tasks
17)	Project Plans + (16)[1]	– Defines – work products to be developed – life cycle model and methodology to be used – customer requirements – tasks to be accomplished – task ownership – project resources – schedules, milestones and target dates – quality criteria – Identifies: – critical dependencies – required work products – project risks and risk mitigation plan – Contingency actions for non-completed tasks

[1] The symbol + { n } is intended to imply that a generic work product description has been created to contain the majority of attributes common to many work products. These work products should also be used as an addition to the specific attributes mentioned for this particular work product.

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
18)	Process Performance Data	– Data comparing process performance against expected levels: – defined input output work products available – meeting minutes – change records – task completion criteria met – quality criteria met – resource allocation and tracking
19)	Meeting Minutes	– Documents meetings held – Defines: – purpose of meeting – attendees – date, place held – what was accomplished – any open issues – next action
20)	Progress Status record / report	– Record of the status of a plan(s) (actual against planned) such as: – status of actual tasks against planned tasks – status of actual results against established objectives/goals – status of actual resource allocation against planned resources – status actual cost against budget estimates – status of actual time against planned schedule – status of actual quality against planned quality – Record of any deviations from planned activities and reason why
21)	Analysis Results	– What was analyzed – Who did the analysis – The analysis criteria used: – selection criteria or prioritization scheme used – decision criteria – quality criteria – Records the results – what was decided / selected – reason for the selection – assumptions made – Potential risks
22)	Risk Analysis Record / Report	– Identifies the risks analyzed – Records the results of the analysis – potential ways to mitigate the risk – assumptions made – constraints
23)	Risk Management Strategy / Plan + (59)	– Project risks identified and prioritized – Mechanism to track the risk – Threshold criteria to identify when corrective action required – Proposed ways to mitigate risks: – work around – corrective actions activities / tasks – monitoring criteria – mechanisms to measure risk

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
24)	Quality Statement / Policy	– Statement is official, approved – States commitment to quality principles – Identifies who is expected to follow policy
25)	Quality Strategy / Plan	– Objectives / goal for quality – Defines the activities tasks required to ensure quality – References related work products – Method of assessment / assuring quality – References any regulatory requirements, standards, customer requirements – Identifies the expected quality criteria – Specifies the monitoring time fame and quality checkpoints for the defined life cycle and associated activities planned – Target time-frame to achieve desired quality – Method to achieved goals – tasks to be performed – ownership for tasks – resource commitments – Identifies the quality criteria for work products and process tasks – Specifies the threshold/tolerance level allowed prior to requiring corrective actions – Defines quality measurements and benchmark data – Defines the quality data collection mechanism and timing of the collection – Specifies mechanism to feed collected quality data back into process impacted by poor quality – Approved by the quality responsible organization/function
26)	Improvement Opportunity	– Identifies what the problem is – Identifies what the cause of a problem is – Suggest what could be done to fix the problem – Identifies the value (expected benefit) in performing the improvement – Identifies the penalty for not making the improvement
27)	Quality Criteria	*Defines expectations for quality:* – Establishes what is an adequate work product (required elements, completeness expected, accuracy, etc.) – Identifies what constitutes the completeness of the defined tasks – Establishes life cycle transition criteria and the entry and exit requirements for each process and / or activity defined – Establishes expected performance attributes – Establishes product reliability attributes
28)	Quality records	– Defines what information to keep – Defines what tasks/activities/process produce the information – Defines when the data was collected – Defines source of any associated data – Identifies the associated quality criteria – Identifies any associated measurements using the information – Identifies any requirements adherence to create the record, or satisfied by the record

Work product characteristics table

WP Id.#	Work product type	Work product characteristics

29)	Assessment / Audit Records	– States the purpose of assessment
		– Method used for assessment
		– Requirements used for the assessment
		– Assumptions and limitations
		– Identifies the context and scope information required:
		  – date of assessment
		  – organizational unit assessed
		  – sponsor information
		  – assessment team
		  – attendees
		  – scope / coverage
		  – assessee information
		  – assessment Instrument (checklist, tool) used
		– Records the result
		  – identifies the required corrective actions
		  – improvement opportunities

30)	Review Strategy / Plan	– Defines: – what to be reviewed – roles and responsibilities of reviewers – criteria for review (checklists, requirements, standards) – expected preparation time – schedule for reviews – Identification of: – procedures for conducting review – review inputs and outputs – expertise expected at each review – review records to keep – review measurements to keep – resources, tools allocated to the review
31)	Review Records	– Provides the context information about the review – what was reviewed – lists reviewers who attended – status of the review – Provides information about the coverage of the review – checklists – review criteria – requirements – compliance to standards – Records information about the readiness for the review – preparation time spent for the review – time spent in the review – reviewers , roles and expertise – Identifies the required corrective actions – risk identification – prioritized list of deviations and problems discovered – the actions, tasks to be performed to fix the problem – ownership for corrective action – status and target closure dates for identified problems

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
32)	Reuse Plan	– Defines the policy about what items to be reused – Defines standards for construction of reusable objects – defines the attributes of reusable components – quality/reliability expectations – standard naming conventions – Defines the reuse repository (library, CASE tool, file, data base, etc.) – Identifies reusable components – directory of component – description of components – applicability of there use – method to retrieve and use them – restrictions for modifications and usage – Method for using reusable components – Establishes goal for reusable components
33)	Reuse Strategy	– Identify the goals for reuse are stated – Identify the commitment for creating reusable components – Determine which product lines and type of artefacts should be supported with reuse – Identify system and software components which can be reused within the organization – Identify the reuse repository and tools
34)	Reusable Object	– Developed to be: – highly reliable – generically defined (generic names, structures, etc.) – interfaces (inputs and outputs) clear – data encapsulated – Modification controlled – Modifications are downward compatible – Specification for usage defined – Specification for tailoring defined
35)	Reuse Repository	– Repository for reusable components (library, file, data base, tool) – Storage and retrieval capabilities – Ability to browse content – Listing of contents with description of reusable attributes – Ability to identify associated system information – type of object maintained – supported software / applications – associated hardware configuration information – required parameter information
36)	Measures (general applies to all specific measures)	– Available to those with a need to know – Understood by those expected to use them – Provide value to the organization/project – Non-interruptive to the work flow – Appropriate to the process, life cycle model, organization – Are accurate – source data is validated – results are validated to ensure accuracy – Have appropriate analysis and commentary to allow meaningful interpretation by users

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
37)	Project Measures + (36)	*Coverage for key elements in the project plan such as:* – Monitors key processes and critical tasks provides status information to the project on: – project performance against established plan – resource utilization against established plan – time schedule against established plan – process quality against quality expectations and / or criteria – product quality against quality expectations and / or criteria – highlight software performance problems, trends – References any goals established
38)	Process Measures + (36)	– Measures about the process' performance: – ability to produce sufficient work products – adherence to the process – time it takes to perform process – defects related to the process – Measures the impact of process change – Measures the efficiency of the process
39)	Quality Measures + (36)	– Measures quality attributes of the work products defined – product is adequate to do the job intended – product is defect free – product is usable – product is complete – product accurate – product's is reliable – Measures the results of project activities – tasks are performed on schedule – product's development is within the resource commitments allocated – Measures quality attributes of the "end customer" product quality and reliability
40)	Risk Measures + (36)	– Identifies the probability of risk occurring – Establishes measures for each risk defined – Measure the change in the risk state

41)	Field Measures + (36)	– Measures attributes of the performance of system's operation at field locations, such as:
		  – field defects
		  – performance against defined service level measures
		  – system ability to meet defined customer requirements
		  – support time required
		  – user complaints (may be third party users)
		  – customers requests for help
		  – performance trends
		  – problem reports
		  – enhancements requested

Work product characteristics table

WP Id.#	Work product type	Work product characteristics

42)	Service Level Measures + (36)	– Real time measures taking while a system is operational, it measures the system's performance or expected service level
		– Identifies things like:
		  – capacity
		  – throughput
		  – operational performance
		  – operational service
		  – service outage time
		  – up time
		  – job run time

43)	Benchmarking Data + (36)	– Identifies key process / product / market need information to be benchmarked – Measurement reflects comparison of the current performance against some well defined criteria or historical information (or benchmark)
44)	Product Needs Assessment	*Coverage for key elements (as appropriate to the application):* – Definition of the need: – reason product is needed – features and functions desired – requirements to be satisfied – Constraints: – cost limitations – date / schedule requirements – specific support software required – interfaces requirements – associated equipment or hardware required – regulatory standards and/or requirements – operational impacts – Business case: – expected benefit – expected cost (including projected installation, conversion and/or maintenance) vs. profit expectations – market window, target delivery dates
45)	Acquisition Strategy / Plan	– Identifies what needs to be acquired – Establishes the approach for acquiring the product or service – Established the evaluation criteria – Identifies any constraints / risk

46)	Market Analysis Record / Report	– Contains information about:
		  – what was analyzed
		  – the selection criteria & prioritization scheme used
		  – the analysis criteria used
		– Records the results which identify the:
		  – market opportunities
		  – "market window"
		  – business drivers
		  – cost / benefit
		  – potential customers and their profiles information
		  – any assumptions made
		  – alternate solutions considered and / or rejected
		  – risks and/or constraints (regulatory issues)
		– Defines the product offering and target release

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
47)	Request for Proposal (RFP) (Requester)	– Reference to the requirements specifications – Identifies desired characteristics, such as: – system architecture, configuration requirements or the requirements for service (consultants, maintenance, etc.) – quality criteria or requirements – project schedule requirements – expected delivery / service dates – cost / price expectations – regulatory standards / requirements – Identifies submission constraints: – date for resubmitted of the response – requirements with regard to the format of response
48)	Supplier Proposal Response (Response to RFPs)	– Defines the suppliers proposed solution – Defines the suppliers proposed delivery schedule – Identifies the coverage identification of initial proposal – identifies the requirements that would be satisfied – identifies the requirements that could not be satisfied, and provides a justification of variants – Defines the estimated price of proposed development, product, or service
49)	*Subcontractor* or *Supplier* History records	– List of potential subcontractor/suppliers – Qualification information – Identification of their qualifications – Past history information when it exists
50)	Commitments / Agreements	– Signed off by all parties involved in the commitment / agreement – Establishes what the commitment is for – Establishes the resources required to full fill the commitment, such as: – time – people – budget – equipment – facilities
51)	Contract (product or service)	– Signed – Defines what is to be purchased/delivered – Identifies time frame for delivery or contracted service dates – Identifies monetary considerations – Identifies any warranty information – Identifies any customer service requirements – References to any performance expectations constraints – References to any quality expectation / constraints – As appropriate to the contract the following are considered: – references to any acceptance criteria – references to any special customer needs (i.e., confidentiality requirements, security, hardware, etc.) – references to any problem resolution procedures – identifies any interfaces to independent agents and subcontractors

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
52)	Requirement Specification (Internal or External) (Product, Service, Customer, System, Software, Documentation)	– Each requirement is identified – Each requirement is unique – Each requirement is verifiable or can be assessed – Consideration is given to the following (as appropriate to the product or service and type of requirement) *Products / Application Requirements* – identify any required feature and functional characteristics – identify any necessary performance considerations/constraints – identify any necessary external interface considerations/constraints – identify any necessary internal interfaces considerations/constraints – identify any required system characteristics/constraints
52)	(cont'd)	– identify any security considerations/constraints – identify any environmental considerations/constraints – identify any operational considerations/constraints – identify any associated documentation considerations/constraints – identify any installation considerations/constraints – identify any support considerations/constraints – identify any design constraints – identify any reliability or quality expectations *Service Requirements* – identify any performance expectations – identify any time schedule / constraints – identify any tasks to be performed – identify any responsibilities – identify the method of communication, project reporting expected – identify any quality expectations / controls *Document Requirements* – purpose / objectives defined – proposed contents (coverage) defined – intended Audience defined – identification of supported software release, system information – identification of associated software requirements and designs satisfied by document – identification of style, format, media standards expected – definition of the intended distribution requirement

Work product characteristics table

WP Id.#	Work product type	Work product characteristics

53)	System Design / Architecture	– Provides an overview of all system design
		– Describes the interrelationship between system components
		– Describes the relationship between the system components and the software
		– Specifies the design for each required system component consideration is given to things like:
		  – memory/capacity requirements
		  – hardware interfaces requirements
		  – user interfaces requirements
		  – external system interface requirements
		  – performance requirements
		  – commands structures
		  – security / data protection characteristics
		  – system parameter settings
		– Reusable components
		– Mapping of requirements to system components

54)	High Level Software Design	– Describes the overall software structure – Identifies the required software components – Identifies the relationship between software components – Consideration is given to: – any required software performance characteristics – any required software interfaces – any required security characteristics required – any database design requirements – any required error handling & recovery attributes
55)	Low Level Software Design	– Provides detailed design (could be represented as a prototype, flow chart, entity relationship diagram, pseudo code, etc.) – Provides format of input/output data – Provides specification of data storage needs – Establishes required data naming conventions – Defines the format of required data structures – Defines the data fields and purpose of each required data element – Provides the specifications of the program structure

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
56)	Software Units (Code)	– Follows established coding standards (as appropriate to the language and application): – commented – structured or optimized – meaningful naming conventions – parameter information identified – error codes defined – error messages descriptive and meaningful – formatting - indented, levels – Follows data definition standards (as appropriate to the language and application): – variables defined – data types defined – classes and inheritance structures defined – objects defined – Entity relationships defined – Data base layouts are defined – File structures and blocking are defined – Data structures are efficient – Algorithms defined are efficient – Functional interfaces defined – Best practices for language used defined
57)	Build Lists	– Identification of aggregates of the software application system – Identification of required system components (parameter settings, macro libraries, data bases, job control languages, etc.) – Necessary sequence ordering identified for compiling the software release – Input and output source libraries identified
58)	Traceability Record / Mapping	– Identifies requirements to be traced – Identifies a mapping of requirement to life cycle work products – Provides the linkage of requirements to work product decomposition (i.e., requirement->design->code->test->Deliverables, etc.) – Provides forward and backwards mapping of requirements to associated work products throughout all phases of the life cycle Note: this may be included as a function of another defined work product (example: A CASE tool for design decomposition may have a mapping ability as part of it's features)

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
59)	Test Strategy / Plan (all test plans)	– Identification of test purpose – Identification of the responsible test plan owner – Identifies the approach to performing the test – Identification of components to be tested – Identify aggregates and sequence for testing – Identification of required system configuration (software, hardware, interface components) – Identification of the associated development owner for components to be tested – Identification of associated test scripts/test cases – Sequence ordering of how testing will be executed – Identification of *requirements* which will be validated by tests (i.e., customer requirement, regulatory requirements and system requirements – Identification of the problem reporting mechanism – Identification of the test tools and resources required (test channels, analyzers, test emulators, etc.) – Identification of the test schedule – Identification of the test completion criteria – Official source libraries and versions of software defined
60)	Test Script	– Defines what is being tested – Defines the required system configuration for the test – Identifies all required software components – Identifies special initializations, parameter setting, etc. – Identifies the input date required – Sequences the ordering of the test cases – Defines the expected test results – Identifies what requirements were met by performing the test
61)	Test Case	– Provides executable set of test instructions – Purpose defined – Mapped to test scripts, requirements
62)	Test Results	– Records results of testing – Identifies what components were tested – Identifies date test was executed – Status at completion of test (actual test results compared to predicted results in test plan(s)) – Record of test configuration at time of test – Record of trouble reports generated from testing
63)	Unit Test Strategy / Plan + (59)	– Identifies strategy for verifying unit functionality (i.e., a program, a block, a module, a routine) against the requirements and design – Specifies how basic program requirements will be verified
64)	Software Test Plan + (59)	– Identifies strategy for verifying software features and/or functions operate as defined in the requirements
65)	Integration Test Strategy / Plan + (59)	– Purpose of integration defined: – validation of a subset of the system (all programs required to make a sub-system work, a feature work, etc.) – validation of the integration of software to other system components (hardware, support equipment, interfaced system)

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
66)	System Test Plan + (59)	– Identifies strategy for verifying the integration of system components as defined in the system architecture specification – Provides test coverage for all components of the system: – software – hardware – external interfaces – customer documentation – installation activities – initialization – conversion programs
67)	Regression Test Strategy / Plan + (59)	– Plan for validating that existing systems / features-functions have not been impacted by a change – Plan for validating that change has not impacted working components of the system (interfaces, operations, etc.) – Plan for validating that change is compatible with existing system requirements (downward compatible) – Identification of the requirements for system component *not* changed – Identification of what system components are to be regression tested (i.e., features, functions, interfaces, fixes) – Identification of the changes made – Identification of the regression test cases to be executed – Conditions for execution of regression testing
68)	Acceptance Test Strategy / Plan + (59)	– Identified activities to be performed to test "deliverable" end customer product – Identifies who has responsibility for performance of acceptance test activities (supplier or customer) – Identifies the system configuration requirements for site – Identifies the installation requirements for site – Provides a plan for validating the "delivered" software – Identifies how to validate installation activities at customers site were performed correctly – Identifies how to validate the deliverables satisfied the customer requirements – Identifies associated test scripts/test cases – Identifies actions to be take upon acceptance of product
69)	Release Strategy / Plan + (16)	– Identifies the functionality to be included in each release – Identifies the associated components required (i.e., hardware, software, documentation etc.) – Mapping of the customer requests, requirements satisfied to particular releases of the product
70)	Release Package	– Includes the software – Includes and associated release elements such as: – system software components – required hardware – associated customer documentation – parameter definitions defined – command language defined – installation instructions – release letter

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
71)	Release (Notes) Information	*Coverage for key elements (as appropriate to the application):* – Description of what is new or changed (including features removed) – System information and requirements – Identification of conversion programs and instructions – Identification of the component list (version identification included) – software modules, libraries, etc. – associated documentation list – associated hardware requirements – New / changed parameter information and/or commands – Backup and recovery information – List of open know problems, faults, warning information, etc. – Identification of verification and diagnostic procedures – Technical support information
72)	Integrated Software	– All components specified on a software build list for the aggregate is included – Fully configured aggregate of the software components: – parameters defined – commands defined – data loaded or converted
73)	System	– All components of the product release are included – Any required hardware – Integrated software – Customer documentation – Fully configured set of the "system components": – parameters defined – commands defined – data loaded or converted
74)	Installation Strategy Plan + (16)	– Identifies product deployment objectives – Identifies schedules for deployment activities – Identifies schedule constraints – Identifies impacted site locations – Identifies site environment configuration – Identification of the required components for the installation with appropriate version information (consideration given to at least the following): – released software – required maintenance fixes – support software required (conversion programs, validation routines, associated system interfaces, data base management system) – required customer documentation – installation instructions – identification of required hardware and peripheral equipment – Identification of supporting information or materials required: – parameter information – operation and maintenance information – pre-conversion information, materials or installed equipment

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
74)	(cont'd)	– Type of installation (new vs. conversion of existing system, maintenance) – Identification of backup and recovery procedures – Identification of customer contacts and technical support personnel – Identification of *go / no-go* decision criteria– Identification of verification process: – of required tasks to prepare deliverables required – of components required at site – of installation procedures – of pre- installation construction or conversion activities – of system integration, release builds, etc. – Identification of customer acceptance requirements

75)	Installation Guide	*Coverage for key elements (as appropriate to the application):*
		– Tasks for loading/installing product sequentially order by execution requirements
		  – downloading of software from delivery files
		  – up-loading to appropriate software to files, folders, libraries, etc.
		  – partial or upgrade installation instructions, where applicable
		  – initialization procedures
		  – conversion procedures
		  – customization / configuration procedures
		  – verification procedures
		  – bring-up procedures
		  – operations instructions

75)	(cont'd)	– Installation requirements identified: – associated hardware, software, customer documentation – conversion programs and instructions – initialization programs, system generation information – components and descriptions – minimum configuration of hardware/software required – backup / recovery instructions – validation programs – configuration parameters (e.g. size requirements, memory, etc.) – Customer / technical support contacts
76)	Packaging Record	– Content information of what is shipped or delivered electronically – Special handling instructions
77)	Distribution List	– List of current list of receivers and their delivery address – Identification of media expected for delivery (manual, CD-ROM email, etc.)
78)	Delivery Instructions	*Coverage for key elements (as appropriate to the application):* – Sequential ordering of tasks to be performed – Applicable releases identified – Identification of all delivered components with version information – Identification of any necessary backup and recovery procedures

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
79)	Delivery Record	– Record of items shipped/delivered electronically to customer – Identification of who it was sent to – Identification of address where delivered – Identification of the date delivered
80)	Handling and Storage Guide	– Defines the tasks to perform in handling and storing products – Provides a description of how to store the product including: – storage environment required – the protection media to use – packing materials required – what items need to be stored – Provides retrieval instructions
81)	Acceptance Record	– Record of the receipt of the delivery – Identification of the date received – Identification of the delivered components – Records the verification of any customer acceptance criteria defined – Signed by receiving customer
82)	Customer Support Procedures	*Coverage for key elements (as appropriate to the product or contract):* – Tasks to follow in providing support defined – Defines the availability and coverage the support provided: – hot-line # – hours of availability – appropriate expertise – cost – Defines a schema for classification of customer request and /or problems: – definition of request type – definition of priority/severity – definition of response time expectations, by type and severity – Standards for what information to retain from a customer, such as: – company and location – contact information details – description of the request – reference to supporting information sent (dumps, files) – customer system site configuration information (product, release, version, last update) – impacted system(s)

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
82)	(cont'd)	– impact to operations of existing systems – criticality of the request – expected customer response/closure requirements – Definition of customer escalation procedures – Identification of *customer support tools* available and procedures for using them, such as: – mechanism used to record customer requests – status reports – systems available to reproduce problems – ability to reproduce customers software environment – ability to reproduce problems – rest emulators – rest scripts – dial-in ports – dump analysis tools
83)	Customer Request record (internal or external)	– Identifies request purpose, such as: – new development – enhancement – internal customer – operations – documentation – informational – Identifies request status information, such as: – date opened – current status – date assigned and responsible owner – date verified – date closed – Identifies priority/severity of the request – Identifies customer information, such as: – company/person initiating the request – contact information and details – system site configuration information – impacted system(s) – impact to operations of existing systems – criticality of the request – expected customer response/closure requirements – Identifies needed requirements/standards – Identifies information sent with request (i.e., RFPs, dumps, etc.)

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
84)	Problem Report record	– Identifies the name of submitted and associated contact details – Identifies system configuration information (such as: release versions, system software, hardware configuration, etc.) – Identifies the group/person(s) responsible for providing a fix – Includes a description of the problem – Identifies any associated support information (dumps, files, etc.) – Identifies the severity of the problem (critical, major, minor..) – Identifies the status of the reported problem – Identifies the components of the product affected – Identifies the applicable software product release and version information – Identifies the date "opened" – Identifies the target release(s) problem will be fixed in – Identifies the expected closure date – Identifies any associated problem reports, customer requests, duplicate problems, associated fixes – Identifies any closure criteria
85)	Customer Satisfaction Survey	– Identification of customer and customer information – Date requested – Target date for responses – Identification of associated software and hardware configuration – Ability to record feedback
86)	Customer Satisfaction Data	– Determines levels of customer satisfaction with software products and services – Mechanism to collect data on customer satisfaction: – results of field performance data – results of customer satisfaction survey – interview notes – meeting minutes from customer meetings
87)	Communications Mechanism	*A way to distribute information:* – Clear description of what is being communicated – Ability to specify date information sent – Ability to distribute to all impacted – Identification of the impact: (software, development, customer, organization, etc.) – Provides a clear identification as to who/what the message applies – Mechanism for recipient to respond when required (return information) – The distribution media used is accessible to all with a need to know – The distribution list is current and includes all with a need to know – Ability to specify target return date information
88)	Training strategy/plan + (16)	– Defines current staff capabilities – Defines the skills required – Outlines course available to achieve training goal

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
89)	Training records	Record of employee's training – Identifies employee's name – Identifies any courses taken (date, hours, course title) – Identifies current skills/capabilities/experience level, lists: – formal education – in-house training – mentoring – Identifies future training needs – Identifies current status of training requests
90)	Training Material	– Synchronized to current supported versions of the software – Updated and available for new releases – Coverage of system, application, operations, maintenance as appropriate to the application – Courses listings and availability
91)	Configuration Management (CM) plan	– Defines or references the procedures to control changes to configuration items – Defines measurements used to determine the status of the CM activities – Defines CM audit criteria – Approved by the CM function – Identifies configuration library tools or mechanism – Specifies the location and access mechanisms for the CM library. – Archival and retrieval mechanism specified
92)	Configuration Management (File, Library, System)	– Version control – Check in/out capability – Can recreate any release or test configuration – Maintain configuration item descriptions – Ability to report configuration status – Changes to configuration items are tracked to change/user requests
93)	Configuration Item	– Item which is maintained under configuration control (software, documents, work products) – Version identification is maintained – Description of the item is available including things like: – type of item – associated configuration management library, file, system – responsible owner – date when place under configuration control – status information (i.e., development, baselined, released) – relationship to lower level configured items – identification of the change control records – identification of change history
94)	Change Request	– Identifies purpose of change – Identifies request status (new, accepted, rejected) – Identifies requester contact information – Impacted system(s) – Impact to operations of existing system(s) defined – Impact to associated documentation defined – Criticality of the request, date needed by

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
95)	Change Control Record	– Used as a mechanism to control change to baselined products/ products in official project release libraries – Record of the change requested and made to a baselined product (work products, software, customer documentation, etc.) – identification of system, documents impacted with change – identification of change requester – identification of party responsible for the change – identification of status of the change – Linkage to associated customer requests, internal change requests, etc. – Appropriate approvals – Duplicate requests are identified and grouped

96)	Change History	– Historical records of all changes made to an object (document, file, software module, etc.)
		  – description of change
		  – version information about changed object
		  – date of change
		  – change requester information
		  – change control record information

97)	Corrective Actions (logs, plans, minutes)	– Identifies the initial problem – Identifies the ownership for completion of defined action – Defines a solution (series of actions to fix problem) – Identifies the open date and target closure date – Contains a status indicator
98)	Tracking system	– Ability to record customer and process owner information – Ability to record related system configuration information – Ability to record information about problem or action needed – date opened and target closure date – severity/criticality of item – status of any problem or actions needed – information about the problem or action owner – priority of problem resolution – Ability to record proposed resolution or action plan – Ability to provide management status information – Information is available to all with a need to know – Integrated change control system(s)/records
99)	Work-around (temporary solutions)	– Problem identification – Release and system information – Temporary solution, target date for actual fix identified – Description of the solution – limitations, restriction on usage – additional operational requirements – special procedures – applicable releases – Backup/recovery information – Verification procedures – Special installation instructions

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
100)	Product Configuration	– Overview of the system's configuration – Defines each component and their position in the architecture of the system – Defines the key system interfaces – Defines any network considerations – Defines the hardware configuration – Defines any system performance/parameter settings

101)	Database Designs	*Coverage for key elements (as appropriate to the application):*
		– Definition of design characteristics:
		  – database management system used
		  – type of system (relational, hierarchical, object oriented, networked)
		  – format of records, tables, objects
		  – database access mode
		  – associated software (programs, user screen formats, reports)
		  – supported database language
		– Definition of logical and physical views, models:
		  – records (data layouts, fields, tables, structures)
		  – field names and definitions
		  – data definitions, classes, structure, etc.
		  – entity / relationships
		  – classes, inheritance scheme
		– Definition of user views
		  – screen layouts
		  – field access
		  – data access
		  – commands
		  – Input / output interface considerations
		– Database usage information (contents, application systems, usage restrictions, etc.)
		– Identification of constraints:
		  – security considerations
		  – data access considerations
		  – back-up and recovery considerations
		  – system restart considerations
		  – system generations considerations
		  – performance considerations

102)	Back-up / Recovery Records	– Date of back-up – Listing of what was backed-up with associated versions – Listing of where it was backed-up to – Identification of associated system attributes and configuration at time of back-up – Identification of associated recovery procedures

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
103)	Recovery plan	– Identifies what is to be recovered – Procedures / methods to perform the recover – Schedule for recovery – time required for the recovery – critical dependencies – Resources required for the recovery – list of backups maintained – staff responsible for recovery and roles assigned – special materials required – required work products – required equipment – required documentation – Locations and storage of backups – Contact information on who to notify about the recovery – Verification procedures – Cost estimation for recovery
104)	Development Environment	– Floor plan – Environmental safety considerations – Regulatory requirements – Contractual requirements – Security considerations – Facility configuration – Special environmental requirements (e.g. air conditioning, raised floor, power, etc.) – Individual workspace needs defined – Workstations requirements – Supporting software – Tools – Communication equipment – Disaster recovery plan
105)	*Customer Documentation* Test Plan	– Meets customer requirements – Approved by customer – Identifies deliverable documentation – Define or reference templates – Document verification addressed

Work product characteristics table

WP Id.#	Work product type	Work product characteristics
106)	Customer Documentation	– Coverage for key elements (as appropriate to the application)*:* *External system documents* – system overview, architecture, design guide – feature guide (functional descriptions of system components) – diagnostic guide: error messages and codes – operating commands reference guide – installation, operations and maintenance guide – technical support guide *Internal customer documents* – requirements – designs – test plans – plans – records – Documentation kept synchronized with latest associated software release: – available with delivery of a new or changed version of the software – updated with maintenance releases (as appropriate to change request resolution) – Ordering procedures – Current site distribution and maintenance list maintained
107)	Installation records	– Record of what was installed – Release and system configuration information recorded – Special site specific information recorded – Identification of any acceptance testing performed – Installation performance information captured – ability to bring up system after installation conversion – number of faults found after the installation or conversion – time to install – Customer approval signatures
108)	System Component	– Hardware components – Software components – Customer documentation – Training materials
109)	Personnel Records	– Relevant information about personnel including: – Name, address, date of birth, marital status – Grade, pay, appraisal history – Disciplinary history

Annex E (informative)

Assessment instrument concepts

E.1 Assessment instrument indicators

E.1.1 Introduction

The practices in the process model in part 2 of this International Standard are the criteria against which an assessment is performed. These practices represent "good practice", but in order to make them applicable to all software applications and domains, they are defined as abstract, high level concepts without constraining the ways in which they may be implemented. Consequently, practices are subject to wide interpretation which can have an adverse effect on the repeatability and reliability of assessment results.

Assessment indicators represent a set of attributes that might be found in an instantiation of a process and hence can be used to judge adequacy. Indicators are not requirements: they provide a set of detailed discriminators used to assess whether a particular instantiation of a practice meets the intent of the practice in the process model in part 2.

The performance of a process typically produces tangible work products (inputs and outputs associated with the execution of the practices). The indicator set represents a common starting point for assessing these process artefacts and the practices that produced them. Use of the standard set of indicators increases the consistency of assessor judgement and enhances the repeatability of the results.

The output of the assessment, in the form of a process profile, shows the adequacy ratings of the generic practices of the process, but it does not show why a particular practice was assigned a particular rating. Indicators help to identify what is present or missing from a process or work product and provide guidance to the assessor when assigning a rating of adequacy to a practice. The information provides an "indication" of the extent to which a practice supports the purpose of the process. The detailed information captured during the assessment about the presence or absence of specific indicators provides the valuable input into analysis and process improvement planning.

An assessor’s judgement of practice adequacy is always made within the context of the process of the assessment scope and purpose. Organizational business goals, and the size, complexity and criticality of the software, are factors that influence an assessor's judgement. As these factors are unique to every assessment, the standard set of indicators do not include everything an assessor must consider in reaching a judgement. Indicators can be characterized best as "guidance, memory joggers, triggers, discriminators, hints, examples".

In addition, since many organizations employ different techniques to create software, the absence of some indicators in some situations may not be significant. Care should be taken when using the assessment indicators to understand that the set provides a consistent set of probes to help recognize the characteristics of adequacy in a consistent way, not an all inclusive checklist of required elements.

The process model in part 2 of this International Standard provides the flexibility to define extended processes. Consequently, the indicator set may be tailored by adding indicators to support the practices of the extended processes. The standard set of assessment indicators may also be tailored to suit a particular assessment objective, application domain, or risk. Tailoring should be performed in a manner that retains the common basis for result comparison.

Two types of assessment indicators, process indicators and process management indicators, are defined to address the base and generic of practices defined in the process model in part 2.

E.1.2 Process indicators (PI)

E.1.2.1 Overview

Process Indicators (PI) provide guidance to the assessor on how to judge a base practice to determine its existence or adequacy rating. There are two main components associated with process indicators: base practices and work products. To avoid duplication, base practices are not listed in the standard set of indicators in this document other than by reference to the practice number in the process model in part 2. It is anticipated that when an assessment instrument is created, the practices will be extracted directly from the process model in part 2 of this International Standard and used in conjunction with the work products types and their associated characteristics.

The performance of tasks similar to those defined in the base practices provides the first indication that an implemented process includes the defined good practice. This provides some evidence of the existence of base practices. To support the judgement of practice adequacy, associated work product types and characteristics are defined. The characteristics of the work products assist the assessor in understanding what elements to expect in a meaningful instantiation of a work products type.

Figure 1 shows the relationship of process indicators to the organization's process. Work product characteristics are mapped to a sample of the organization's input and output work products. The organization's process activities are reviewed to ensure performance of the base practices.

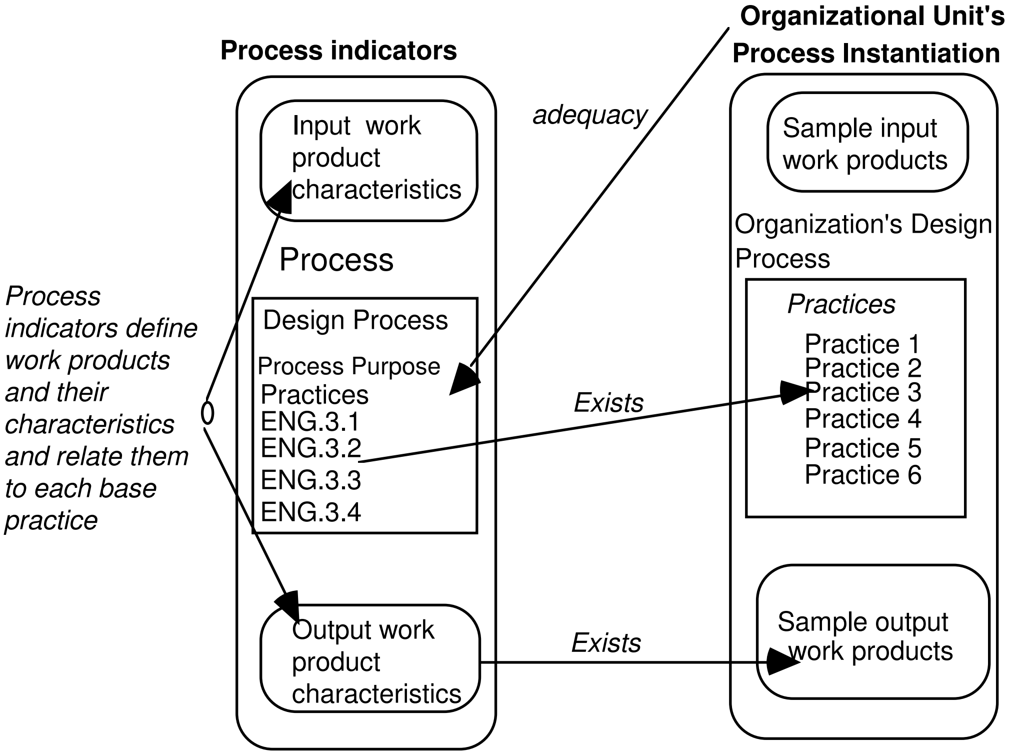

Process indicators
Organizational Unit's Process Instantiation
Input work product characteristics
adequacy
Sample input work products
Organization's Design Process
Process
Process indicators define work products and their characteristics and relate them to each base practice
Design Process
Process Purpose
Practices
ENG.3.1
ENG.3.2
ENG.3.3
ENG.3.4
Practices
Practice 1
Practice 2
Practice 3
Practice 4
Practice 5
Practice 6
Exists
Output work product characteristics
Exists
Sample output work products

Figure 1 - Process indicators usage

E.1.2.2 Work products

Work products are the result of the execution of the practices within an organization. The existence of sample work product types provides further evidence that the defined practice is actually performed in the organizational unit. However, practice existence alone does not provide evidence of a sufficient implementation. To assist the assessor in this evaluation, a standard set of work product types and their associated characteristics is defined. Work products characteristics are used in the judgement of adequacy. This judgement is usually performed simultaneously by using process indicators and process management indicators (see below). Understanding the connectivity between these two sets of indicators is important in the assessment of generic practices for the process under review. Examining work products can also provide evidence for the rating of generic practices. For example, a document may have a version number implying that is under configuration management (see generic practice 2.2.2).

As work product names can vary from organization to organization, the exact names associated with specific work product types are not significant. However, an assessor would expect to find an equivalent representation of the work product that would provide the coverage of the attributes defined in the work product characteristic set. For example, in assessing the process Develop Software Design (ENG.3), an assessor may find four work products: Functional Framework; Signal Specification; Flow Diagram; and Interface Specification. Using the work product characteristics, the assessor may find that these four products collectively contain the characteristics expected of the work product type High Level Software Design.

The existence of work product characteristics tends to indicate a adequate instantiation of the output of a performed practice. This does not imply that not finding all the defined characteristics makes for an inadequate implementation. If some characteristics are missing, the assessor makes a judgement as to the significance in this particular application. If the missing characteristics are significant, the information should be recorded by the assessor.

There may be circumstances when the defined characteristics are not-applicable. For example, when an organization is employing a new type of process technology that is unique to their process. This should be recorded in the assessment record for subsequent analysis and process improvement purposes.

E.1.3 Process management indicators

Process management indicators are associated with each generic practice in capability levels 2-5. Similar to process indicators, they complement the assessor’s ability to recognize the performance of generic practices. They provide assistance in rating generic practice adequacy, and help to identify the ability of the organization to manage a process effectively.

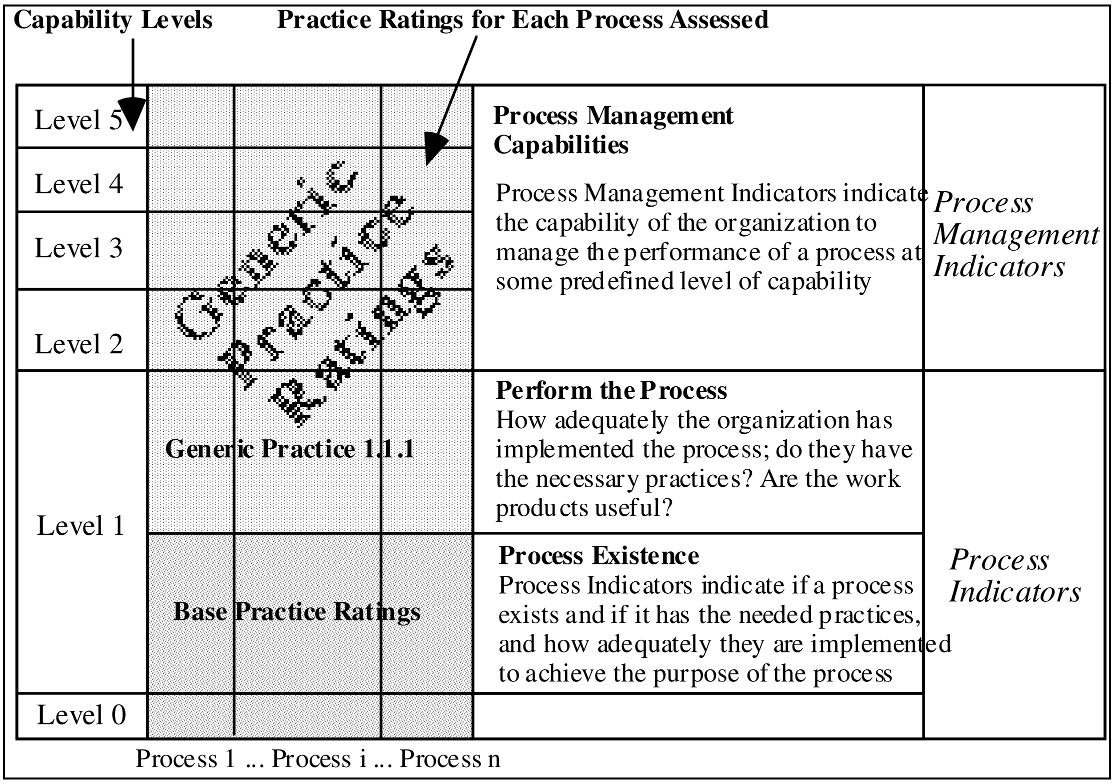
In addition, process management indicators provide a structured way of recording in the assessment record what was found in a particular instantiation of a process. During an assessment, process management indicators are used in conjunction with the process indicators to give the assessor a view of process capability.

Process management indicators are used to probe the attributes of a process that affect the way that the organization manages the process. They assist the assessor in judging the adequacy of the implemented generic practices. Figure 2 shows the relationship between the process management indicators and process indicators to the process model defined in part 2 of this International Standard.

The assessment examines the process of the organization against this process model. The architecture of the process model uses the concept of generic practices which may be applied to any process, and groups the generic practices into capability levels such that the attainment of each level adds significantly to the capability to perform that process.

The process management indicators and process indicators provide a framework for assessment, and help to ensure that:

- assessors have the ability to interpret the organization's instantiation of a process consistently against the process model in part 2;
- the data are captured for subsequent analysis to support the sponsor’s needs;
- the information needed for the organizational unit to plan and perform process improvement is captured;
- assessment results are representative, reliable and repeatable.



Process Management Capabilities

Process Management Indicators indicate the capability of the organization to manage the performance of a process at some predefined level of capability

Perform the Process

How adequately the organization has implemented the process; do they have the necessary practices? Are the work products useful?

Process Existence

Process Indicators indicate if a process exists and if it has the needed practices, and how adequately they are implemented to achieve the purpose of the process

Figure 2 Relationship of process management indicators and process indicators to the process model.

E.2 Assessment instrument data handling

An assessment instrument, even paper-based, should provide a repository for information about the adequacy of practices, the existence of assessment indicators, and other types of information (organizational observations, notes about particular judgements, profiles, etc.). Data collection and retrieval mechanisms affect the usability of the assessment instrument.

Although information about the assessment indicators should be recorded, it does not form part of the process profile, and is typically not given back to the sponsor of a capability determination. Although, some sponsors may desire supporting information for analysis in process capability determination, the use of the data captured within an assessment instrument should be the subject of a pre-assessment agreement between all parties involved (the assessor, the sponsor, the organizational unit). The agreement should cover storage, maintenance and use of the detailed records.

The characteristics defined by the indicators provide a detailed record of what was found in the organizational unit. The data captured are significant both for assessor's evaluation and for subsequent analysis and planning for process improvement.

Assessments can generate a large quantity of data. The support provided by automated assessment instruments and tools to handle the data significantly affect the efficiency and effectiveness of the assessment. It is important, therefore, that the assessment instrument is able to provide the right level of support for gathering, processing and storing assessment data.

E.3 The relationship of assessment instruments to ratings

Assessment ratings are assigned for the base practices and generic practices of assessed processes according to the requirements for rating in part 3 of this International Standard. The output is represented as process profiles, containing generic practice ratings and derived capability level ratings, and an assessment record, containing the base practice ratings and supporting information. Actual practice adequacy ratings are determined for each assessed process instance. Generic practice adequacy ratings may be aggregated to form a view of the performance of a process at each of the capability levels. The diagram in figure 3 shows how the information contained in the assessment instrument, indicators and practices, come together to support a rating.

There are many approaches that can be used to gather data. The method and approach will depend on many factors including:

- the size of the organization being assessed;
- the number of organizational units involved in the assessment;
- the level of organizational participation in performing the assessment (collecting the data, demonstrating conformance);
- the maturity of the supplier-sponsor relationship (the level of trust between the organization and sponsor);
- the needs of the sponsor;
- the expertise and ability of the assessor(s);
- the needs of the organization.

The assessment instrument design should be of a scope to cover the intended approach. Guidance on the factors which affect the usability of various types of assessment instrument is included in the informative annexes of this part of the International Standard.

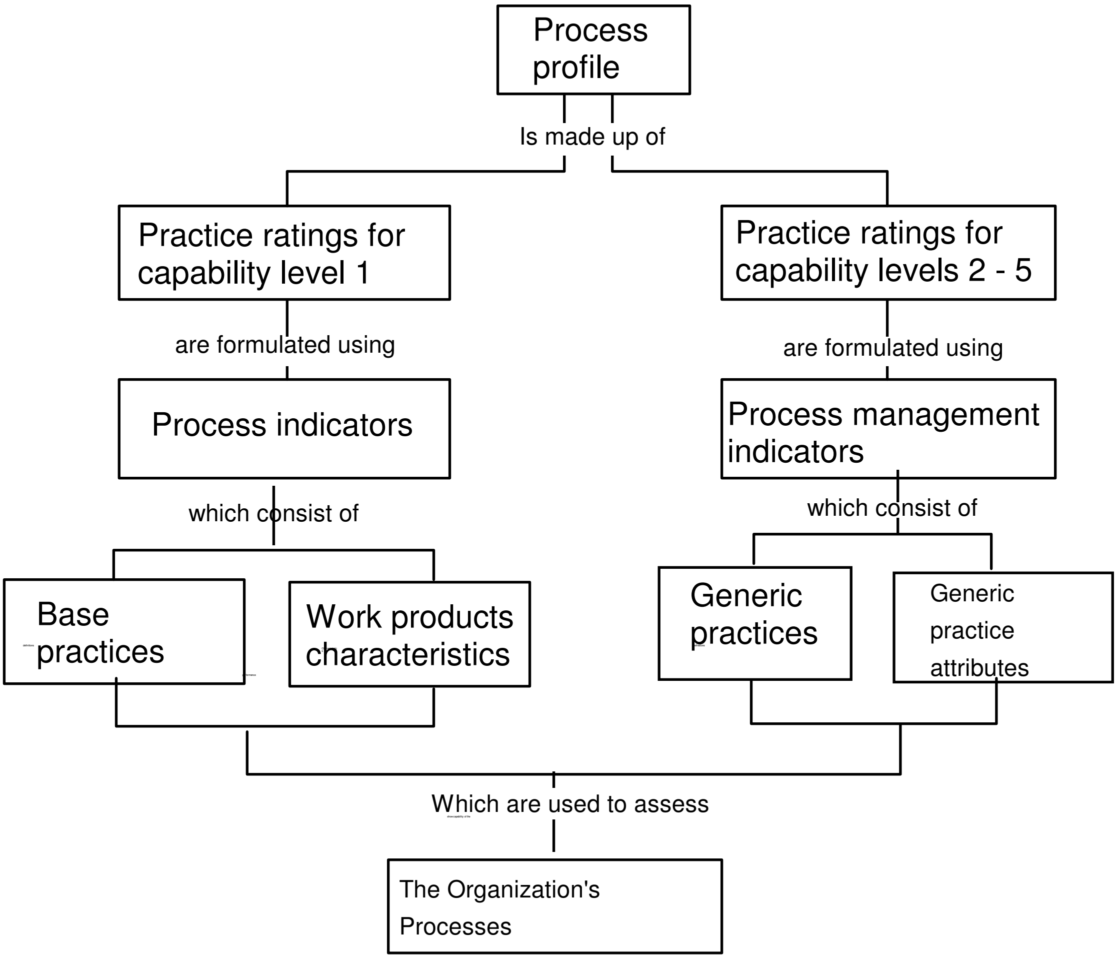

Process profile
Is made up of
Practice ratings for capability level 1
Practice ratings for capability levels 2 - 5
are formulated using
are formulated using
Process indicators
Process management indicators
which consist of
which consist of
Base practices
Work products characteristics
Generic practices
Generic practice attributes
Which are used to assess
The Organization's Processes

Figure 3 – Putting the elements together to determine a rating

Annex F (informative)

Construction, selection and use of an assessment instrument

F.1 Introduction

This section gives guidance on the purchase or construction of an assessment instrument to support the objectives specified in this standard. It specifies the types of assessment instruments and the associated features and functions that impact the design options related to an instrument's effectiveness and usability. This section:

- describes the two basic types of assessment instrument;
- sets out the modes of use and usability for both types of assessment instrument;
- highlights key considerations in the creation or tailoring of an assessment instrument;
- identifies some of the issues to consider in selecting an assessment instrument.

The classification scheme for quality characteristics of software in ISO/IEC 9126:1991 forms a useful overall framework for consideration in the specification, construction, selection and use of an assessment instrument. In addition, Annex G describes and classifies a number of desirable features of assessment instruments which should be considered by users selecting instruments suitable to their own specific needs and assessment contexts and by instrument developers creating assessment instruments targeted at specific methodologies or approaches.

F.2 Basic types of assessment instrument

There are two basic types of assessment instrument, paper-based manual instruments and automated computer-based instruments, which have different characteristics. An understanding of the benefits and limitations of each type helps to ensure that a chosen assessment instrument supports the assessment purpose and scope. The appropriateness of an assessment instrument depends on the planned mode of use and assessment methodology. To ensure optimum performance (effectiveness and efficiency), assessment instruments should be selected or designed to match the assessment approach.

F.2.1 Paper-based assessment instruments

Using a paper-based instrument in an assessment demands careful foresight and planning. A paper-based tool may be adequate for an assessment of limited scope where only a few processes are to be assessed, but may become unwieldy for an assessment of broader scope. Assessing all the base and generic practices for each process, and generally for more than one instance of each process, generates a significant amount of data to record, track and manage.

There are a variety of formats for paper-based assessment instruments which can be used effectively, depending on the approach, style of assessor or methodology. A paper-based assessment instrument may be implemented, for example, as a questionnaire, a checklist, or an assessment recording form. The usability of a paper-based assessment instrument is an important design consideration. Its usability has a strong relationship to the effectiveness of the instrument in a particular assessment mode.

In an assessment, a paper-based assessment instrument is most effectively used:

- for collection of a limited amount of data, such as in a focused assessment of a few process areas;
- in a distributed approach, such as when forms are distributed throughout an organization for self-assessment;
- when the sampled work-products and process data are collected incrementally and reviewed prior to the commencement of on-site assessment activities, such as interviews;
- when sampling an organization to obtain a pre-assessment of the potential level of capability;
- when developing a prototype or trialing a new assessment methodology.

The benefits of using a paper-based instrument include low initial development cost, portability and relative ease of construction.

The limitations associated with paper-based assessment instruments include:

- the inability to support automated scoring and the aggregation of results across multiple instantiations or organizational units;
- the inability to change content dynamically to suit the scope of the assessment or to tailor for organizational characteristics discovered during the assessment;
- the limited ability to select indicators dynamically to suit the individual assessment interview needs;
- the limited ability to store and retrieve assessment results for subsequent use in process improvement or capability determination;
- the inability to generate result profiles or help in the performance of gap analysis;
- the large amounts of paper to be managed; especially when used in large organizations with multiple process instantiations.;
- the extensive configuration management control of the instruments created and the results collected;
- the potential insecurity of the data collected;
- the difficulty of use in organizations whose processes differ from those for which the assessment instrument was designed;
- the dependency on the assessor and the assessment method. Additional assessor training may be required on the concepts associated with the process model defined in part 2 of this International Standard.

F.2.2 Computer-based assessment instruments

A computer based assessment instrument may be implemented as a spreadsheet, a data base application system, an expert system or integrated into a CASE tool application. Computer based assessment instruments may be integrated into the software development life cycle, allowing for more real time improvement opportunities. Well-designed tools can enhance the trust and credibility of self-assessment. With the right tools, process owners can perform a self-assessment and get trusted results. A computer-based assessment instrument has several advantages over a paper-based design, including:

- the ability to be implemented and used in a distributed manner, to collect data incrementally at set milestone check points in the performance of a process or when a number of organizational units are to be assessed incrementally;
- the effectiveness of storing and retrieving assessment results, making the results more usable for process improvement planning or capability determination analysis;
- the ability to assist the assessor with post-assessment analysis of the results such as the analysis of process improvement results against past performance history, or of a supplier profile against an established target profile;
- the ability to build assessment expertise directly into the tool, allowing a less qualified assessor to perform the actual assessment. This can release scarce resources and transfer them to actual process improvement activities rather than assessment activities;
- the ability to perform dynamic scoping and tailoring, allowing for customization to support specific cultural, organizational, sponsor, or assessment team needs;
- the ability to assist the assessor with the processing of the assessment data collected;
- the ability to aggregate and generate results in a variety of formats to suit individual sponsor needs. For example: reports, charts, profiles, lists of practice conformance attributes, etc.;
- the ability to secure the data captured in the assessment to ensure confidentially;
- the ability to process data from multiple process instantiations or across multiple organizational units simultaneously to encourage self-assessment.

Any computer-based assessment instrument design should maximize its usability and support for the given assessment mode. Failure to consider options for maximizing efficiency in collecting and processing the data could render it ineffective. The major limitations associated with an automated assessment tool involve its high initial start-up cost and the time associated with building and maintaining a computer system. Automated assessment instruments are also subject to the limitations of computer software. A poor design or implementation can do the opposite of what is intended.

A table in Annex G lists characteristics associated with assessment instruments and their use. This table provides assistance to sponsors and assessment teams for the selection of appropriate features to support their particular needs. A designer or a purchaser may use these tables as a guide to the benefits and limitations associated with a particular assessment instrument, or mode of use. The quality characteristics defined in Annex G are applicable to the design of a any tool and should be considered.

F.3 Modes of use and usability of types of assessment instrument

An assessment instrument may be used in several modes:

- by an assessor or assessment team to conduct an assessment. The results may be captured by a paper-based instrument, or a lap-top computer;
- by process owners and/or organizational unit representatives during preparation for and prior to an assessment. The results can be captured by the instrument for subsequent processing, or to demonstrate conformance for external validation by a third party assessor, thereby reducing the time and cost associated with an assessment;

  NOTE : This is the most efficient way to collect data prior to conducting interviews given the large number of practices contained in the process model defined in part 2 of this International Standard.
- by organizational unit representatives continuously throughout the software development life cycle, and at defined milestones to measure process adherence, process improvement progress or to gather data to facilitate a future assessment.

  NOTE : This type of distributed approach is most effective when using automated tools integrated into the life cycle such as performance monitoring tools, project management tools or CASE tools.
- after the assessment to retrieve or organize the assessment data to facilitate process improvement planning or analysis for capability determination.

  NOTE : The detailed data captured during the assessment are valuable inputs to an organization and enhances the organization's understanding of the ratings of practices and process.

F.4 Creating an assessment instrument

This section gives guidance on the creation of an assessment instrument using the assessment indicators contained in Annexes A to D, the practices contained in the process model defined in part 2 of this International Standard, and the requirements in clause 4 of this document.

The instrument designer should understand the intended use and assessment methodology that the tool will be expected to support. The instrument may be targeted at experienced assessors for use in a standalone mode, or distributed throughout an organization to enable less experienced assessors (such as the process area representatives) to use the tool in self-assessment mode. An assessment tool enables much of the data to be gathered prior to the assessor's visit, expediting the assessment process and increasing the organizational representatives' ownership of the assessment results. The more sophisticated the assessment tool the less assessor expertise is required.

In creating an assessment instrument, the designer should:

- understand the methodology and approach to be used for the assessment;
- select the tool characteristics based upon the planned implementation, methodology, or sponsor need. Annex G outlines the design options available;
- choose an appropriate type of instrument based on the methodology and intended use of the instrument;
- review the indicator set presented in Annexes A to D in relation to the scope and context of the assessment or the objectives of the instrument supplier. Indicators are selected based upon the scope, or other performance considerations;
- select the process, practices and associated indicators relevant to the scope of coverage (see clause 4). An automated instrument can enable the selection to be performed dynamically (i.e. as the instrument is being used);
- tailor or format the indicators in the selected set. An automated assessment instrument can enable tailoring or scoping of the indicators to be performed dynamically (i.e. during the assessment);
- consider the requirements of the sponsor(s) with respect to the usage mode, business needs for results presentation, interfaces to other products, storage of results, confidentiality, etc.

F.5 Tailoring to suit a particular assessment need

Specific requirements for what may be tailored and what is required when tailoring an assessment instrument are given in clause 4 of this document. The designer or qualified assessor participating in the tailoring of the assessment instrument should also address the following guidelines:

- the assessment instrument should be under configuration management during its creation and maintenance;
- the assessor should ensure that after customization, the requirements in clause 4 continue to be met;
- the requirements of the sponsor(s) should be considered when tailoring an assessment instrument, or example, the method of use, the input medium and the output presentation format.

F.6 Selecting an assessment instrument

Prior to selecting an assessment instrument, the purchaser should review the requirements and recommendations in clause 4 of this document and the requirements for assessment in part 3 of this International Standard. The following issues should be considered:

- the intended mode of use, method of assessment, and the scope of the elements contained in the process model defined in part 2 of this International Standard;
- the purpose of the assessment – process improvement (part 7 of this International Standard), or the determination of supplier process capability (part 8 of this International Standard);
- the direct impacting of any customization on the tool's performance;
- the quality aspects described in Annex F;
- which of the features or functions covered in Annex F are required;
- supplier identified limitations, or usage requirements;
- support for the instrument (training, hot-line, documentation, etc.);
- the use of a supplier's past history or current assessment results as valuable input to a procurement decision;
- the evaluation of the conformance of the tool to the requirements of this International Standard;
- the desirability of placing the tool under configuration management control.

Annex G (informative)

Quality and design attributes

G.1 Introduction

This annex contains a list of desirable features to be considered in building or purchasing an assessment instrument.

Where software-based tools are developed or used to support assessments, ISO/IEC 12119 - 1995 may provide a useful mechanism for demonstrating or verifying their conformance to the requirements in this part of the International Standard.

F.2 lists quality features classified by quality characteristics as defined in ISO/IEC 9126:1991 (Functionality, Reliability, Usability, Efficiency, Maintenability and Portability). Since some features enhances more than one quality characteristic, the feature has been described under the quality characteristic to which it contributes most.

F.3 lists design considerations,benefits and drawbacks of different types of assessment instruments (manual, on-line and expert systems).

F.4 lists desirable instrument attributes associated with different assessment purposes, types, approaches and scopes.

These tables are intended as guidance and can be helpful to designers or purchasers of assessment instruments.

G.2 Quality attributes of assessment instruments

G.2.1 Functionality

Functionality encompasses a set of attributes that bear on the existance of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs.

Features related to the functionality characteristic are described in other parts of this document. Additionally, the following paragraph addresses the security aspects of assessment instruments.

G.2.1.1 Security

The security of the data captured by assessment instruments needs to be considered. A computer-based instrument may need security mechanisms in order to support specific confidentiality requirements imposed by the sponsor or assessed organization. The following security capabilities should be considered:

- preventing unauthorized access by others;
- incorporating an access password on automated tool implementations;
- putting appropriate restrictions and proprietary markings on documents used to collect, and report findings;
- locking files and limiting access to the data collected (automated or manual);
- the method to be used to maintain the confidentiality of any support data captured.

G.2.2 Reliability

Reliability encompasses a set of attributes that bear on the capability of an automated assessment instrument to maintain its level of performance under stated conditions for a stated period of time.

Features that may impact the reliability of an automated instrument are described below.

G.2.2.1 Repeatable results

Any automated features of the tool should be checked to ensure that it gives repeatable results when performing aggregations or data manipulations.

G.2.2.2 Tool reliability

The tool should reliably store data indefinitely without corruption or loss. The tool should not exhibit failures which corrupt data under the expected operational conditions.

G.2.3 Usability

Usability encompasses a set of attributes that bear on the effort needed for use, and on the individual assessment of such use by a stated or implied set of users.

The following features contribute to the usability of an assessment instrument.

G.2.3.1 Instruction and usage support

Desirable features are on-line help, context-sensitive help, and complete and clear user documentation.

G.2.3.2 User-friendly interface

The assessment instrument should be easy to use. In automated tools, friendly interfaces such as windowing, pull-down menus, pop-up tables etc., are desirable.

With all tools, information gathered is useful for assigning adequacy to different practices. Having a consolidated view of all information related to a practice or process may be helpful to the assessor. In cases such as this, the instrument may provide automatic spreading of the information to the related point of use.

Automated tools may also provide automatic prompting for missing information, and provide a way to check coverage of the organization.

Paper based tools should be organized by the intended usage in a particular assessment to support, for example: practices needed for specific interviews; ease of finding the data; ease of distributing the forms to appropriate assessors or organizational unit representatives; and ease of collating results.

G.2.3.3 Availability of training and support

Automated assessment instruments should be accompanied by appropriate training materials and supporting services provided by the creator/supplier.

Paper-based assessment instrument creators should supply appropriate directions on using the paper instruments, collating information from various instances and generating results, using the information contained on the paper forms.

Assessment tool training should be readily available. Training may take many forms, including on-line tutorials, instructor lead courses, self-study. In developing training for an assessment instrument consideration should be given to:

- the type of assessment instrument, its key features, and its use;
- assessment approach(es) supported by the assessment instrument;
- how the instrument meets the requirements specified in clause 4;
- tailoring or customization of the indicators in the tool;
- processing the data captured during an assessment to establish a rating profile / score;
- how to access and use the assessment results.

Users of a computer based assessment instrument my need additional training in:

- installation including the required environment;
- features of the assessment instrument;
- entering, modifying and storing data;
- automated results generation, reporting, etc.

G.2.3.4 Industry/application domain specific

Instruments can be built for specific usage in specific assessment contexts. Flexibility to add and delete information from an assessment instrument may be a desired feature.

G.2.3.5 Usability of user documentation

User documentation should be provided for automated tools. Associated documentation should be easy to understand and provide information on its operation, use, features, and limitations. A paper-based tool should provide forms, definitions and usage instructions. All documentation should be correct and consistent with the operation of the instrument.

G.2.4 Efficiency

Efficiency encompasses a set of attributes that bear on the relationship between the level of performance of the instrument and the amount of resources used under stated conditions.

G.2.4.1 Efficiency of use

The design of the tool and the intended method of assessment can affect its performance and use. Considerations should be given to the following:

- the implementation medium used;
- processing and transaction entry speed;
- distribution capabilities (e.g. paper-based mailing lists, automated networking capabilities);
- data storage, archiving and retrieval;
- sorting abilities;
- data results segregation and presentation abilities;
- data presentation and input design (paper: form design; automated: screen layout and human factors such as ease of data entry, elements per screen, function keys, etc.).

G.2.4.2 Methodology impacts

The assessment methodology employed affects the type and efficiency of the assessment instrument design. Considerations should be given to:

- the approach to data collection;
- the support for distributive and incremental collection of data;
- the assessment type (team based, tool-based self-assessment, assisted self-assessment, etc.).

G.2.5 Maintainability

Maintainability encompasses a set of attributes that bear on the effort needed to make specified modifications (for instance tailoring).

G.2.5.1 Ease of tailoring

Tailoring of the assessment instrument may be required in order to support the creation of extended process, limiting the scope, etc. This should be easy to accomplish possibly directly by the user.

G.2.5.2 Supporting the latest version of the standard

Assessment instruments need to keep pace with the evolution of the standard, so they should be built to allow easy upgrading of the practices and indicator data within the assessment instrument.

G.2.6 Portability

Portability encompasses a set of attributes that bear on the ability of a assessment instrument to be transferred from one environment to another. Again the intended usage and the defined methodology will impact the design when portability of the tool is required. Factors to consider in deciding whether a portable implementation is required are:

- meeting the needs of the users;
- requirements for distributed collection and joint analysis;
- the ability to use the tool(s) in remote locations;
- volume of data collected;
- the need to download data collected for analysis.

G.3 Attributes of types of assessment instruments

G.3.1 Manual instruments (questionnaires, checklists)

Design considerations	– distributed input capabilities – ability to split by process, job function – ability to maintain records – how ratings will be aggregated together from different forms – how to segregate process ratings from extended process ratings – how it will be used
Benefits	– low initial development cost – no training on operational attributes – portable-location independent
Drawbacks	– difficult to scope for organizational characteristics once created – pre-assessment preparation required – assessors may need more training on the concepts associated with the standard process and practices – more difficult to add, change or delete extended practices once created – more difficult to analyze results of multiple organizational units – manual aggregation of results of multiple organizational units – manual calculation of score – difficult to store and use past results for follow-up – security of data – may result in a large amount of paper for large organizations.

G.3.2 On-line instruments (Databases, CASE tools)

Design considerations	– ability to add extended processes as required – during assessment, ability to scope the assessment to the context information – ability to automatically calculate a score – ability to automatically produce the presentation of the results – assessment result storage and retrieval – distributed processing desirable – portability considerations (usability for team interviews, distributed inputs, simultaneous inputs) – ability to handle multiple assessors' inputs – ability to download large amounts of data – performance considerations (access speed, update speed) – usability for team interviews, self-assessment
Benefits	– medium development cost – easy assessment result storage and retrieval – ease of scoping during the assessment – ability to calculate score – ability to generate results, reports, etc.
Drawbacks	– additional training required on how to use an instrument – assessor training or expertise needed on the standard – performance considerations (access speed, update speed) – cost of maintenance and improvement as standard changes – portability is by design – distributed usage is by design

G.3.3 Expert systems

Design considerations	– level of assessor expertise to build into the instrument – ability to add extended processes as required – ability to scope the assessment to the context information – portability (usability for team interviews, distributed inputs, simultaneous inputs) – ability to automatically calculate the score – ability to automatically generate reports, profiles, presentation of results – storage and retrieval capabilities – ability to integrate with other tools (metrics, case, etc.)
Benefits	– less training required for the person performing the assessment – expertise of assessor is built into the instrument – ability to automatically calculate the of score – ability to automatically generate reports, profiles, presentation of results – storage and retrieval capabilities
Drawbacks	– higher development cost – additional training required on how to use instrument – portability is by design only – distributed usage is by design – performance considerations (access speed, update speed) – maintenance and improvement of tools as knowledge base grows

G.4 Instrument attributes associated with usage and methodology

G.4.1 Assessment purpose

Assessment Purpose	Desirable Instrument Attributes
Process improvement	– capture whether practice is or is not implemented – determine how adequate the implemented process is – capture process information related to what needs to be improved – ability to capture information by organizational unit – record scope of the assessment as defined in the assessment input – capture history to demonstrate improvement
Capability determination	– capture whether practice is or is not implemented – determine how adequate the implemented process is – aggregation of scores for all organizational units assessed – record scope of the assessment as defined in the assessment input

G.4.2 Assessment types

Assessment type	Desirable Instrument Attributes
Full assessment	– contains all base practices, and all indicators
Focused assessment	– contains base practices and indicators on process (or job function) being assessed
Basic assessment	– could be a standard vendor developed tool – support for standard practices
Extended process assessment	– capability to support added processes, practices and indicators – ability to delete added processes – ability to calculate and display adequacy results of processes and existence of base practices separately from extended processes.
Initial assessment	– ability to record / store assessment results – ability to record context information – ability to handle a full assessment of all process to baseline the organization
Follow-up assessment	– ability to record assessment results, perhaps incrementally – ability to access historical information – ability to use stored historical data – ability to use past results – ability to perform sampling of processes – ability to display profile changes from previous to current assessment

G.4.3 Assessment approach

Assessment approach	**Desirable instrument attributes:**
Tool based	– ease of data entry and retrieval – expertise of assessor built into the tool: process model architectural concepts may be needed because the user may have less experience with assessment concepts – documentation on how to use tool, install the tool etc. – human factors: table of contents, help screens, tutorials – contains basic indicators and base practices for process being assessed – results may need to be stored for subsequent validation by third party or for process improvement planning – ability to be distributed throughout an organization – ability to input data from multiple sources simultaneously – ability to be distributed by job function of the organizational unit representative, or process owner – ability to maintain an audit trail of who input data – security to restrict assess to organization unit or process owner – output capability: results generation capabilities (profile generation) – output capability: report generation capabilities segregated by organizational unit/process owner – ability to scope the context by attributes representative inputting the data (i.e., process area, job function, etc.)
Team based	– needs to contain indicators and base practices for process being assessed – ability to scope to context attributes of the organization or interview (processes assessed, job function of the interviewee, etc.) – ability to be used in a distributed fashion by multiple team members – ability to assimilate results from multiple sources – portability of the tool to go to remote sites – real-time performance: speed of data input and retrieval is critical – ability to call up practices required for specific interviews – ability to load data prior to the interview (documentation review information, organization model, types of interviews, etc.) – output capability: formal presentation of the results may be required – output capability: interim feedback to participants may be required – output capability: results generation capabilities (profile generation) – output capability: report generation – security considerations if used on-site – restrict access to the results to organizational unit and appropriate representatives

G.4.4 Assessment scope

Assessment scope	Desirable Instrument Attributes
Single organizational unit or one process instantiation	– one profile or score is required – usually one process instantiation to assess – ability to store past scores – ability to record ratings from more than one person
Multiple organizational units or multiple process instantiation	– ability to have the ability to aggregate scores from various instances of the same process – ability to store past scores – ability to merge the results of different instances recorded by different tools for the same assessment. – ability to record ratings from more than one person – ability to record ratings from more than one process instance – ability to record results more than once, multiple profiles may be required – ability to aggregate scores from multiple instances

Annex H (informative)

References

1. ISO/IEC 9126:1991: Software Quality Characteristics:

2. ISO/IEC 12207-1: Information Technology - Part 1: Software life-cycle process

The following documents contain definitions and may provide general guidance to terms in the indicator set.

3. ISO/AFNOR: Dictionary of Computer Science

4. ISO 8402: Quality Management and Quality Assurance; vocabulary

5. ISO 2382/1: Data Processing - Vocabulary

6. ISO/IEC 2382-20: Information Technology - Vocabulary

SPICE

Consolidated product

Software Process Assessment –
Part 6 : Qualification and training of assessors

Version 1.00

(Formerly ATQG Version 1.00)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

These terms and conditions apply to the set of documents developed by the SPICE Project, and published within the Project as Version 1.0, with the following titles:

– Part 1 : *Concepts and introductory guide*
– Part 2 : *A model for process management*
– Part 3 : *Rating processes*
– Part 4 : *Guide to conducting assessment*
– Part 5 : *Construction, selection and use of assessment instruments and tools*
– Part 6 : *Qualification and training of assessors*
– Part 7 : *Guide for use in process improvement*
– Part 8 : *Guide for use in determining supplier process capability*
– Part 9 : *Vocabulary*

1. You may copy and distribute verbatim copies of any or all of the Documents as you receive them, in any medium, provided that you conspicuously and appropriately publish with each copy a copy of these Terms and Conditions. You may charge a fee for the physical act of transferring a copy.
2. You may copy extracts from these documents in materials for internal or public use, providing you provide clear acknowledgment of the source of the material, by citation or other appropriate means.
3. You may not copy, modify, sub-license, or distribute the Documents except as expressly provided under these Terms and Conditions.

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout

Members: Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : ***Concepts and introductory guide***

Product Manager: Terry Rout

– Part 2 : ***A model for process management***

Product Managers: Al Graydon, Mark Paulk

– Part 3 : ***Rating processes***

Product Manager: Harry Barker

– Part 4 : ***Guide to conducting assessment***

Product Manager: Harry Barker

– Part 5 : ***Construction, selection and use of assessment instruments and tools***

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : ***Qualification and training of assessors***

Product Manager: Ron Meegoda

– Part 7 : ***Guide for use in process improvement***

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : ***Guide for use in determining supplier process capability***

Product Manager: John Hamilton

– Part 9 : ***Vocabulary***

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards)

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 6

Qualification and training of assessors

Contents

Annexes

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialling to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being coordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

This part of the standard (Part 6) is for guidance only.

Introduction

Conducting a software process assessment in accordance with the provisions of this International Standard assumes that the assessment team includes at least one qualified assessor. The qualified assessor has the primary responsibility for ensuring that the requirements are met during the assessment.

As described in parts 3 and 4 of this International Standard, rating the assessed practices and processes ultimately depends on the skilled judgement of the assessors. The various elements of the standard provide the framework within which assessors exercise judgement, working together to remove, or at least reduce to a minimum, any subjective elements. Nevertheless, the achievement of an acceptable level of consistency, repeatability and reliability of results relies on competent assessors with appropriate skills, experience, and knowledge of the software process, of the model for processes described in part 2 of this International Standard, and of the conduct of assessment and rating described in parts 3 and 4 of this International Standard.

The qualified assessor in a team has the pivotal role of ensuring that other team members collectively have the right blend of specialized knowledge and assessment skills. The qualified assessor provides the necessary guidance to the team, and helps to moderate the judgements and ratings made by the other members of the team to ensure consistency of interpretation.

This part of the International Standard is concerned with assessor competencies and appropriate education, training and experience including mechanisms that may be used to demonstrate competence and to validate education, training and experience.

This guide is primarily directed to assessors, to those responsible for the selection and development of assessors, and to sponsors of assessments seeking assurance that an assessor is appropriately qualified for the task. In addition, it is useful to organizations wishing to offer appropriate assessment training, or in the future, to organizations or bodies wishing to institute registration schemes for suitably qualified assessors.

1 Scope

This part of the International Standard defines the initial and ongoing qualification of assessors and provides guidance for the preparation and qualification of assessors to perform software process assessments. It describes mechanisms that may be used to demonstrate assessor competence and to validate an assessor’s education, training and experience.

The guidance in this document is applicable to an organizational unit or a sponsor of an assessment wishing to select or specify the type of assessors to perform either self-assessments or independent assessments.

The guidance is also applicable to the identification and demonstration of the competencies necessary for the performance of assessments, and to the process of obtaining those competencies.

Guidance on the competence and qualification of those who perform process capability determination or process improvement activities is outside the scope of this guide.

2 Normative references

There are no normative references in this part of the International Standard.

3 Definitions

For the purposes of this part of this International Standard, the definitions in *Software Process Assessment - Part 9 : Vocabulary* apply.

4 An overview of the assessor and qualification

4.1 The role of the assessor

The role of the assessor, as described in part 4 of this International Standard, is to assess the capability of the software process of an organizational unit in a constructive and objective manner. The assessment should be focused on the process and not the people implementing the process. The role varies depending on the assessment approach as shown in table 1 below.

Table 1 - The role of the assessor in different assessment approaches

Self-assessment approach	Independent assessment approach
Is task and people oriented.	Is task oriented.
Guides the assessment.	Controls the assessment.
Delivers an approach.	Delivers a rating.
Promotes discussion.	Regulates discussion.
Works with projects.	Works separately from projects.
Uses organizational unit's business goals.	May be indifferent to organizational unit's business goals.
Influences through results obtained, relationships established and expertise.	Influences through position and expertise.
Seeks compliance and commitment.	Determines process adequacies.
Is like being a change agent.	Is like being an auditor.

4.2 Philosophy

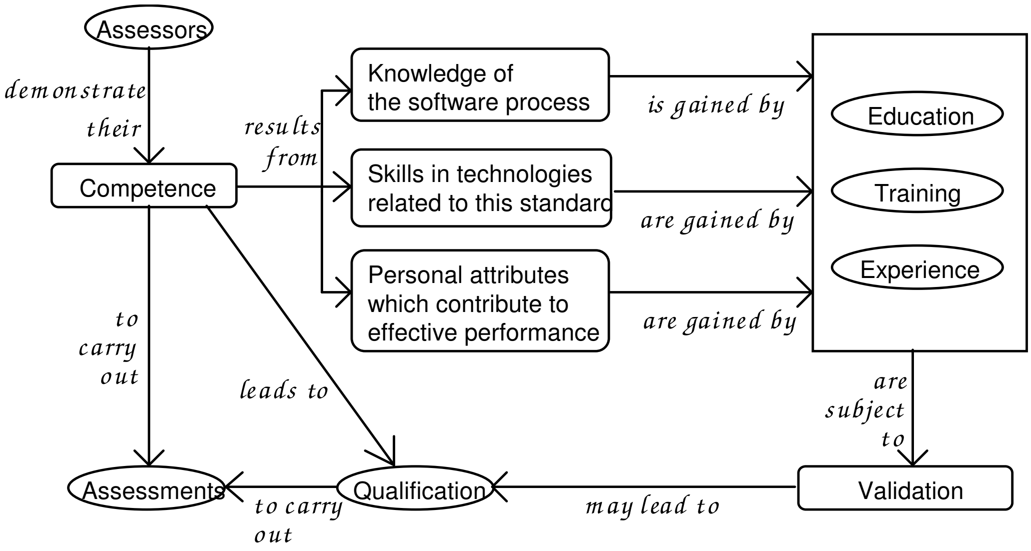


Figure 1 - Entity relationships

Figure 1 shows the key entities and their relationships which may be articulated as follows:

- assessors demonstrate their competence to carry out assessments;
- it is this competence which leads to assessor qualification;
- competence results from
  - the knowledge of the software process;
  - skills in the principle technologies of this International Standard including assessment, rating, assessment instruments, and the process model;
  - personal attributes which contribute to effective performance;
- the knowledge, skills and personal attributes are gained by a combination of education, training and experience;
- an alternative to demonstrable competence is to validate an intending assessor's education, training and experience.

4.3 The process of qualification and on-going qualification

4.3.1 General

Qualified assessors obtain their qualification as shown in figure 2. In addition, the qualification to carry out assessments should be maintained (renewed). The process of qualification and the maintenance of qualification is described in 4.3.2 to 4.3.4 below.

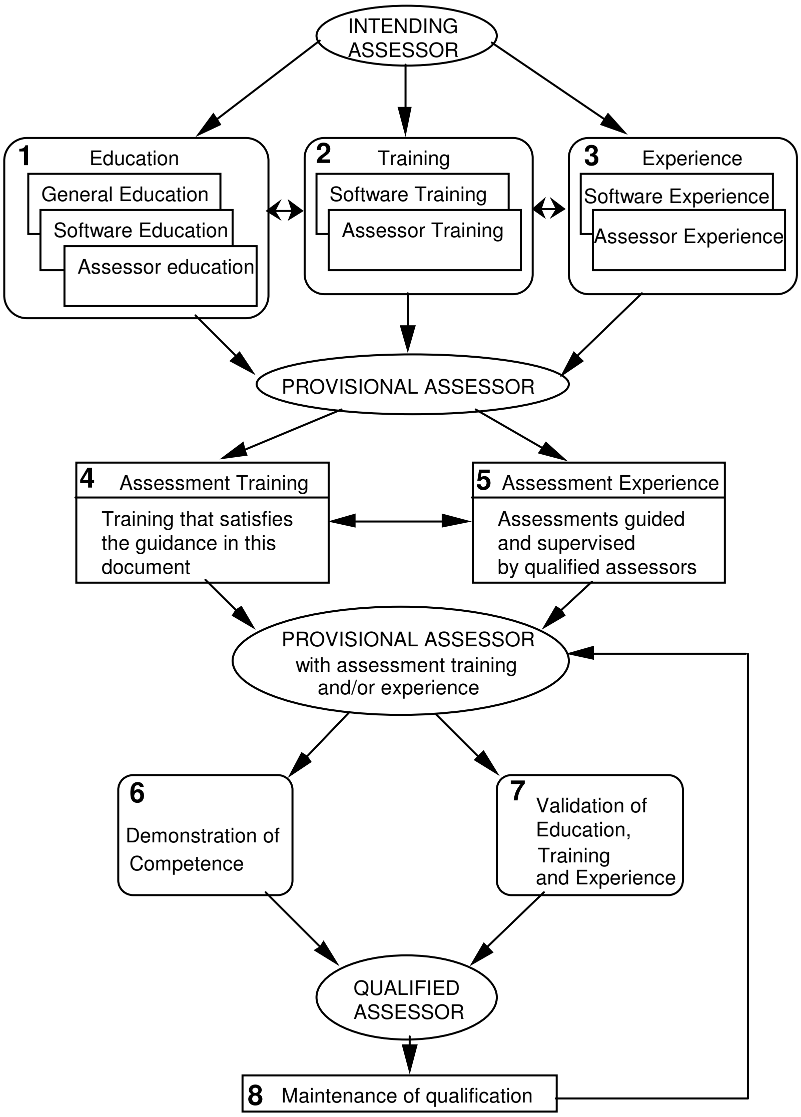

INTENDING ASSESSOR
1 Education
General Education
Software Education
Assessor education
2 Training
Software Training
Assessor Training
3 Experience
Software Experience
Assessor Experience
PROVISIONAL ASSESSOR
4 Assessment Training
Training that satisfies the guidance in this document
5 Assessment Experience
Assessments guided and supervised by qualified assessors
PROVISIONAL ASSESSOR with assessment training and/or experience
6 Demonstration of Competence
7 Validation of Education, Training and Experience
QUALIFIED ASSESSOR
8 Maintenance of qualification

Figure 2 - Path to become a qualified assessor

4.3.2 Becoming a provisional assessor

A provisional assessor is a person who is competent to carry out assessments under the guidance and supervision of a qualified assessor; i.e., an assessor who has reached the required levels of education, training and experience but who has not yet completed the relevant training and/or participated in a sufficient number of assessments conducted according to the provisions of this International Standard.

A provisional assessor, therefore, should be competent to carry out software audits or assessments. A provisional assessor should be trained and experienced in the software process as well as in software process assessment or software quality assessment. In addition, a provisional assessor should have an acceptable level of formal education. Formal education is a combination of general education, software education, and assessor education.

Acceptable levels of education may comprise

– formal courses offered by a college or university;

– professional courses organized by recognized local or international bodies;

– vendor sponsored courses;

– employer sponsored courses.

Acceptable levels of training may comprise

– training provided by recognized local or international bodies;

– training provided by vendors and trainers using the guidance in this part of the International Standard.

Acceptable levels of experience may comprise

– direct "hands-on" experience in specialist areas such as software engineering, software development/maintenance, software quality, or quality assurance;

– management overseeing software specialist areas such as software engineering, software development/maintenance, software quality or quality assurance.

4.3.3 Becoming a qualified assessor

To become a qualified assessor, an assessor should already be a competent software development/maintenance professional or a software audit/assessment professional as described above. In addition, the assessor should have completed training based on the guidance in this part of the International Standard and should have participated in assessments conducted according to the provisions of this International Standard.

4.3.4 Maintenance of the qualification

To maintain (renew) the qualification, assessors should update their knowledge and skills by engaging in a number of professional activities as well as carrying out further assessments conducted according to the provisions of this International Standard.

5 Assessor competence

5.1 The software process

An assessor should be familiar with software development and maintenance including various life cycle models (see figure 2, item 6) and be able to demonstrate competence in at least one of the process categories of the process model described in part 2 of this International Standard.

An assessor should also be able to demonstrate familiarity with the software process, and should be experienced with the use of one or more development models such as Waterfall or Rapid Prototyping.

In addition, an assessor should show an understanding of the activities required to support the software process, including when and how they should be applied according to the development model chosen within the application domain in which the assessor is experienced.

An assessor should be familiar with a range of relevant software engineering standards.

5.2 Assessment technology

Assessors should demonstrate competence in all aspects of the technology of assessment pertaining to this International Standard, particularly the core aspects included in parts 2 to 5 as shown in figure 3.

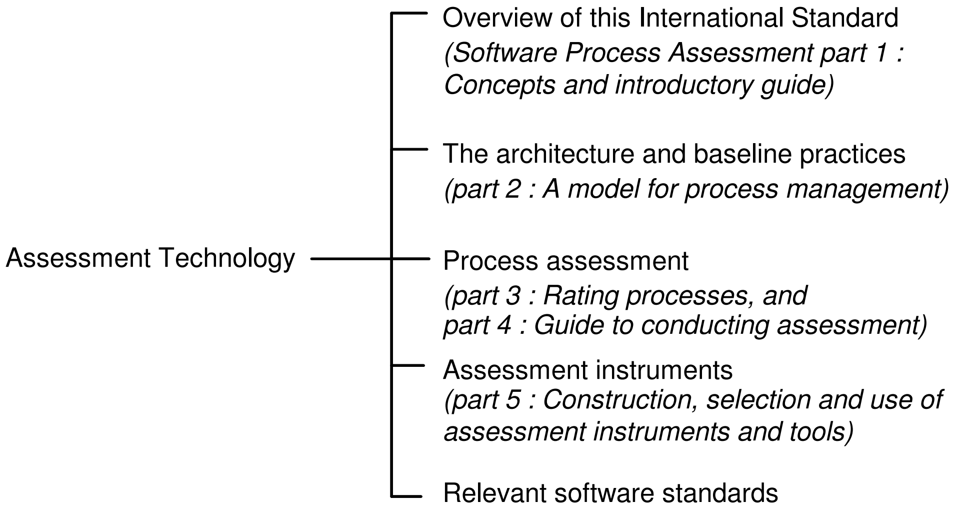

Assessment Technology
Overview of this International Standard
(Software Process Assessment part 1 : Concepts and introductory guide)
The architecture and baseline practices
(part 2 : A model for process management)
Process assessment
(part 3 : Rating processes, and part 4 : Guide to conducting assessment)
Assessment instruments
(part 5 : Construction, selection and use of assessment instruments and tools)
Relevant software standards

Figure 3 - Demonstrable elements of assessment technology

5.3 Personal attributes

5.3.1. General

Assessors should possess the personal attributes shown in figure 4 and described below.

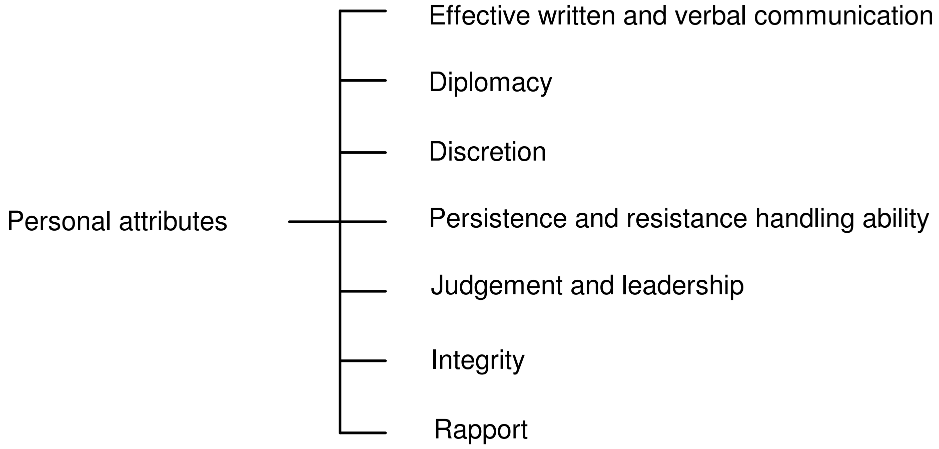

Personal attributes
Effective written and verbal communication
Diplomacy
Discretion
Persistence and resistance handling ability
Judgement and leadership
Integrity
Rapport

Figure 4 - Personal attributes

5.3.2. Effective written and verbal communication

Assessors who perform assessments will interact with members of the organizational unit being assessed. They may be feeding back the results of the assessment in the form of written reports and/or presentations. Assessors should be able to communicate the findings of the assessments in a clear, non-judgmental style. Assessment findings should be documented in clear and unambiguous language.

5.3.3. Diplomacy

Assessors should act with professionalism and decorum at all times. Independent assessors are guests of the organizational unit being assessed and their conduct should be above reproach at all times.

5.3.4. Discretion

Assessors should develop and maintain the confidence of the assessment participants. In particular, assessors should preserve the confidentiality of the results of the assessment and of information received during an assessment in accordance with the terms of any confidentiality agreement included in the assessment constraints (see parts 3 and 4 of this International Standard).

5.3.5. Persistence and resistance handling ability

Assessors should be persistent in carrying out the duties that are expected of them. They should be able to resolve any conflicts and handle any resistance that they may experience from assessment participants.

5.3.6. Judgement and leadership

It is critical that the organizational unit being assessed has confidence in, and respect for, the assessment team leader, team co-ordinator and team members. If they are not respected within the organizational unit, then the assessment findings may not be accepted by the organizational unit.

5.3.7. Integrity

The assessment team leader, team co-ordinator and team members should have no conflict of interest in performing the assessment. For example, the assessment team members, the leader and the co-ordinator should not be individuals whose performance is being measured by the improvements enacted within the organizational unit. If the team members' individual performances are being measured by the outcome of the assessment, they cannot be considered objective.

5.3.8. Rapport

Individuals who because of their organizational position or personality will stifle the open and honest flow of information should not participate in the assessment. For example, managers who evaluate the performance of individuals involved in the projects being assessed should not be assessment team members. Project personnel might be reluctant to disclose problem areas to their own management as their individual performance may be affected.

6 Validation of education, training and experience

6.1 Overview

Validation of an assessor’s education, training and experience (figure 2, item 7) is an alternative to the demonstration of competence as a means of qualification. The education, training and experience may be validated by a review of these elements. The right balance is of prime importance. In general terms, the balance includes general education, software education and assessor education together with training and experience in both software development activities and assessments.

The following factors should be considered when reviewing the education, training and experience of an assessor.

Duration: The amount of time the assessor has spent in a particular process category. (See part 2 of this International Standard for process categories).

Range: The assessor’s breadth of exposure to the process categories.

Depth: The level of specialization.

Responsibility: The extent to which an assessor has held responsibility in terms of both range and depth.

Currency: How recent is the assessor’s education, training and experience, and the extent to which the assessor’s knowledge and skill have been updated.

6.2 Education

Education is shown in figure 2, item 1. Assessors should maintain evidence of their formal education in terms of certificates and official course outlines for validation. The following levels of educational achievement may be considered as appropriate in the categories of general education, software education, and assessor education.

General education: In general, a degree or equivalent of any discipline from an educational establishment.

Software education: A degree or equivalent in Computer Science, Software Engineering or similar, or as an alternative, formal education in these areas or the software process supplementing a general degree, for example CQA.

Assessor education: Qualification as an assessor or auditor, general assessment experience and specific software engineering education.

6.3 Training

Training is shown in figure 2, item 2. An assessor's training should be recorded (see Annex A) for validation.

Acceptable training would cover at least some aspects of software development.

In order to be familiar with software development and maintenance processes, the assessor should have been trained, or have validated experience, in all the processes in the Engineering (ENG) process category.

Project management or technical leadership training provides a background in the Customer Supplier (CUS) and the Organizational (ORG) process categories. Assessors need not have been trained in each process in the two process categories, but should be familiar and conversant with the topics. Assessors should have extensive training in at least one of the processes in these two process categories.

6.4 Experience

6.4.1. General

Experience is shown in figure 2, item 3. Assessors' experience should be recorded (see Annex B) for validation.

Some of the factors which should be taken into account when assessing the relevance of experience in each of the process categories are addressed in 6.4.2 to 6.4.6 below. In lieu of personal experience, the teaching of the particular subject at a suitable level may suffice.

There is an interaction between experience and training: training alone is insufficient. There is also a beneficial interaction between experiences in different roles. For example, team leaders or managers of projects may have had contact with software configuration management and software quality assurance functions. The experience gained may overlap and cover a number of process categories in any particular assignment.

In consequence, recent graduates, or individuals who have spent their entire working lives in a single process category, are unlikely to have accumulated sufficiently broad experience.

6.4.2. Customer-supplier process category

The key element of these practices is joint customer and supplier interaction. Participation in activities within an organizational unit with a recognized quality management system would be helpful. The provision of customer references would aid verification.

6.4.3. Engineering process category

Assessors should show evidence of work experience that shows the use of some of the development practices within this process category. Experience solely in the development of user documentation is insufficient.

6.4.4. Project process category

Ideally, assessors should demonstrate that they have managed a project or projects in the software industry for a period of at least one (1) year. The project should have included management of subcontracted activities. Experience which shows acceptance of responsibility for human resource management in the project category would be relevant to the selection and training of assessment team members.

6.4.5. Support process category

A key feature of these practices is the development of plans and the measurement of performance against these plans. Relevant experience includes developing project or user documentation.

Assessors should be able to demonstrate familiarity with software quality assurance and quality management systems. Examples include participation in activities within an organization with an approved quality management system, familiarity with independent assessments, or qualification as an auditor or assessor under a national scheme.

6.4.6. Organization process category

Assessors should be able to demonstrate experience as managers, consultants or assessors involved in the processes in this process category.

6.5 Training in assessments using this International Standard

This activity is shown in figure 2, item 4. Assessors' training should be recorded (see Annex A) for validation.

A training course to cover the requirements and assessment elements of this International Standard should comprise at least the following topic areas:

6.5.1. Overview of this International Standard

- Background
- Architecture and principles
- The component parts of the International Standard
- Vocabulary and definitions
- Comparison of this International Standard with other standards/methodologies
- Assessment vs. auditing
- How to use the parts of the International Standard

6.5.2. The process model

(based on part 2 : A model for process management)

- Process categories
- Processes and the base practices
- Capability levels, common features and generic practices
- Extended processes
- How to use part 2 of this International Standard.

6.5.3. Process Assessment

(Based on part 3: Rating processes and part 4 : Guide to conducting assessment)

- Assessment preparation
- Conduct of assessments
- Determination of actual ratings
- Determination of derived ratings
- Validation of ratings
- Presentation of assessment results
- Requirements for conformance
- How to use parts 3 and 4 of this International Standard

6.5.4. Assessment Instruments

(Based on part 5 : Construction, selection and use of assessment instruments and tools)

- Selecting instruments
- Building instruments
- Using Instruments
- How to use part 5 of this International Standard

6.6 Experience of assessments using this International Standard

This activity is shown in figure 2, item 5. Assessors' experience of conducting assessments using this International Standard should be recorded (see Annex C) for validation.

In addition to the training mentioned above, it is recommended that a qualified assessor should have:

- participated as a provisional assessor in at least two (2) assessments conducted according to the provisions of this International Standard
- or participated as a provisional assessor in one (1) assessment and as an observer in three (3) assessments conducted according to the provisions of this International Standard.

Training (clause 6.4) and participation in assessments should be formally documented by the trainer or the assessment team leader respectively.

6.7 Maintenance of the qualification

Assessors should maintain (renew) the qualification (figure 2, item 8) by engaging in a combination of the following activities

- on the job experience as a qualified assessor;
- attending professional seminars;
- giving presentations;
- teaching or developing courses;
- engaging in professional association activities;
- publishing articles or books;
- self training or education using this International Standard;
- active involvement or leadership in the organizational unit's improvement teams.

Assessors' professional activities should be recorded (see Annex D and Annex E) for validation.

6.8 Maintenance of records

The following records should be maintained by all assessors and intending assessors

- educational certificates and course outlines;
- training records (see Annex A);
- verified records of experience (see Annex B);
- verified records of attending training course(s) in this International Standard (see Annex A);
- verified records of participation in assessments conducted according to the provisions of this International Standard (see Annex C);
- assessment logs (see Annex D);
- logs of professional activities (see Annex E).

Annex A (informative)

Training record

The following template may be used to record an assessor’s training.

Table 9 - Training Record

Training course	Description of training	Dates	Hours	Training provider

Training course: The name of the training course and where held.

Description of training: A short overview of the training specifying the covered processes and process categories provided by either the assessor or the training provider.

Dates: Start and end dates of training.

Hours: Number of hours of training.

Training provider: The name and the signature of the training provider with the training provider's official stamp or logo. Alternatively, a certificate of completion bearing this information may be attached.

Annex B (informative)

Record of experience

The following is an example of an assessor's record of experience in the software process.

Table 10- Record of experience

Process Category	Description of experience	Dates	Level	Verification
CUS				
ENG				
PRO				
SUP				
ORG				
Other				

Process category: Process categories of the process model in part 2 of this International Standard. The assessor may describe any other category which may be relevant.

Description of experience: Short overviews covering the involvement in different processes within the process categories prepared by the assessor.

Level: Level of involvement (i.e., as an assessor, trainee, management, supervisor). Assessors may describe their involvement in the same process category in more than one cell if the level of involvement is different.

Dates: Dates of involvement in different categories.

Verification: Signature and the position of the supervisor, manager or referee who can verify the assessor's experience in each category.

Annex C (informative)

Record of participation

The following template may be used to record an assessor’s participation as a provisional assessor or as an observer in assessments conducted according to the provisions of this International Standard.

The involvement in assessments should be verified by a qualified assessor or the assessment team leader. Each assessment is recorded in a format similar to the one below and is completed by a qualified assessor or an assessment team leader.

Name of the person:
Date:
No. of days for the assessment:
Scope of the assessment:
Process categories/areas assessed by the person:
Organization/Organizational unit:

Effective Communications:

Were the discussions with the customer reasonable?	Yes/no
Was a satisfactory understanding of this International Standard shown?	Yes/no
Was the inter team relationship satisfactory?	Yes/no

Judgement and Leadership:

Were the assessment activities completed in a timely manner?	Yes/no
Were the interviews conducted satisfactorily?	Yes/no

Integrity:

Reasonable sample taken?	Yes/no
Range of activity satisfactory?	Yes/no
Depth of questioning satisfactory?	Yes/no
Review of results consistent?	Yes/no

Rapport :

Communication - telling the good and bad news:	satisfactory/unsatisfactory
Review of the programme:	satisfactory/unsatisfactory
Conduct:	satisfactory/unsatisfactory
Team Management:	satisfactory/unsatisfactory

Comments: (on Diplomacy, Discretion, Persistence and Resistance handling ability)

Performance: Acceptable/More Experience Required/Not acceptable

Name and signature of qualified assessor/ team leader: ....

Annex D (informative)

Assessment log

The following is a sample of an assessment log which may be used to record the details of assessments conducted according to the provisions of this International Standard which an assessor has performed as a qualified assessor.

Table 11- Assessment log

Date	Assessment	No of days	Categories assessed	Verification

Date: Start date of the assessment.

Assessment: A short description of the assessment to be written by the assessor. The description should include the level of involvement of the assessor (i.e.as team member, team leader, team co-ordinator) and the number of assessors in the team.

No of days: Duration of the assessment in days.

Categories assessed: The process categories covered by the assessment.

Verification: The signature of a senior manager of the organizational unit assessed, with the stamp or logo. Individual assessment logs may be retained to maintain confidentiality.

Annex E (informative)

Professional activities log

The following template may be used to record the professional activities of an assessor for maintenance (renewal) of the qualification.

Table 12- Professional activities log

Date	Activity	Location	Hours

Date: Date of the professional activity.

Activity: The title and a short description of the activity.

Location: Address with room numbers if applicable.

Hours: Estimated number of hours of the activity.

Annex F (informative)

Mechanisms for the demonstration of competence

F.1 General

An intending assessor may demonstrate competence in each category through a number of mechanisms. The choice of an acceptable mechanisms may be at the discretion of the sponsor of an assessment or the employer of the assessor. The same mechanisms may be used for self evaluation.

The following are examples of such mechanisms.

F.2 Example 1 for demonstration

Table 2 is an example of a matrix which may be set up to determine the competence of an assessor. The left hand column with the title "Category of competence" consists of the broad categories of competence which an assessor should demonstrate. Each sub-category demonstrated may be written in the appropriate cell under the appropriate method of demonstration (see table 2). Finally the number of sub-topic categories may be counted to determine the level of competence.

Alternatively, the sub-categories (instead of the main topic categories) may be provided in the left hand column with the title "Category of competence". A tick may be placed in the appropriate cell when competence in the particular sub-category is demonstrated. The score, represented by the total number of ticks, may be used to determine the level of competence.

Table 2 - Demonstration of competence against different categories

	Method of demonstration			
Category of competence	**Career progression**	**Technology awareness**	**Breadth of performance**	**Other**
CUS Process Category				
ENG Process Category				
PRO Process Category				
SUP Process Category				
ORG Process Category				
Assessment technology				
Personal attributes				

F.3 Example 2 for demonstration

The following example is based on a joint employee-supervisor review. It encourages assessors to describe their own competencies. This approach is particularly helpful in assessing one's own competence to perform assessments conducted according to the provisions of this International Standard, and if used on a regular basis, for building competence over time.

1. Rate your current level of competence to perform assessments, conducted according to the provisions of this International Standard, on a scale of High/Medium/Low.
2. Rate the level of feedback you received on your performance in assessments in the past (High/Medium/Low).
3. Conduct a joint discussion with your supervisor or referees to identify the areas of competence which are relevant to your current assignment or any past assignments. List the assignments against each area of competence in a matrix. Then list specific actions taken, personal attributes established, or outcomes produced which you have used to demonstrate your competence in each of the relevant areas (table 3).

Table 3 - Demonstration of competence against assignments

Areas of competence	Assignments	How demonstrated
Competence 1	Assignment 1	
	Assignment 2	
Competence 2	Assignment 1	
	Assignment 2	

4. Rate the need for improvement of your competence in software assessment (High/Medium/Low).
5. Develop an action plan to improve your competence. Identify the items or areas to be improved, methods of improvement (e.g., training, reading, work assignments, self-paced learning, mentoring) and ways to measure progress.
6. Implement your plan and describe successes, failures and the reasons.
7. Identify what needs to be done next.

Table 4 - Self improvement

Item to improve	Improvement method	Method to measure progress

Annex G (informative)

Mechanisms for the validation of education, training and experience

G.1 General

Validation of an assessor's education, training and experience may be performed in a number of ways. The choice of an acceptable mechanisms should be at the discretion of the sponsor of an assessment or the employer of the assessor. The same mechanisms may be used for self evaluation.

Example mechanisms are described below.

G.2 Example 1 for validation

The following mechanism is based on allocating points to a number of criteria. The example includes a suggestion of the way points may be distributed, the maximum that may be attained in each category, and the acceptable minimum to become an assessor. The allocation of points may be adjusted based on the duration, range, responsibility, depth, and currency of an assessor's education, training and experience.

	Points
Education (Maximum = 4, Minimum = 2)	
Degree or equivalent level of education in any discipline.	1
Any formal course in the Software Process, Computer Science, Software Development, Software Engineering, or Software Quality	1
Degree or equivalent level of education in the Software Process, Computer Science, Software Development, Software Engineering, or Software Quality	2
Assessor education in terms of a national or an international scheme. (e.g., CQA, TickIT or SEI appraiser)	2

<u>Training</u> (Maximum = 10, Minimum =6)

Customer/Supplier process category [CUS]	1
Engineering process category [ENG]	1
Project process category [PRO]	1
Support process category [SUP]	1
Organization process category [ORG]	1
Training Course on this International Standard	5

<u>Experience</u> (Maximum = 5, Minimum = 3)

Customer/Supplier process category [CUS]	1
Engineering process category [ENG]	1
Project process category [PRO]	1
Support process category [SUP]	1
Organization process category [ORG]	1

Once an assessor’s suitability is quantified, the outcome may be as shown in table 5.

Table 5 - Recommended outcome of validation

Number of points scored	Recommended outcome
11 or above	suitable to be an assessor
7 to 10	more education, training or experience required.
below 7	not suitable at present

G.3 Example 2 for validation

The following example is a "check list" type of approach that a sponsor or an employer (or the intending assessor) may use to determine the adequacy of education, training and experience by examining a number of items. (See tables 6, 7 and 8.) A tick may be placed in the appropriate position after the check.

The sponsor or employer should determine the minimum number of fully adequate areas and partially adequate areas which would qualify an assessor to perform an assessment.

Table 6 - Validation of the software process against a checklist

Items to validate	Adequate				**Items to Examine**	**Notes and Commentary**
	Fully	**Partially**	**No**	**Unknow**		
Software process						
a. Education					• Education accreditation • Degree earned • Number of credit hours • Subject studied	Base or higher degree in a software related discipline preferred.
b. Training					• Training supplier • Type (video, instructor led etc.) • Classroom hours • Subjects matter • Other assessment models	Training and education alone are not sufficient to qualify an assessor.
c. Experience					• Covers assessment scope • Expertise in at least one process. • Business domain • Application domain • Process variants, if applicable • Other assessment accreditation • Level of responsibility attained	Experience in the process categories and processes of the assessment scope is required.

Table 7 - Validation of assessment technology against a checklist

Items to validate	Adequate				Items to Examine	Notes and Commentary
	Fully	Partially	No	Unknow		
Assessment Technology						
a. Education					• Educational Institution • Degrees or certificate earned • Classroom hours	Formal education may be used to gain understanding of this International Standard. Education alone is insufficient to qualify an assessor.
b. Training					• Trainer credentials • Type (video, instructor led etc.) • Coverage (clause 6.5): – Components of the standard – Process model and baseline practices – Process assessment – Assessment Instruments	Training may be used to obtain knowledge of the components of this International Standard; assessment methodologies; managing and conducting an assessment; design of extended processes as well as identifying the mapping of OU processes to the processes defined in part 2 of this International Standard.
c. Experience					• Previous assessments conducted • Previous assessment and assessor evaluations • Assessment methodologies used • Provisional assessor requirements satisfied • Creating an assessment methodology, if applicable • Assessment Instrument(s)	This is validating the set of experience for consideration of an assessor as a qualified assessor.

Table 8 - Validation of personal attributes against a check-list

Items to validate	Adequate				Items to Examine	Notes and Commentary
	Fully	Partially	No	Unknow		
Personal Attributes						
a. Education					• Education accreditation • Degree earned • Number of credit hours	Formal education may include courses in ethics or business philosophy.
b. Training					• Training supplier • Type (video, instructor led etc.) • Classroom hours • Subject matter: – Total Quality leadership – effective meetings – team building – communication skills – change management	Look for training completion, understanding and application of principles.
c. Experience					Assessment evaluations Presentations Writing skills Leading change (self-assessments)	Experience is the most reliable indication that an individual posses the personal attributes required of an assessor.

Key:

Fully adequate: The information submitted clearly demonstrates that the assessor has the knowledge and skills in the specific area to successfully perform assessments conducted according to the provisions of this International Standard.

Partially adequate: The information submitted indicates that the assessor has at least some of knowledge and skills necessary to successfully perform assessments conducted according to the provisions of this International Standard. Additional information may be requested. Alternately, the composition of the assessment team may be altered to include individuals whose knowledge and skills can augment those of the assessor.

Not adequate: The information submitted clearly indicates that the assessor does not posses the knowledge and skills in the specific areas to successfully perform assessments conducted according to the provisions of this International Standard.

Unknown: The information submitted does not address the specific knowledge and skills outlined in this part of the International Standard. Additional information may be needed before a determination can be made.

Annex H (informative)

Glossary

CQA	Certified Quality Analyst of the Quality Assurance Institute, Orlando, Florida, USA.
SEI	**S**oftware **E**ngineering **I**nstitute, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA.
TickIT	A program for ISO9000 registration of software auditors.

Annex J (informative)

References

1. IT and the Competency Debate "Skill Vs Knowledge", A major issue - A. W. Goldsworthy, Australian Computer Journal, August 1993, pp 113 to 123.
2. Auditors' guide for software sector quality system registration under ISO 9001, Draft release 0.31 of 21 September 1993, Part 4 pp 4-1 to 4-16.
3. General criteria for certification bodies operating certification of personnel, BS 7513 : 1989, EN 45013 : 1989.
4. Programme for the certificate in quality systems and auditing principles and the registration of quality system auditors, New Zealand Organization for Quality Assurance Incorporated.
5. Professional qualification criteria for Information Technology quality system auditors, Draft version June 1993, ICIT Technology.
6. The National Registration Scheme for TickIT Auditors, Version of 1 January, 1993.
7. COMPETENCE - COMPETENCY: there is a difference!, Australian and UK Management Standards, Robin Plummer, Management (Australian Institute of Management journal), August 1994, pp 27 to 29.
8. ISO10011 - Guidelines for Auditing Quality Systems.
9. ISO12207 - Information Technology - Software Life Cycle Processes.

Consolidated product

Software Process Assessment – Part 7 : Guide for use in process improvement Version 1.00

(Formerly PIG 1.00)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

These terms and conditions apply to the set of documents developed by the SPICE Project, and published within the Project as Version 1.0, with the following titles:

– Part 1 : *Concepts and introductory guide*
– Part 2 : *A model for process management*
– Part 3 : *Rating processes*
– Part 4 : *Guide to conducting assessment*
– Part 5 : *Construction, selection and use of assessment instruments and tools*
– Part 6 : *Qualification and training of assessors*
– Part 7 : *Guide for use in process improvement*
– Part 8 : *Guide for use in determining supplier process capability*
– Part 9 : *Vocabulary*

1. You may copy and distribute verbatim copies of any or all of the Documents as you receive them, in any medium, provided that you conspicuously and appropriately publish with each copy a copy of these Terms and Conditions. You may charge a fee for the physical act of transferring a copy.
2. You may copy extracts from these documents in materials for internal or public use, providing you provide clear acknowledgment of the source of the material, by citation or other appropriate means.
3. You may not copy, modify, sub-license, or distribute the Documents except as expressly provided under these Terms and Conditions.

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout
Members:	Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : *Concepts and introductory guide*

Product Manager: Terry Rout

– Part 2 : *A model for process management*

Product Managers: Al Graydon, Mark Paulk

– Part 3 : *Rating processes*

Product Manager: Harry Barker

– Part 4 : *Guide to conducting assessment*

Product Manager: Harry Barker

– Part 5 : *Construction, selection and use of assessment instruments and tools*

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : *Qualification and training of assessors*

Product Manager: Ron Meegoda

– Part 7 : *Guide for use in process improvement*

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : *Guide for use in determining supplier process capability*

Product Manager: John Hamilton

– Part 9 : *Vocabulary*

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards)

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 7

Guide for use in process improvement

Contents

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialling to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being coordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

This part of the standard (Part 7) is for guidance only.

Introduction

The needs and business goals of an organization are often centred around achieving enhanced customer satisfaction and greater competitiveness. For organizations with a dependence on software, these key management concerns become drivers that initiate software process improvement throughout the organization with goals of higher software quality, lower development and maintenance costs, shorter time to market, and increased predictability and controllability of software products and processes.

Software process improvement is best considered as a continuous process, where an organization moves continually around an improvement cycle. Within this cycle improvement is accomplished in a series of steps or specific improvement actions such as introducing new or changed practices into software processes or removing old ones. An important step in the improvement cycle, however, is the execution of some form of data gathering to establish the initial state, and subsequently to confirm the improvements.

This part of the International Standard provides guidance on using software process assessment as the primary means of understanding the current state of an organization's software processes, and on using the results of the assessment to formulate and prioritize improvement plans. This guidance is embodied within a general framework for the use of process metrics in software process improvement.

The improvement framework is built on the framework for quality improvement embodied in ISO 9004-4. It is assumed that the reader is familiar with the concepts in ISO 9000-4.

This part of the International Standard is primarily aimed at the management of an organization considering or undertaking a software process improvement programme, possibly as a result of a process capability determination; members of improvement teams, particularly leaders and facilitators; software engineers; and external consultants helping organizations to undertake software process improvement.

This process improvement guide addresses the following topics:

– an overview of process improvement – the factors which drive software process improvement and general principles which underpin it;

– a methodology for process improvement – an eight step model for improving software processes within a continuous improvement cycle;

– cultural issues – aspects of organizational culture that are critical for successful process improvement;

– management – software process improvement from a management perspective including an overall framework for process measurement.

Annexes provide supplementary information including examples of the use of the process measurement framework; an illustrative case study of application of this guide; mappings between *Improve the Process* (the ORG.3 process in part 2 of this International Standard), the Plan-Do-Check-Act (PDCA) cycle and the steps in the model described in this guide; and cross references to ISO 9004-4.

1 Scope

This part of the International Standard provides guidance on using software process assessment as part of a framework and method for performing software process improvement in a continuous cycle. The guidance covers:

- invoking a software process assessment;
- using the results of a software process assessment;
- measuring software process effectiveness and improvement effectiveness;
- identifying improvement actions aligned to business goals;
- using the process model in part 2 of this International Standard as a route map for improvement;
- cultural issues in the context of software process improvement;
- dealing with management issues for software process improvement.

The guidance provided does not presume specific organizational structures, management philosophies, software life cycle models or software development methods. The concepts and principles are appropriate for the full range of different business needs, application domains and sizes of organization, so that they may be used by all types of software organizations to guide their improvement activities.

An organization may select all or any subset of the software processes from the process model (defined in part 2 of this International Standard) for assessment and improvement in the light of its particular circumstances and needs.

The guidance provides a framework for implementing improvements as a continuous cycle, but there is no reason why the organization could not employ the guidance for a single cycle of improvement activity.

2 Normative References

There are no normative references ion this part of the International Standard.

3 Definitions

For the purposes of this part of this International Standard, the definitions in *Software Process Assessment - Part 9 : Vocabulary* apply.

4 Overview of process improvement

4.1 Drivers

The needs and business goals of an organization are often centred around achieving enhanced customer satisfaction and greater competitiveness. For organizations with a dependence on software, these key management concerns become drivers that initiate software process improvement throughout the organization with goals of higher software quality, lower development and maintenance costs, shorter time to market, and increased predictability and controllability of software products and processes.

4.2 Process improvement basics

Successful changes to the software process start at the top of the organization. Senior management leadership is required to launch a change effort and to provide continuing resources and impetus, although ultimately, everyone in the organization is involved.

In summary:

- software process improvement demands investment, planning, dedicated people, management time and capital investment;
- process improvement is a team effort – those not participating may miss the benefits and may even inhibit progress;
- effective change requires an understanding of the current process and a goal – to use a map, you must know where you are and where you want to get to;
- change is continuous not a one-shot effort – it involves continual learning and evolution;
- software process changes will not be sustained without conscious effort and periodic reinforcement.

4.3 General principles

The needs and business goals of the organization determine the software process improvement goals that help to identify improvement actions and their priorities. Software process improvement is accomplished in a series of steps or specific improvement actions such as introducing new or changed practices into software processes or removing old ones. The process model in part 2 of this International Standard may be used to identify practices to be included to improve the capability of each process. Achievement of process improvement goals should be measured quantitatively .

The basic principles of software process improvement described are:

- software process improvement is conducted on the basis of process assessment and process effectiveness measures;
- software process assessment produces a current process capability profile which may be compared with a target profile based on the organization's needs and business goals;
- process effectiveness measures relate the identification and priorities of improvement actions to the organization's needs and business goals, and achievement of software process goals;
- software process improvement is a continuous process. Improvement goals identified and agreed within the organization are realized through a process improvement programme that continues through multiple cycles of planning, implementing and monitoring activities;
- improvement actions identified within a process improvement programme are implemented as process improvement projects;
- metrics are used for monitoring the improvement process in order to indicate progress and to make necessary adjustments;
- software process assessment may be repeated in order to confirm that the improvements have been achieved;
- mitigation of risk is a component of process improvement and should be addressed from two viewpoints:
  - the risk inherent in the current situation;
  - the risk of failure in the improvement initiative.

Software process improvement plans and records may be used to support process capability determination, when the proposed process capability needed to meet a contractual requirement exceeds the currently assessed profile (see part 8 of this International Standard).

4.4 Process improvement context

The context for software process improvement and its major interfaces are shown in figure 1. The interfaces are as follows:

- the organization's needs and business goals which are main stimuli for process improvement;
- industry norms and benchmarks providing reference information for improvement planning;
- improvements in the organizational unit's (OU's) software processes as a result.

Process improvement uses other components of this International Standard as follows:

- process assessment (as described in the Parts 3 and 4 of this International Standard) is invoked to establish the current capability;
- the assessment results consist of process profiles and all the contextual information held in the assessment record;
- the process model within part 2 of this International Standard is used as a route map to define processes to be improved, set priorities and identify improvement actions;
- existing improvement initiatives may need to be adjusted to support a new target capability resulting from a process capability determination (see part 8 of this International Standard);
- improvement plans and records may assist in establishing customer confidence during process capability determination (see part 8 of this International Standard).

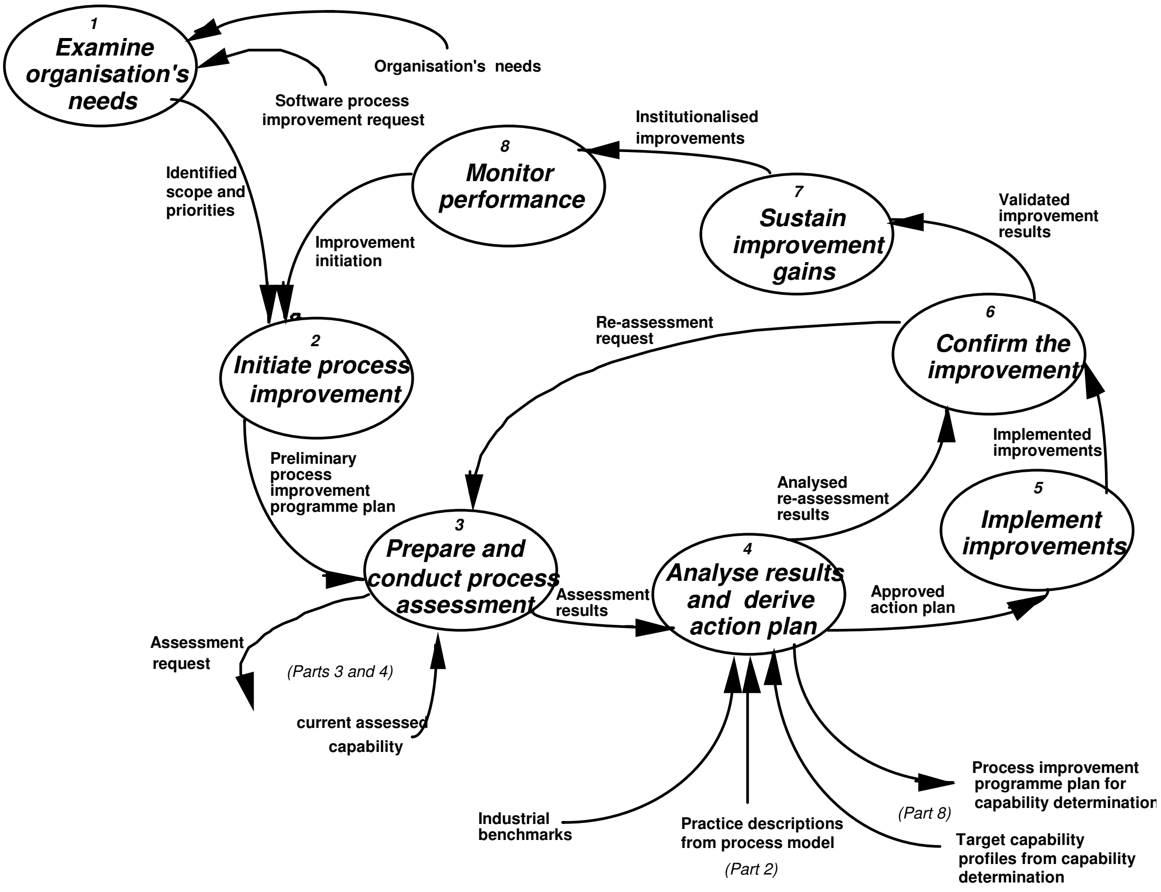
Industry p
Impr
Software proces
benchma
Org
improvement re
soft
Organization's
needs and
business goals
PR

Figure 1 – Process improvement context

5 Methodology for software process improvement

When an organization is well motivated and managed for software process improvement (see clauses 6 and 7), it undertakes and implements a number of process improvement activities. Software process improvement benefits accumulate permanently when an organization pursues process improvement activities in a consistent and disciplined series of steps based on data collection and analysis.

Figure 2 illustrates the steps of a methodology for continuous software process improvement using the components of this International Standard.



1
Examine organisation's needs
Organisation's needs
Software process improvement request
Institutionalised improvements
8
Monitor performance
7
Sustain improvement gains
Validated improvement results
Identified scope and priorities
Improvement initiation
Re-assessment request
6
Confirm the improvement
2
Initiate process improvement
Implemented improvements
Preliminary process improvement programme plan
Analysed re-assessment results
5
Implement improvements
3
Prepare and conduct process assessment
4
Analyse results and derive action plan
Assessment results
Approved action plan
Assessment request
(Parts 3 and 4)
current assessed capability
Process improvement programme plan for capability determination
(Part 8)
Industrial benchmarks
Practice descriptions from process model
(Part 2)
Target capability profiles from capability determination

Figure 2 – Software process improvement steps

A comprehensive process improvement programme may identify improvement goals to be attained over several iterations of the improvement cycle.

The steps in the improvement cycle are described in detail below in terms of their activities and tasks, and their inputs and outputs.

5.1 Examine the organization's needs and business goals

A process improvement programme starts with the recognition of the organization's needs and business goals, usually based on one of the main drivers identified in 4.1. This recognition could be derived from any of:

- formulation of a mission statement or a long-term vision (see 7.1);
- analysis of organization's business goals;
- analysis of the organization's current shared values (see 6.2);
- the organization's readiness for undertaking a process improvement programme;
- other internal or external stimuli.

External stimuli that may trigger a process improvement programme include:

- declining market share;
- marketing analysis;
- feedback from customers;
- competitiveness changes in the market;
- requirements to meet specific industry benchmarks;
- new requirements from society.

Internal stimuli that may trigger a process improvement programme include:

- declining or unsatisfactory profitability;
- declining staff satisfaction.

From an analysis of the organization’s needs and existing stimuli for improvement, the objectives of process improvement may be identified and described in terms of:

- goals for product quality, time to market, and production costs preferably expressed as quantitative targets,
- an ability to predict and control the production process and its related risks.

The final stage of the preliminary definition of the goals for the improvement programme is setting the priorities of the process improvement objectives.

The improvement goals direct the choice of the processes to be assessed, the definition of improvement targets and ultimately the identification of the most effective improvement actions.

Once the analysis of the organization's needs and business goals has been completed, it is essential to build executive awareness of the necessity for a process improvement programme. This requires both managerial and financial commitments. The objectives of such a process improvement programme should be clearly stated and understood, and expressed using measurable process goals. The process improvement programme should form part of the organization’s overall strategic business plan.

The executive decision to undertake improvement, together with the identification of a preliminary process improvement programme budget and the main process improvement priorities, enable the improvement process to progress through the following steps:

- initiate process improvement (see 5.2);
- carry out a software process assessment in the sectors where improvement is believed to be beneficial (see 5.3);
- complete the process improvement programme plan with the action plan resulting from analysis of assessment results (see 5.4);
- implement improvements according to detailed process improvement project plans (see 5.5);
- confirm the improvements (see 5.6);
- sustain the improvement gains by maintaining the new, improved level of performance until stability has been reached (see 5.7);
- monitor performance to continue the process improvement programme (see 3.8) comparing results against the measurable goals of the process improvement programme plan developed in 5.2 and 5.4.

5.2 Initiate process improvement

The process improvement programme should be considered as a project in its own right, and planned and managed accordingly. A process improvement programme plan should be produced at the beginning of the programme and subsequently used to monitor progress. The plan should include the relevant background and history and the current status, if possible expressed in specific, numerical terms. The improvement goals derived from the organization's needs and business goals provide the main requirements for the plan. The plan should include a preliminary identification of the improvement scope in terms of both the organizational boundaries for the improvement programme and the processes to be improved.

A plan should be produced at the beginning of the programme, and subsequently used to monitor progress. The plan should cover all the process improvement steps, although initially the plan may only give outline indications of the latter stages. It is important to ensure that key roles are clearly identified, that adequate resources are allocated, that appropriate milestones and review points are established, and that all risks associated with the plan are identified and documented in the plan. The plan should also include activities to keep all those affected by the improvement informed of progress.

5.3 Prepare for and conduct a process assessment

5.3.1 Prepare assessment input

This step gives guidance on how to define the assessment inputs needed to carry out a software process assessment as described in part 4 of this International Standard.

5.3.1.1 Sponsor and owner

Preparation for an assessment begins with the identification of a sponsor and owner for the assessment. The sponsor is a senior manager, who is committed to the process improvement programme, requires the assessment to be performed, and provides resources for it. The sponsor ensures that the process assessment inputs (purpose, scope, constraints and responsibilities) are adequately defined so as to meet the needs of the process improvement programme. It is likely that the assistance and advice of a qualified assessor will be of help to the sponsor in formulating these inputs.

The owner of the assessment is a management role within the organization that has the authority to ensure that the assessment can be carried out effectively, and takes ownership of the assessment output. These roles of sponsor and owner may be vested in a single individual, but both must be committed to the concepts of process improvement through process assessment.

5.3.1.2 Qualified assessor

The responsibility for ensuring that an assessment is conducted in accordance with the provisions of this International Standard is vested in the qualified assessor who leads, or is part of, the assessment team. A key factor in selecting an assessment team, and particularly the qualified assessor, is credibility with the management and staff of the organizational unit. Depending on the local circumstances, a qualified assessor drawn from outside the organizational unit may appear to be more credible on account of a more independent viewpoint.

5.3.1.3 Assessment responsibilities

The sponsor, the owner, and the qualified assessor have specific responsibilities with respect to a process assessment as described in Parts 3 and 4.

The sponsor should ensure that the assessors may access all the relevant information to make the assessment successful, and in particular that they are aware of the improvement programme goals. The owner should ensure that all the process instances are accessible to the assessors.

5.3.1.4 Assessment purpose

The overall purpose of the assessment is to provide information about the process capability of the organizational unit in the form of the assessment results. The statement of assessment purpose will guide the assessment team during the conduct of the assessment, particularly with regard to the amount, nature and content of the information they should capture during the assessment to aid improvement. The purpose statement should make clear that the assessment is being done as part of a process improvement programme, and should contain clear descriptions of quality improvement goals (see 7.3.4 and 5.1) and specifically the goals whose attainment is expected to be dependent on the assessment results. All the information on the improvement background defined in the previous steps 5.1 and 5.2 should be made available.

5.3.1.5 Assessment scope

The assessment scope defines the boundaries for the assessment, both organizationally and in terms of the processes to be included.

From an improvement point of view the process improvement programme may address an entire organization, part of an organization, a single project, or even a part of a project. A process assessment, however, addresses an organizational unit with a coherent process context, particularly the application domain, size, criticality and complexity, and quality characteristics of its products or services. If the process improvement programme spans different organizational units with different types of operation, then each of them should be assessed separately.

The broader the scope of an assessment, the greater the assessment effort needed to arrive at a representative result. Therefore, the sponsor may wish to limit the scope to those processes that are expected to have the greatest potential for improvement. Priority should be given to the processes or process categories that directly affect improvement goals derived from the organization’s needs and business goals. The scope statement should include any assumptions or expectations in the process improvement programme plan about the strengths and weaknesses of process categories, single processes, or practices.

The selection of process instances should be done by picking out a representative sample that provides a reliable picture of the current status of the software process. Therefore, it is useful to assess projects that are considered both as the worst examples and as the best examples of the organization's current process performance. In this way the variability between the worst and the best cases of the organization can be found and taken into account in the improvement.

The scope is defined initially in terms of the processes operated and understood by the organizational unit. Ultimately, however, the organizational processes need to be mapped to the processes in the process model in part 2 of this International Standard to enable the assessment team to conduct the assessment. The mapping may be undertaken by the assessment sponsor, or by the assessment team. If an organizational process cannot be mapped on one of the processes in the process model, it should be defined as an extended process according to the requirements in part 2 of this International Standard.

In addition to specifying the processes to be assessed, the scope should clearly identify and justify which process instances (projects) are to be assessed, the organizational unit and its characteristics, and the product or service characteristics (process context). The product or service characteristics, in particular, provide the context within which the assessment team will judge the adequacy of the implemented practices and affect the validity of comparisons with other industry benchmarks.

5.3.1.6 Assessment Constraints

The sponsor may wish to restrict the freedom of choice of the assessment team in selecting process instances for assessment, in selecting individuals for interview, and in how information may be used. Any restrictions, which may be positive or negative (for example, to include or exclude process instances or individuals), are documented as the assessment constraints.

Similarly, the sponsor may wish to place constraints on which individuals are to be interviewed by the assessment team. For process improvement, the process owners of each assessed process should always be involved in the assessment.

Ownership of results and confidentiality of information may be an issue in many organizations. Whatever the situation, the assessment constraints should include a clear statement about how the information and assessment results may be used, and hence the confidentiality arrangements that apply.

5.3.2 Conduct a process assessment

A process assessment, as described in the part 4 of this International Standard, is initiated using the assessment input prepared as described in 5.3.1. The assessment delivers its results as the assessment output which consists of:

- the current process profile (i.e. the practice adequacy ratings and process capability level ratings for each assessed process);
- the assessment record (i.e. any information which is pertinent to reviewing the results of the assessment) including, in particular, the set of base practice ratings for each process assessed and any supporting evidence for the ratings such as assessment indicators;
- additional information for process improvement.

The additional information may include the recognition of:

- existing best practice that could be adopted and institutionalized in the organization;
- experiences about the previous adoption of specific methods or tools in the organization;
- cultural issues that might either foster or jeopardize improvement initiatives;
- organizational issues that might foil the potential benefits of process improvement;
- training needs.

The assessment output will show not only which processes are at a relatively low capability level, but also the degree of variability across the organizational unit. Both aspects are useful in determining priorities for an improvement programme.

The assessment record should be archived with the results, both as an aid to subsequent understanding, and in case there is a later need to verify that the assessment was carried out in accordance with the provisions of this International Standard.

5.4 Analyse assessment output and derive action plan

Information collected during the assessment, in particular the capability level ratings, the generic practice ratings and the base practice ratings, is analysed in the light of the organization's needs to:

- identify areas for improvement;
- set qualitative software process goals, and quantitative improvement targets;
- derive a process improvement action plan, and integrate it with a revised process improvement programme plan.

Management should approve the areas for improvement, the goals and targets, the action plan and the revised process improvement programme plan, thereby committing the organization to undertake the planned improvements. The decision should be communicated clearly to all affected staff.

5.4.1 Identify and prioritize improvement areas

Improvement areas should be identified and prioritized based on a number of factors as outlined in figure 3. The factors in detail are:

- the assessment output, which shows weak and strong areas of the assessed processes and process instances;
- the organization's needs, which provide general improvement goals (see 5.1) to be achieved through the improvement programme;
- effectiveness measures, which, if already in place, identify improvement priorities for the organization generally related to the improvement drivers described in 4.1;
- industry norms and benchmarks that provide a basic comparison framework for assessment results;
- risks related to either not achieving the stated improvement goals or failure of improvement actions.

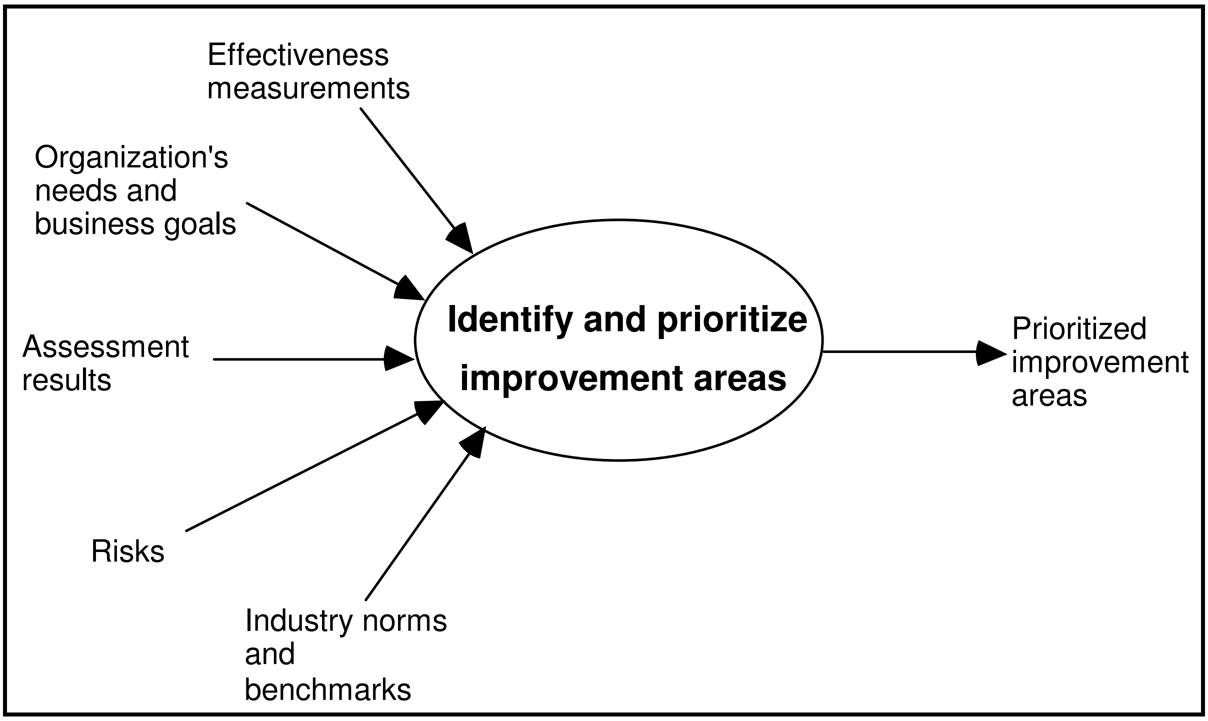

Effectiveness measurements
Organization's needs and business goals
Assessment results
Risks
Industry norms and benchmarks
Identify and prioritize improvement areas
Prioritized improvement areas

Figure 3 – Identifying and prioritizing the improvement areas

5.4.1.1 Analyse assessment results

Analysis of the assessment results provides information about the current strengths and weaknesses of the process and indicates opportunities for improvement. The analysis may be performed using process capability level ratings and practice ratings together with the assessment context information.

The strong points are identified as the processes or process instances with the highest process capability level ratings. Strengths may support process improvement as follows:

- strong process instances may provide experience of good practices that could be adopted and institutionalized in the organization;
- processes with the highest process capability level rating within a process category or a set of interrelated processes may show an opportunity for improving the effectiveness of the rest of the process category or set of interrelated processes.

The weaknesses may be:

- process instances and processes with low process capability level ratings;
- processes with missing base practices that are needed to enable the process to achieve a process purpose aligned with a specific need of the organization;
- incomplete coverage of generic practices within common features and of common features within capability levels that are necessary to achieve a specific need;
- low scores for generic practices and common features across assessed processes that may indicate weakness in specific process categories (for example low scores at process capability levels 2 and 3 may show weaknesses in the PRO and SUP process categories).

The variability of process capability levels ratings across process instances should be evaluated in the light of the process context to identify specific improvement actions.

Similarly, the process capability level ratings of related processes should be compared. Improvement actions may be identified to correct any imbalance.

5.4.1.2 Analyse the organization's needs and improvement goals

The processes and their relationships should be analysed in order to evaluate which processes have direct impact on the improvement goals identified in the process improvement programme plan. Specific relationships between single processes should be considered in order to identify processes which should be addressed together to fulfil certain improvement goals. In this way, a priority list of processes to be improved may be derived. The processes in this list with low process capability level ratings may provide the best opportunity for improvement.

5.4.1.3 Analyse effectiveness measurements

Organizations with previous experience in process improvement may already have effectiveness measurement in place. Where these are related to the existing organization's needs and derived improvement goals, it may be worth analysing the current measurements to get a better understanding of what improvement is needed (see 7.3).

5.4.1.4 Analyse the risks in not achieving improvement goals

The impact on the organization's needs and business goals of failing to achieve improvement goals should be evaluated in order to understand the urgency and to set the priority of improvement initiatives. The risk analysis will provide hints on prioritizing improvement areas and will help to gain commitment and funding to carry out the improvement actions as needed.

5.4.1.5 Analyse risks of improvement action failure

The potential risks of failure of an improvement action should be analysed in order to support the definition of improvement priorities and to assure the proper commitment and organizational support. These risks may be related to specific organizational issues, to specific time frames, to cultural issues, to commitment or to funding issues. More precisely the risks may derive from:

- existing schedule constraints;
- existing psychological or cultural barriers, possibly derived from previous experiences;
- organizational issues preventing the successful execution of improvement actions.

The risk mitigation strategy may be influenced by hints from the assessment on where to find the main improvement potential. This potential may be:

- in organizational areas or process instances with positive capability to change;
- synergistic with existing contractual commitments or well-understood customer expectations.

5.4.1.6 List improvement areas

A prioritized list of improvement areas may be provided as a combined result of analysing all the factors listed above. The selected improvement areas define the scope of the improvement actions to be identified. The scope could include:

- processes to be included;
- organizational boundaries for improvement;
- process instances to be either included or excluded in the improvement initiative.

5.4.2 Define specific improvement goals and set targets

Targets for improvement should be quantified for each priority area. These may be either target values for process effectiveness, or target capability profiles, or combinations of the two. They should be set in the light of the organization's needs, using the approach outlined in section 7.3. This will typically require the iteration of a number of steps until a set of targets has been identified which meets the organization's needs, which can be objectively measured, and which can reasonably be achieved. The key steps are:

- to define qualitative goals for each priority area for improvement;
- to devise suitable metrics to measure achievement of these goals;
- to set appropriate target values for these metrics, taking due account of risks.

More mature organizations, and those which have already made use of this methodology during previous improvement cycles, may already have established goals, metrics and targets. These should be reviewed for their continuing suitability and adjusted as appropriate in the light of a current assessment of the organization's needs.

When setting capability levels as targets for processes, the following points should be considered:

- it is desirable for related processes to be at the same capability level, unless there are overriding considerations;
- it is generally unrealistic to seek to increase the capability of a process by more than one level in a single cycle of improvement, since each level builds on the capabilities of the ones below it.

5.4.3 Derive action plan

A set of actions to improve software processes should be developed. Such actions, taken together, should meet the process goals and quantified targets set in the previous step (see 7.3). Possible improvement actions may interact, support each other or work against each other: such effects should be analysed. Care should be taken to select a set of actions which support each other in achieving the complete set of goals and targets. It is also desirable to include some improvement actions which yield clear short term benefits, particularly if the organization is new to process improvement, in order to encourage acceptance of the process improvement programme.

When carrying out this task the organization should:

- evaluate a number of scenarios to arrive at a set of actions which best meets the organization's needs;
- use the base practices and generic practices defined in the process model as a specification of improvement actions;
- define success criteria for each action and how progress will be measured (the metrics used to set the targets provide suitable measurements);
- evaluate initial estimates of costs and benefits, schedule and risks for the proposed actions;
- identify responsibilities for the actions, and agree the responsibilities with those affected by the actions;
- identify recruitment and training needs.

The set of agreed actions should be documented as a process improvement action plan containing the following information:

- improvement actions with associated process goals and improvement targets;
- responsibilities for actions;
- initial estimates of costs, benefits and schedule (detailed estimates are made at 5.5.2 below);
- risks to products and to the organization if actions are taken or not taken and the implications for timing change appropriately.

The process improvement action plan is a tactical plan, developed to meet the organization's needs, which should be integrated with the process improvement programme plan (originally developed at a strategic level during step 5.2). The process improvement programme plan should be reviewed at this point and updated if necessary. Both the process improvement action plan and the updated process improvement programme plan should be approved by management to ensure that the organization is committed to implementing them.

5.5 Implement improvements

The process improvement action plan is implemented in order to improve the organization's software process. Implementation may be simple or complex depending on the contents of the action plan and the characteristics of the organization.

In general several process improvement projects will be initiated, each concerned with implementing one or more process improvement actions. Such projects will often not only cover initial implementation of improvements as described in this section, but the subsequent steps 5.6 and 5.7. Four main tasks are involved in each process improvement project:

- selecting the operational approach to implementation;
- preparing and agreeing the process improvement project plan (a detailed implementation plan);
- implementing the process improvement actions according to the process improvement project plan;
- monitoring the process improvement project.

5.5.1 Operational approach to implementation

Where there are alternative operational approaches to implementation, they should be evaluated and the most suitable selected. For instance, it may be possible to implement a given action either in small steps through piloting in a selected unit, or throughout the whole organization at the same time, or somewhere between these two extremes. Among the factors to consider are costs, time scales, and risks.

5.5.2 Detailed implementation planning

A detailed implementation plan should be developed, including the following:

- the objectives of the process improvement project;
- a description of the approach to implementation;
- the organization and responsibilities;
- the time schedule and resources;
- risk management, including assessment, monitoring and mitigation;
- monitoring policy;
- specification of success criteria, including process goals and improvement targets.

The process improvement project may need to carry out a deeper analysis of improvement opportunities than that already carried out in step 5.4. Where appropriate, the plan should include:

- further collection and analysis of data to establish the underlying causes of unsatisfactory current effectiveness measurements and process profiles;
- evaluation of alternative proposals for eliminating the causes, including analysis of costs and benefits;

- arrangements to capture cost and resource usage data, for instance if it is desired to carry out cost-benefit analysis.

Those implementing the actions and affected by them should be involved, or be consulted, while developing the plan and in evaluating alternatives, in order both to draw on their expertise and enlist their co-operation.

5.5.3 Implementing improvement actions

It is critical for successful improvement that due account is taken of human and cultural factors. In particular the following should be considered:

- how management can give support and leadership (see 6.1);
- what changes may be needed in values, attitudes and behaviour (see 6.2);
- how to establish commitment to goals and targets (see 6.3);
- how to foster open communication and teamwork, including implications for organizational structures and reporting lines (see 6.4);
- whether changes are needed to recognition and reward systems (see 6.5);
- what education and training is required (see 6.6).

5.5.4 Monitoring the process improvement project

The process improvement project should be monitored by the organization's management against the process improvement project plan in order to:

- ensure tasks progress as planned, and initiate appropriate corrective action if they do not;
- check that achievement of the planned goals and targets continues to be both realistic and relevant to the organization's needs;
- gather data on effort and resources expended, in order to improve estimates for future process improvement projects;
- evaluate the impacts of the implemented improvement actions on the process capability level ratings.

Records should be kept for use both to confirm the improvements (see 5.6) , and to improve the process of process improvement (see process ORG.3 *Improve the process* in part 2 of this International Standard).

5.6 Confirm improvements

When the process improvement project has been completed, the organization should:

- confirm that the planned goals and targets have been achieved and that the expected benefits have been delivered;
- confirm that the desired organizational culture has been established;
- re-evaluate risks associated with the improved process;
- re-evaluate costs and benefits.

Management should be involved both to approve the results and to evaluate whether the organization's needs have been met. If, after improvement actions have been taken, measurements show that process goals and improvement targets have not been achieved, it may by desirable to redefine the process improvement project or activity by returning to an appropriate earlier step.

5.6.1 Improvement targets

Current measurements of process effectiveness should be used to confirm achievement of process effectiveness targets (see 7.3). The possibility of having introduced desirable or undesirable side effects should be investigated.

A further process assessment should be used to confirm achievement of targets expressed as process capability levels. The scope of this re-assessment should be related to the scope of the initial assessment. The scope might cover only the processes affected by the improvement actions, particularly where these had a narrow focus. Where several improvement projects were undertaken, however, consideration should be given to a re-assessment of wider scope to check for potential side-effects arising from the parallel improvement actions.

5.6.2 Organizational culture

The effect of the improvements on organizational culture should be reviewed to establish that desired changes have taken place without undesirable side-effects.

5.6.3 Re-evaluate risks

The organization should re-evaluate the risks of using the improved process to confirm that they remain acceptable, and to determine what further actions are required if they are not.

5.6.4 Re-evaluate cost-benefit

The costs and benefits of the improvements may be re-evaluated and compared with earlier estimates made at step 5.4 and 5.5. These results are useful to support planning of subsequent improvement actions.

5.7 Sustain improvement gains

After improvement has been confirmed, the software process needs to be sustained at the new level of performance. The improved process should be used by all those for whom it is applicable. This requires management to monitor institutionalization of the improved process, and to give encouragement when necessary. Responsibilities for monitoring should be defined, as well as how this will be done, for instance by using appropriate effectiveness measurements (see section 7.3).

If an improved process has been piloted in a restricted area or on a specific project or group of projects, it should be deployed across all areas or projects in the organization where it is applicable. This deployment should be properly planned and the necessary resources assigned to it. The plan should be documented as part of the process improvement project plan or the process improvement programme plan as appropriate. Consideration should be given to:

- who is affected;
- how to communicate both the changed process and the benefits expected from it (note: changes should be properly documented and approved);
- what education and training are necessary;
- when to introduce changes to the different areas, taking business needs into account;
- how to ensure that the changes have been made (for instance by conducting audits);
- how to ensure that the improved process performs as expected (for instance by monitoring capability levels and/or effectiveness measures - see section 7.3).

5.8 Monitor performance

The performance of the organization's software process should be continuously monitored, and new process improvement projects should be selected and implemented as part of a continuing process improvement programme, since additional improvements are always possible.

5.8.1 Monitoring performance of the software process

The performance of the organization's software process should be monitored as it evolves over time. The effectiveness and conformance measures used for this should be chosen to suit the organization's needs and business goals (see 7.3). Management should regularly review their continuing suitability. The risks to the organization and its products from using the software process should also be monitored and action taken as risks materialize or become unacceptable.

5.8.2 Reviewing the process improvement programme

The process improvement programme should be reviewed regularly by management to ensure that:

- both the improvement programme and individual improvement projects, including their goals and targets, remain appropriate to the organization's needs;

- further improvement projects are initiated when and where appropriate as previous improvement projects have been completed;
- the process improvement process is itself improved in the light of experience;
- continuous improvement becomes and remains a feature of the organization's values, attitudes and behaviour.

Further process assessments can be an important component of the continuing improvement programme, for instance in the following circumstances:

- where a long term goal to achieve higher capability levels is to be approached by stages;
- when changing organizational needs indicate a requirement to achieve higher capability levels;
- when there is a need to give a fresh impetus to improvement.

The extent to which improved processes have been institutionalized should be considered before scheduling further process assessments. It may be more cost-effective to delay assessing a process until improvements have been fully deployed, rather than expend resources assessing a process which is in transition, when the results can be difficult to interpret.

6 Cultural issues

Software process improvement should be strongly supported by leadership, communication and motivation throughout the whole organization. Improvement actions can only be carried out efficiently if the appropriate cultural issues are acknowledged at all levels. Moreover, major problems found in software processes often arise from cultural issues. Consequently, cultural issues should be one of the factors considered in prioritizing improvement actions.

6.1 Management responsibility and leadership

The successful use of this International Standard to improve software processes requires the same high degree of management leadership and commitment as any other approach to process improvement and organizational change. The responsibility for leadership and for creating the environment for continuous process improvement belongs to all levels of management, but particularly to the highest. Senior management should be aware of how the success of the organization depends on quality software and the ability to improve software processes.

The commitment of middle management may pose a particular risk to successful process improvement, particularly in less mature organizations. Largely concerned with meeting project commitments in the short term, middle management may pay little attention to process improvement benefits, which tend to be medium to long term, and often resent diverting scarce project resources to process improvement projects. A mitigation strategy to counter the risk is to ensure that senior management is committed to the costs and impact of process assessment activities and improvement actions on the projects to which they are applied.

Process assessment can identify areas of weakness in management responsibility and leadership as being a risk to the software process in general. An appropriate response is to raise the awareness amongst senior and middle managers of importance of software and software process improvement, possibly through training initiatives. Furthermore, analysis may suggest the need to change the role of middle management. Instilling teamwork principles and placing the emphasis on communication could change the relationship between middle managers and development teams from enforcement to facilitation, and from imposing ideas to helping teams develop their own ideas. The management approach should take account of the specific characteristics of software staff and software development work. Software production, requiring educated staff and high intellectual engagement, provides better results in a co-operative environment.

6.2 Values, attitudes and behaviour

Effective process improvement often implies a new set of shared values, attitudes and behaviour, which may include:

- focusing attention on both external and internal customer satisfaction;
- targeting employee satisfaction by establishing an appropriate recognition system (see 6.5);
- involving the entire software supply chain in process improvement, from suppliers to customers;
- demonstrating management commitment, leadership and involvement by communicating purpose and goals;
- emphasizing process improvement as a part of everyone's job and helping everybody to gain an understanding of how individual activities can be beneficially channelled towards the common goals of the team;
- considering quality, cost and time scale goals as priorities to improve processes;
- establishing open communication with access to data and information;
- promoting teamwork and respect for the individual;
- objectively measuring process performance and making decisions based on realistic metrics agreed by all parties in the organization (see 7.3).

Process assessment can help an organization to understand which changes are necessary in values, attitudes and behaviour. If current values, attitudes and behaviour do not contribute to meeting the organization's needs, the process improvement programme should include appropriate cultural change.

6.3 Process improvement goals and motivation

The organization's needs should be analysed to yield goals for improving the software process (see section 7.3). Targets should be set either in terms of use of good software engineering practice (increasing process capability levels), or in terms of the effectiveness with which the process meets the organization's needs (process effectiveness measures), or a combination of both. Less mature organizations are likely to emphasize the former and more mature organizations the latter. Industry benchmarks can be used as a reference to set appropriate improvement goals.

Staff motivation to achieve these goals will be strengthened if progress is made visible through regular measurement. Furthermore, the goals have to be understandable, challenging and pertinent. Strategies to achieve improvement goals should be understood and agreed to by everyone. Goals should be reviewed regularly and must reflect any change in the organization's needs.

6.4 Communication and teamwork

When analysing assessment results it is important to look for organizational and personal barriers that are causing a lack of communication and teamwork, thereby interfering with the effectiveness and efficiency of the software process. Communication and teamwork require trust and skills. Good teamwork skills improve the ability to perform activities with the high degree of parallel work typical of software projects. Training should be considered as a means of improving the quality and effectiveness of teamwork skills.

Before conducting an assessment, agreement should be reached over ownership and confidentiality of the results and other information gathered during the assessment. This will help to build the necessary trust for effective process improvement. It is important that the individuals and groups responsible for the processes which are being assessed understand that the objective is to improve the processes, and not to assign blame to individuals. It is also necessary to communicate and discuss the assessment findings with the assessees before finalizing any recommendations. Unless this is done, individuals or groups may reject the findings, and may resist changes arising from the findings, thereby jeopardizing the outcome.

6.5 Recognition

The recognition process and individual reward system may help to encourage attitudes and behaviour necessary for successful process improvement (see 6.2). The definition of an appropriate recognition and reward system, consistent with the effort needed to achieve the improvement goals, should therefore be considered when planning improvement actions (see 5.4). The reward system should be designed in such a manner that it recognizes group performance and teamwork and avoids promoting destructive internal competition.

6.6 Education and training

On-going education and training are essential for everyone. Education and training programmes are important in creating and maintaining an environment where process improvement can flourish.

The effectiveness of education and training should be regularly assessed. Training separated from the use of the newly acquired skills is rarely effective. The assessment results include practice adequacy scores related to the extent to which staff have received suitable training in the processes they use, which should be taken into account when planning improvement actions (see generic practice 2.1.5 *Ensure training* and process ORG.4 *Training* in the process model in part 2 of this International Standard).

Training in process improvement concepts, specifically, will increase the organization's readiness for process improvement (see generic practice 2.1.5 applied to process ORG.3 *Improve the process*). Important concepts that should be covered include process and quality concepts, process improvement concepts, process management skills, tools and techniques for process improvement, cultural change skills and supporting skills.

Process assessment concepts should be explained to all levels of the organization (from management to staff) being assessed. The assessors should fulfil the basic educational requirements and receive training as defined in part 6 of this International Standard. Untrained assessors are less likely to produce objective, consistent and reliable results on which to base a successful process improvement programme.

7 Management

The full potential of software process improvement can only be realized when applied and co-ordinated within a structured framework. This requires software process improvement to be organized, planned, and measured, and all process improvement activities to be subject to management review. It is very unlikely that permanent long-term changes will result without consideration of the managerial implications of software process improvement.

7.1 Organizing for process improvement

For software process improvement to be performed effectively, the entire organization should be involved. The general organizational principles for quality improvement described in ISO 9004-4 apply to software process improvement.

Responsibilities for process improvement activities are divided amongst senior management, the process improvement programme, process improvement projects, process owners, and the organizational units. The differing responsibilities are reviewed below.

7.1.1 Senior management

Senior management involvement is needed to translate the broad understanding of the needs and business goals of the organization into the investment in software process improvement, and to provide the necessary commitment and decision making.

The responsibilities of senior management include:

- defining the mission, objectives and needs of the organization (input to step 5.1);
- preparing strategic and business plans, including software process improvement goals, resource estimates and time scales;
- approving the process improvement programme plan (see 5.2 and 5.4);
- approving responsibilities for process improvement projects;
- ensuring that the appropriate resources to support process improvement are provided;
- monitoring improvement results to ensure targets have been met (see 5.6);
- initiating and supporting activities aimed at institutionalizing improved processes (see 5.7);
- regularly reviewing the overall process improvement programme to ensure its continued appropriateness to the business (see 5.8);
- fostering changes in values, attitudes, and behaviour to support software process improvement (see clause 6).

7.1.2 Process improvement programme management

Processes are generally intersective in nature and flow across organizational boundaries. Therefore, a process improvement programme must have an overall process view, which covers entire processes to be improved. Managing the programme can be done by assigning software process improvement responsibilities across these boundaries. Typically this can be organized by setting up a cross functional action team or establishing an organizationally independent software engineering process group to deal with all software process issues.

Process improvement programme management responsibilities across organizational boundaries include:

- establishing systematic software process measurements including both software process assessments (see 5.3), and effectiveness measurements;
- evaluating measurement results;
- setting software process improvement targets, and agreeing these with the process owner and the organizational units involved (see 5.4);
- identifying improvement actions and gaining the agreement of the process owner and the organizational units involved (see 5.4);
- naming the project leader for each process improvement project;
- participating in development of process improvement project plans with those responsible for each improvement project, the process owners and the organizational units involved (see 5.5);
- monitoring the progress of improvement projects towards their targets (see 5.5);
- supporting continuation of software process improvement (see 5.8);
- reviewing the software process improvement process itself in the light of the lessons learnt (see 5.8).

7.1.3 Process improvement project management

Process improvement actions defined by the process improvement programme management and approved by the process owners will be managed as improvement projects by applying a general project management approach. Improvement projects are managed by a project leader named by the process improvement programme management.

Process improvement project management responsibilities are:

- preparing and updating the process improvement project plan in consultation with the owner of the process to be improved and representatives of the organizational units involved (5.5);
- obtaining the approval of the process owners for the process improvement project plan and the changes to the processes (5.5);
- acquiring sufficient physical and human resources for implementing the process improvement project plan (5.5);
- organizing the implementation project in consultation with the owners of the process to be improved and representatives of the organizational units involved (5.5);

- monitoring and controlling the implementation process (5.5);
- reporting the status of the implementation for both the process owners and senior management (5.5).

7.1.4 Responsibilities of the process owners

Each process should have a process owner who is responsible for the whole improvement activity inside the organizational unit. Software process improvement aims to increase satisfaction of both external customers and internal personnel, therefore their viewpoints should also be taken into account by the process owner. Awareness of software process improvement issues as well as collaborative communication are required at all organizational levels (see clause 6): this is one important part of the process owner's responsibility. Process owners, as representatives of the customers of the improvements, should be involved in the whole improvement cycle.

Responsibilities of a process owner include:

- providing information and measurements on the current process status (see 5.3);
- identifying what external customers and internal personnel need from their direct supplier,
- promoting awareness and collaborative communication between internal users and external customers about the improvement action;
- support for planning of the process improvement programme and process improvement projects (see 5.4 and 5.5);
- approving the improvement project plans;
- participating in improvement activities (see 5.5);
- monitoring and confirming the improvement results (see 5.5).

7.1.5 Role of the organizational unit

The processes and practices within an organizational unit are the targets of software process improvement. The staff of the organizational unit will be affected by the changes and so it is important to engage them in the improvement activities. Their opinions and viewpoints should be considered when planning the improvements and they can provide useful feedback on the results of the improvement.

Responsibilities within the organizational units involved in software process improvement include:

- collecting measurements on the practices/processes instantiated within the organizational unit (this includes participation in assessments (see 5.3));
- implementing improvement actions on the organizational unit’s processes (see 5.5);
- monitoring progress of improvement actions (see 5.5).

7.2 Planning for process improvement

The planning of a software process improvement programme is an iterative activity that extends for the entire life of the programme, starting from the definition of improvement goals and continuing through all phases of the improvement cycle.

Three main levels of planning should be performed, resulting in the following documents:

- business plan;
- process improvement programme plan;
- process improvement project plan.

7.2.1 Business plan

A software process improvement programme should meet the organization's needs and business goals, usually by aiming at improving quality, increasing cost control, reducing time to delivery, reducing risk, or some combination of these. Management should set overall improvement goals which may impact the whole software process. The organization's business plan should include software process improvement goals as well as resource forecasts and time constraints for the implementation of process improvement. The business plan should include an overall evaluation of the risks to the organization if the current software process is not improved. Such evaluation may cause the organization to undertake the process improvement programme. The reasons for including this preliminary improvement planning in the business plan are firstly to align improvement with the organization's needs and business goals, secondly to ensure the availability of required resources, and thirdly to ensure the success of the organization's improvement strategy. At this level, process improvement programme planning is the responsibility of the organization's senior management.

Software process improvement goals and constraints (time and resources) stated in the business plan should be considered when:

- examining the organizational needs (see 5.1);
- defining the preliminary process improvement programme plan (see 5.2);
- completing the process improvement programme plan based on the assessment findings (see 5.4);
- confirming the improvement (see 5.6);
- monitoring process performance when an improvement cycle has been completed (see 5.8).

7.2.2 Process improvement programme plan

The process improvement programme plan is the document that addresses the entire process improvement programme defined to meet the goals stated in the business plan. It controls continuous improvement activities throughout the organization, and can be maintained through more than one improvement cycle. The process improvement programme plan will be owned by the process improvement programme management (7.1.2).

Due to its nature, the process improvement programme plan is an evolving document that is updated through three main stages as:

- a preliminary process improvement programme plan;
- the complete process improvement programme plan itself;
- the ongoing process improvement programme plan.

When a process improvement programme is initiated, a preliminary identification of the improvement areas, actions and related risks, together with a preliminary budget and time schedule, is defined in a preliminary version of the process improvement programme plan (as a result of step 5.2). This information will be revised after the process assessment, based on the assessment findings (see 5.4). The plan is then completed with the target profile, gap analysis information and possibly with current and target effectiveness measurements. Resource budget and schedule are revised. The overall organization for the process improvement programme is defined. The plan will also include a risk evaluation with descriptions of the risks if the process improvement programme is not undertaken, the risks if it is, and mitigation strategies to be adopted. Finally the action plan derived from analysis of assessment results (see 5.4.3) is attached to complete the process improvement programme plan.

The process improvement programme plan is then used for monitoring the implementation of the improvement actions (see 5.5). It is continuously updated to reflect the actual status of the improvement actions. If the improvement targets are revised, the process improvement programme plan should be updated accordingly.

The process improvement programme plan can be used as an input to a process capability determination activity (see part 8 of this International Standard). Based on the target capability of the process capability determination and the current capability obtained from the assessment, the process improvement programme plan may be tuned in order to meet the contract requirements. The process improvement programme plan can be used by the customer to establish a level of confidence in the supplier's processes. It can also be used by the supplier organization to identify the capability to be offered to the customer (the “proposed capability” described in part 8 of this International Standard).

7.2.3 Process improvement project plan

Improvement actions defined in the process improvement programme plan will be implemented as projects. When the improvement actions are complex and involve more organizational units, they might need to be implemented as several separate improvement projects. Each identified improvement project will be planned and the results are documented as process improvement project plans.

Typically process improvement projects are unique and innovative efforts for the organization that may require new types of resources and the adoption of new viewpoints. The human inertia of the organization may render the project more difficult to carry out. Therefore, process improvement projects often involve higher risks than projects that are repetitive in nature. As risks are situational and exist in different forms in different improvement steps or activities, a careful and detailed risk analysis should be performed in each step of each project.

Risk analysis at this stage includes reviewing and updating the risk descriptions and mitigation strategies included in the process improvement programme plan. All process improvement project plans should be prepared in consultation with appropriate people at the appropriate level in the organization, and with suppliers and customers of the organization (see section 7.1). Involving everyone greatly increases the opportunities for improvement and the chances of success.

Typically a process improvement project plan should contain:

- the detailed definition of the objectives and scope of the improvement action maintaining traceability with respect to the requirements defined in the process improvement programme plan;
- a detailed and concrete description of the improvement results to be attained in the project;
- a detailed time schedule and work breakdown structure;
- a detailed resource estimate;
- approval criteria for intermediate and final results of the improvement action.

7.3 Measuring process improvement

The role of measurements is crucial in software process improvement. Measurements are needed to show quantitatively the current status of processes and practices against a general understanding of software engineering best practices, and to show to what extent software processes are effective in achieving the organization's needs and business goals.

Best practices provide a general model of the software industry but cannot directly reflect specific characteristics of a single organization. Each organization has its own specific needs and business goals that will influence the selection of industry best practice for use in improvement.

Process effectiveness measures, and base and generic practice adequacy ratings address different facets of understanding of the process. Both may be considered from the organization’s point of view and used together in process improvement.

Although base practices may be rated using either an adequacy or existence scale, it is recommended that base practice adequacy rather than base practice existence be used for process improvement purposes.

7.3.1 Adequacy ratings

The primary output of a process assessment is the set of process profiles containing generic practice adequacy ratings and the assessment record which contains the ratings for the base practices. These ratings provide, at one level, a view of how the organizational unit’s processes conform to the model of good practice expressed in part 2 of this International Standard. However, the essence of the rating scheme is that the ratings represent more than just conformance: practices are rated for adequacy of implementation within the specific context in which the processes operate. The process context and other elements of the assessment input ensure that the assessment results are directly related to the organization.

The assessment results, therefore, provide immediate pointers to potential areas for improvement using the model in part 2 of this International Standard as the route map. Analysis of the results (step 5.4) yields a view of the strengths and weaknesses of the assessed processes. Where several process instances (projects) are assessed, the results can provide a view of the relative strengths, weaknesses and differences between projects, and can highlight systematic weaknesses.

The information gained from the assessment allows an organization to understand the current state of its processes and to set improvement targets expressed quantitatively as process capability levels to be attained for individual processes. This information may either strengthen the initial improvement goals defined for the process improvement programme, or cause them to be revised.

Once the improvement have been implemented (see 5.5), their effects may be confirmed through a reassessment (see 5.6).

7.3.2 Effectiveness measures

Even though an assessment provides adequacy ratings judged against the process context, it is important to recognize that the ratings are also against the high level and, of necessity, universal goals (process purposes) defined in the process model in part 2 of this International Standard. In practice, an organization's needs and business goals may dictate different process goals and priorities, and may lead to the identification of general improvement goals which cannot be immediately expressed as target process capability levels. This creates the need for a different kind of measure, a process effectiveness measure, that addresses organization-specific process goals.

Effectiveness measures address the extent to which the software process achieves goals derived directly from an analysis of the specific circumstances, needs, and business goals of the organization. Effectiveness measures support the choice of processes to be improved and the monitoring of improvement results.

Effectiveness measures are usually defined based on the characteristics of process output. As an example, the effectiveness of a planning process may be related to its efficiency, its accuracy, its reliability or its repeatability, or some combination of these. Since organizations have unique circumstances and needs, the choice of effectiveness measure will differ between different organizations. Whichever one is chosen, however, its value may be measured and compared to a target value or threshold as a means of measuring improvement.

7.3.3 How to use adequacy ratings and effectiveness measures

Adequacy ratings and effectiveness measures may be used together to support software process improvement. An improvement goal might be stated and effectiveness measures identified to quantify the goal. Adequacy ratings from an assessment may provide the analytical information to identify a number of areas for improvement which together would support the achievement of the stated goal.

For instance, considering example 1 in Annex A, it can be seen that the goal of obtaining a large market share for an innovative product might be achieved by reducing the time to market, or improving product quality or both. Reducing time to market can be achieved by establishing reuse in the software organization, and reinforcing configuration management and project management. Improving product quality could be achieved by improved testing and quality control, but possibly also by establishing a proper reuse strategy. Evaluation of the current process profile leads to an understanding of strengths and weaknesses and the choice of the most cost effective improvement action. The choice of improvement areas, priorities and actions is non-deterministic: many factors such as previous investments, existing culture, previous success or failures and so on should be taken into account.

7.3.4 Framework for measuring processes

By making use of both adequacy ratings and effectiveness measurements, the organization's management can ensure that investment in process improvement is as cost-effective as possible. Organizations are recommended to set software process targets expressed either in terms of use of recognized good software engineering practice (i.e. increased process capability level), or in terms of the effectiveness with which the process meets the organization's needs, or a combination of both. Less mature organizations are likely to emphasize the former and more mature organizations the latter.

Figure 4 illustrates a framework for measuring processes, which may be applied with both adequacy ratings and effectiveness measurements and in integrating them to fulfil the improvement purpose.

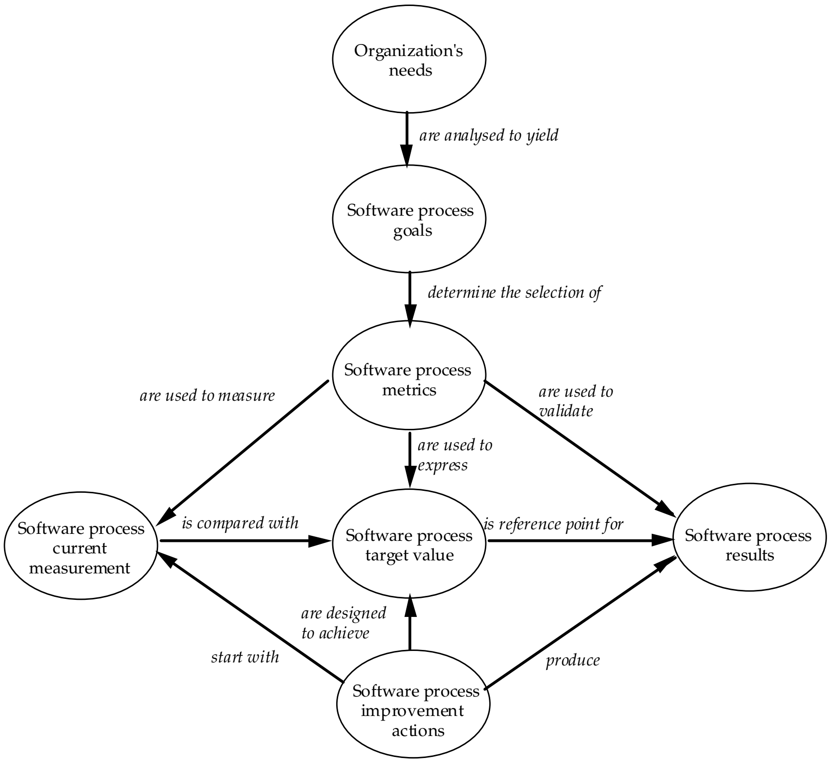


Figure 4 – Process measurement framework

The entities in this model are as follows:

- **software process goal**: an aim or objective of all or part of the software process. A software process goal should be defined, wherever possible, such that a single software process metric can be used to judge the degree to which the goal is achieved;
- **software process metric**: a quantitative measure of the degree to which a software process goal is met. Software process metrics include both effectiveness measures and practice adequacy or process capability level ratings as defined in part 4 of this International Standard;
- **software process target**: a desired value of a software process metric;
- **software process improvement action**: an action planned and executed to improve all or part of the software process. A software process improvement action can contribute to the achievement of more than one software process goal;
- **software process current measurement**: the value of a software process metric before implementing an improvement action;
- **software process improvement result**: the value of a software process metric after taking a software process improvement action.

When using the framework to measure adequacy ratings (see part 3 of this International Standard):

- the software process goal corresponds to the purpose statement of the process in part 2 of this International Standard;
- the software process metric corresponds to the rating mechanism defined in part 3 of this International Standard;
- the current measurements, targets and results are expressed as process capability level ratings.

When using this framework to measure effectiveness, the organization should:

- analyse its circumstances and needs, to identify and define a set of qualitative goals for its overall software process. Software process goals should be chosen to have the most direct and critical impact on meeting the organization's needs which is possible when the general goals are further decomposed and refined to correspond to single processes and individual or team responsibilities. Note that each goal applies to all or part of the organization's overall software process: this may be a single process in the process model in part 2 of this International Standard, or several related processes, or even a single practice;
- devise an appropriate way to measure whether each software process goal has been achieved. This is termed a software process metric. It may be necessary to iterate between the definition of goals and metrics until satisfactory metrics are found to measure the achievement of each qualitative goal;

- use the software process metrics to express a software process target for each defined software process goal. A software process target is a numerical value quantifying the extent to which the goal is achieved, and is chosen to meet the organization's needs. Its value may be better than the current value for the process (in which case it is a target for improvement), or it may be the same as the current value (in which case it is a steady-state constraint on improvement), or in some cases it may be worse than the current value due to previously incorrectly set targets;
- select software process improvement actions, which taken together as a package are designed to achieve the complete set of software process targets. In general, many actions may affect achievement of each target, and each action may affect the achievement of several targets;
- monitor the progress of the software process improvement actions by comparing current values of the metrics against the targets as appropriate;
- validate the software process improvement results (the metric values achieved after the actions have been completed) by comparing them against the targets.

The organization will normally use a mixture of effectiveness and adequacy measures to set targets for improvement. The adequacy measures will indicate the most promising improvement areas according to the distance between the organization's practices and the baseline; on the other hand, effectiveness measures of the processes will provide information about the efficacy of the organization's processes compared to the needs.

7.4 Reviewing process improvement activities

In general, regular reviews of improvement activities should be conducted at all levels of management to ensure that:

- the improvement organization is functioning effectively;
- plans for improvement are adequate and are being followed;
- measurements for effectiveness and for improvement are appropriate and adequate, and indicate satisfactory progress;
- software process assessments are conducted when it is appropriate to do so;
- a reasonable balance between opportunity for improvement and risk to the organization is maintained;
- the results of the review are fed into the next planning cycle.

Appropriate actions should be taken where any discrepancies have been identified.

Annex A (informative)

The application of the process measurement framework

Section 7.3 explains how to use process measures for improvement. The following table gives examples of how the application may be performed.

Organization needs	**Software process goals**	**Software process adequacy metrics**	**Software process effectiveness metrics**	**Software process targets**	**Software process results**
Example 1					
Obtain a large market share for an innovative product to secure additional financing.	1) Short software process time to market; 2) High software quality (i.e. high reliability); 3) Good Engineering and Support process categories and Enable reuse process.	Capability levels of processes in the ENG and SUP process categories and Enable Reuse process.		Raise level 1 ENG, SUP and Enable reuse processes to level 2 where majority of practices are satisfactory.	All processes at desired capability level.
Example 2					
Improve quality in order to increase market share.	ISO 9001 compliance	Capability level of processes which map to ISO 9001.		Capability levels necessary to achieve ISO 9001.	All processes at desired capability levels. Complianc e to ISO 9001 checklist.
Example 3					
Compete with fixed price contracts	accurate estimation		% of projects complete within 10% of plan	90% of projects complete within 10% of plan	67% of projects complete within 10% of plan
	achieve process purpose for process Establish Project Plan (PRO.2)	capability level of PRO.2		PRO.2 at Level 2	PRO.2 at Level 2

Annex B (informative)

The application of the improvement methodology

B.1 Introduction

The example below follows the 8-step model described in clause 5, and shows how to use the process measurement framework in figure 4 as part of this model. Of necessity, some of the detail is omitted, but the principal steps of the process and measurement should be clear. The example describes the whole process, and includes discussion of the process measurement framework within the description of the process steps.

B.2 Example: MovieViews' drive for higher quality

B.2.1 Step 1: Examine the needs of the organization (see 5.1)

MovieViews is a medium-sized company which produces interactive multimedia software for PCs. Unfortunately, even though quality assurance personnel provide independent review of all products, defective products have been distributed and unhappy customers have called the company to report problems. The company needs to reduce and finally eliminate defects which affect customer satisfaction.

The new Managing Director, Mandé Taylor, realized the employees had no sense of personal contribution to the quality of the company's products or understanding of the company vision. As a first step, she wrote a document describing the mission of the company, goals for all aspects of the way the company wants to do business, expectations for the content and quality of the products, and the type of work environment to be supplied for all employees. Mandé organized a series of meetings first with the executives and then with the entire staff to illustrate and discuss the content of this document.

Mandé worked with all executives to build their awareness of the possibilities for growth and profitability in MovieViews, and help them to see their part in the future of the company. She knew a software process improvement programme was instrumental in improving the quality of their products: it was not enough to concentrate solely on finding defective products before the customer saw them. Working with groups of employees to understand the problems, a simple qualitative software process goal was defined:

> *"Improve our ways of developing and upgrading software so that we can produce software products with fewer defects at no higher cost or length of time to market."*

B.2.2 Step 2: Initiate process improvement (see 5.2)

Mandé immediately understood the need for precise data to get a reliable picture of the current status. First she asked the executives to prepare a report containing information on the number and types of error for each family of product, and the phases in which the errors were detected.

On completion of the report, it was immediately clear that the data were not comparable and that there were significant differences between different families of products. As a result Mandé realized that the company did not have a common way of producing software.

She immediately established a team to co-ordinate an improvement programme to address the problem. The team included herself, the Director of Quality Assurance, Doq, the Engineering Director, Ed, the Software Application Manager, Sam, and the Marketing Director, Mark.

Each team member was charged with reporting back to his or her department on the status of the Process Improvement Programme and to make its success a personal issue. This sense of commitment was conveyed to the other employees. Mandé made *"Status of the process improvement programme"* the first item on the agenda of each of her staff meetings. She discussed it at board meetings and made sure that the subject was discussed at the local user group meetings.

Mandé had been recruited from a company with a successful process improvement record. Consequently, she knew that it was essential to reserve resources for process improvement. She decided to include defect reduction as a major goal in the company's annual business plan, and to set aside a budget for improvement and staff training.

At this point, Mandé had two major problems:

(i) she did not know where the defects were actually generated and consequently which parts of the software process to improve;

(ii) she did not know how much the company could improve in the near future (one year), without compromising the production deadlines for new products or reducing the support levels for the existing ones.

Knowing that a plan was needed to make the improvement effort cost-effective, Mandé decided to chair the improvement team herself and make the preparation of an overall process improvement programme plan the first priority for the team.

A preliminary version of the process improvement programme plan addressed the following:

- How does the company take the goals defined in the business plan and map them to the goal of producing a quality software?
- Where should improvement efforts be directed?
- What are the constraints on software process improvements (resource availability to implement improvements, time scales for improvements, etc.)?
- What are the specific implementable, measurable goals that can be derived from the broader software process goal?
- What historical information exists that might help in understanding the current problems, and where improvements are most needed?
- What methods of communication and review will enable every employee to understand the progress in improvements, and the process improvement team to understand what actions need to take place?
- What are the characteristics of the company culture that support change?
- What are the characteristics that may need to change for improvements to happen and persist over time?

As a starting point for the improvement programme it was decided to undertake a process assessment to get a better picture of the current situation.

B.2.3 Step 3: Prepare for and conduct a process assessment (see 5.3)

Before the assessment actually took place, all the technical organizations worked to collect measurements of the software processes as they were currently being performed. Since product quality had been identified as a serious problem, each software engineer was encouraged to keep track of time spent:

- on correction of problems identified by customers;
- for requirements analysis, design, and documentation before starting to code;
- on correction of problems identified by the developer.

These measurements were used to raise the awareness of each software engineer of how much time was being spent correcting problems and how much of this time might be better spent in preliminary activities to prevent those problems from happening.

After the company executives had approved the preliminary version of the process improvement programme plan, they agreed that the Engineering Director, Ed, should sponsor the assessment. They also agreed to assign ownership of the assessment and its results to the Application Manager, Sam.

Ed and his team decided that, although training was taking place on many aspects of software processes and their assessment, no employee was qualified to act as an assessor. Therefore a contract was established with an external consultant, Cas, a qualified assessor. Cas worked with the process improvement team to develop an assessment purpose which read, in part:

> *"The purpose of this assessment is to identify those activities in the software processes which contribute significantly to the inclusion of defects in our software products. This is part of a continuous process improvement programme to reduce the number of defects through the enhancement of worthwhile but inadequate processes, the replacement of inefficient processes and the addition of new processes ".*

The three projects that develop the most important product families (Blueball, Purplepillow, and, Pinksquare) were selected for assessment. Amongst these, Pinksquare had shown the best results and Blueball the worse. It was decided to assess all the processes involved in software production in these units, including the entire life cycle and the supporting activities.

Two major constraints were placed on the assessment. Firstly, project Purplepillow had important delivery dates to meet for the next release of the product. The assessment was to create minimal disturbance to Purplepillow's schedule, even at the expense of depth of coverage. Secondly, the survival of Blueball in the marketplace was dependent on re-engineering the product. An in-depth assessment of the Blueball re-engineering project was essential.

An assessment team was appointed, including Cas, a representative of the Quality Assurance Unit, and an expert from each project in order to provide a good understanding of the needs and characteristics of each project and product family. It was decided to interview some of the marketing staff since customer support was performed within the Marketing Unit, and these staff had the most direct contacts with the customers.

Based on the assessment scope defined by the improvement team, the assessment team decided to assess the following processes:

Customer-Supplier Process Category (CUS)

CUS.3 Identify customer needs

CUS.6 Support operation of software

CUS.8 Assess customer satisfaction

Engineering Process Category (ENG)

all processes

Project Process Category (PRO)

all processes

Support Process Category (SUP)

all processes

Even though the executives agreed that assessment was a worthwhile activity, they were concerned about the time that would be taken for performing it. Process improvement team members planned to devote about 10% of their time to improvement activities. The assessment team members expected to work almost full time for a month to complete the assessment.

During the assessment, records were kept of comments made by interviewees, procedures described by the software engineers, questionable practices, and the opinions of interviewers and interviewees.

B.2.4 Step 4: Analyse assessment results and derive action plan (see 5.4)

The process improvement team analysed the results of the assessment, primarily looking at findings which could be related to product quality. Feedback from the assessment indicated that software was being developed without a clear understanding of requirements, so that the software engineers spent a lot of time going back and making corrections once they did understand the requirements. One conclusion was that last minute changes had the potential to introduce defects into the software. In addition, system testing activities were not performed in a systematic way.

Detailed analysis showed that there was a satisfactory situation in the area of project management: most of the processes were at capability level two and some even at capability level 3 with no significant differences between the projects. Design activities seemed to be satisfactory: ENG.3, ENG.4 and ENG.5 were all at level 2.

The problem areas appeared to be related to requirement definition and system testing. The derived process capability level ratings for these processes were:

Identify customer needs

CUS.3[P1, P2, P3];1=[0, 25,25,50]
CUS.3[P1, P2, P3];2=[0,0,40,60]
CUS.3[P1, P2, P3];3=[0,0,0,100]
CUS.3[P1, P2, P3];4=[0,0,0,100]
CUS.3[P1, P2, P3];5=[0,0,0,100]

Develop software requirements

ENG.2[P1, P2, P3];1=[0,80,20,0]
ENG.2[P1, P2, P3];2=[0,60,40,0]
ENG.2[P1, P2, P3];3=[0,0,0,100]
ENG.2[P1, P2, P3];4=[0,0,0,100]
ENG.2[P1, P2, P3];5=[0,0,0,100]

Develop system requirements and design

ENG.1[P1, P2, P3];1=[0,50,25,25]
ENG.1[P1, P2, P3];2=[0,0,60,40]
ENG.1[P1, P2, P3];3=[0,0,0,100]
ENG.1[P1, P2, P3];4=[0,0,0,100]
ENG.1[P1, P2, P3];5=[0,0,0,100]

Integrate and test system

ENG.6[P1, P2, P3];1=[100,0,0,0]
ENG.6[P1, P2, P3];2=[10,25,50,15]
ENG.6[P1, P2, P3];3=[0,0,20,80]
ENG.6[P1, P2, P3];4=[0,0,0,100]
ENG.6[P1, P2, P3];5=[0,0,0,100]

The improvement team decided to address these processes as the main priorities for the improvement programme.

The assessment team recommended that MovieViews should plan a multi-step increase in capability levels for the selected processes throughout the company: first ensure that the base practices properly implemented for the selected processes, and then concentrate on raising each process to capability level 2.

A major goal of the improvement team was to evaluate how process improvement might affect product quality. A set of metrics was established to achieve this goal. As the level of defects was the main issue, a program of measurement of defects was put in place to provide visibility of improvement. The key measures were to be:

- the number of open problem reports from customers during each reporting period of two weeks [what is the magnitude of the problem?];
- the number of problem reports from customers per product [where are the worst problems?];
- the distribution of problem reports within product by type and severity [do we have many small problems (probably systemic process problems), a few major problems (might be isolated process implementation problems), or some combination of both?].

A target was established to show all employees the goals for these metrics:

- percent of time spent on rework <5%;
- percent of time spent on new projects >50%;
- percent of time spent in reviews >5%;
- percent of time spent in testing, or quality assurance activities <10%;
- number of problem reports from a customer open per reporting period decreased to no more than one;
- no single product with more than one problem report from a customer/100,000 lines of code opened against it;
- no Class 1 (the most severe) problem reports from a customer on any product.

In addition it was decided to use the time cards of personnel in the Software Applications Group and Engineering Department to track costs related to non quality. The time cards were used to measure:

- any scheduled tasks;
- each phase of development;
- rework;
- new projects;
- education and training;
- reviews and other quality management tasks.

This data could be used to derive useful information, including information about quality and cost.

Information About Quality

- the percent of time on rework reflected the quality of work (especially rework for which MovieViews was not getting paid or rework created just because someone was careless);
- the percent of time on new projects indicated how far ahead they were looking towards new business;

Information About Cost

- time spent in reviews, testing or other quality assurance activities - the cost of quality.

Measurements were taken to show the current status. The figures were:

- 35% of time spent on rework;
- 10% of time spent on new projects;
- <1% of time spent on reviews;
- 10% of time spent on testing and other quality assurance activities;
- 40 customer-generated problem reports were open and distributed as follows:

	Problem classification		
PRODUCT	**Class 1**	**Class 2**	**Class 3**
Pinksquare (est. 400.000 lines of code)	2	0	1
Purplepillow (est. 150,000 lines of code)	0	18	4
Blueball (est. 200,000 lines of code)	6	5	4

Process improvement actions were identified and documented in an action plan. These included:

- the introduction of a systematic approach to the collection and analysis of customer needs;
- the definition of a clearer interface between staff supporting the customer and staff designing the products;
- the introduction of a systematic approach to system and acceptance testing;
- the introduction of formal reviews to be held on completion of each of these processes;
- a mechanism to monitor the implementation of the improvements.

The team made an interim presentation to senior management which was well received. Management approved the updated process improvement programme plan, including the action plan and the revised budget and schedules, and authorized the start of the process improvement projects.

B.2.5 Step 5: Implement improvements (see 5.5)

This was the first action plan to be implemented. Others would be implemented on a regular schedule as defined in the process improvement programme plan.

Three improvement project were created: one to carry out improvements on requirements definition; a second on system and acceptance testing; and a third on reviews. Each project established its own project plan describing the strategy to be adopted to implement the improvement.

The first project looked initially for possible changes in the organization, and then to the establishment of a clear procedure to communicate user expectations to the development group.

The second project set out to collect the good practices from the Pinksquare project and to generalize them for transfer to other projects.

The third project hired an experienced consultant to brief them about the practical experiences in putting reviews in place within industrial organizations and to help them to identify the best way of introducing reviews within MovieViews.

Each of the improvement projects was piloted in one of the three product development projects and measurements were taken to get preliminary confirmation of the benefits derived from the selected improvement actions.

B.2.6 Step 6: Confirm improvement (see 5.6)

Even though the targets were not quite achieved, the improvement team was encouraged by the results of the pilot projects, and decided it was worth proceeding. At least the team was confident about the reliability of the data collection, and project staff could see the benefits. The team presented the experiences of the pilot projects in seminars to the entire staff.

Tracking already showed a decrease in product defects as a result of increasing reviews and awareness of the need for the reviews. Some further reorganization was needed, however, to move from reliance on product checking to process reviews. A cost-benefit analysis showed that the cost of process improvement was less than the gain from fewer customer complaints and more products sold to the distributors.

The changes introduced in the pilot projects were carefully reviewed and documented in procedures that were distributed to all units to be used as a reference to plan their own projects.

B.2.7 Step 7: Sustain improvement gains (see 5.7)

Once reviews, including formal preparation and documentation of the results, were being routinely scheduled as part of the software process, the simple metric of whether or not reviews were held was expanded to track data such as the number of changes required as a result of the review; and the hours required by the software staff to make these changes. This, combined with the product data measurements, enabled MovieViews to continue to refine and improve the software process as part of a continuous improvement program.

The review goals were enhanced to include the evaluation of how the newly established processes were being applied. Data collected within reviews started to be used to monitor these processes within the entire organization.

B.2.8 Step 8: Monitor performance (see 5.8)

Senior management continued to be involved with the process improvement programme. Further assessments were planned to confirm the improvements and to identify new ones.

Annex C (informative)

References

This annex describes key sources of material used in creating the guidance in this part of the International Standard.

1. ISO 9004-4 : 1993, Quality management and quality system elements - Part 4: Guidelines for quality improvement

2. IDEAL : An Integrated Approach to Software Process Improvement (SPI), Bill Peterson and Ron Radice, SEI Symposium, Pittsburgh, August 1994.

3. Measurement Definitions for DoD Systems : recommendations for an Initial Core Set, SEI-93-TechReview-2.

4. Concepts on Measuring the benefits of Software Process Improvements, J. A. Rozum, SEI-93-TR-009.

5. Software process Assessment and Improvement : The BOOTSTRAP Approach, P. Kuvaja, J. Simila, L. Krzanik, A. Bicego, G. Koch, S. Saukkonen, Blackwell Business, 1994.

6. ami, a quantitative approach to software management, The ami consortium, South Bank University, London, 1992.

Annex D (informative)

Mapping to the process ORG.3 and the PDCA cycle

The process model in part 2 of this International Standard includes the process *Improve the Process* (ORG.3) whose purpose is "to gain competitive advantage by continually improving the effectiveness and efficiency of the software processes used by the organization". The improvement cycle described in this part of the International Standard is an implementation of that process. This annex provides the mapping between the steps of the improvement cycle and the process ORG.3.

The PDCA (Plan-Do-Check-Act) cycle is widely used by organizations involved in Total Quality Management (TQM) and continuous process improvement for quality. This annex, therefore, also maps the steps of the improvement cycle described in this part of the International Standard to the PDCA cycle. Two levels of mapping have been identified; the upper level corresponds to cross-functional responsibilities and the process improvement programme; the bottom level corresponds to the implementation of improvement actions through improvement projects.

Improvement steps	**ORG.3 Practices**		**PDCA – process improvement programme**	**PDCA – improvement projects**
Examine Organization' s needs	ORG.3.1	Identify improvement activities		
Initiate process improvement	ORG.3.2	Define scope of improvement activities	Plan	
Perform process assessment	ORG.3.3	Understand the process	Do	
Analyse assessment results & derive action plan	ORG.3.4 ORG.3.5	Identify improvement Prioritize improvements		Plan
Implement improvements	ORG.3.7	Change the process		Do
Confirm the improvement	ORG.3.8	Confirm the improvements		Check
Sustain improvement gains	ORG.3.9	Deploy improvements	Check	Act
Monitor performance	ORG.3.6	Define measures of impact	Act	

Annex E (informative)

Mapping to ISO 9004-4

The guidance on software process improvement described in this part of the International Standard builds directly on the guidance of ISO 9004-4. This annex provides the mapping between the clauses of this document and ISO 9004-4.

This document	**ISO 9004-4**
5. Methodology for software process improvement	6 Methodology for quality improvement
5.1 Examine the organization's needs and business goals	6.1 Involving the whole organization
5.2 Initiate process improvement	6.2 Initiating quality improvement projects or activities
5.3 Prepare for and conduct a process assessment	6.3 Investigating possible causes
5.4 Analyse assessment results and derive action plan	6.4 Establishing cause and effect relationships
5.5 Implement improvements	6.5 Taking preventive or corrective actions
5.6 Confirm improvements	6.6 Confirming the improvement
5.7 Sustain improvement gains	6.7 Sustaining the gains
5.8 Monitor performance	6.8 Continuing the improvement
6. Cultural issues	4.2 Environment for quality improvement
6.1 Management responsibility and leadership	4.2.1 Management responsibility and leadership
6.2 Values, attitudes and behaviour	4.2.2 Values, attitudes and behaviours
6.3 Process improvement goals and motivation	4.2.3 Quality improvement goals
6.4 Communication and teamwork	4.2.4 Communications and teamwork
6.5 Recognition	4.2.5 Recognition
6.6 Education and training	4.2.6 Education and training
7. Management	5. Managing for quality improvement
7.1 Organizing for process improvement	5.1 Organizing for quality improvement
7.2 Planning for process improvement	5.2 Planning for quality improvement
7.3 Measuring process improvement	5.3 Measuring quality improvement

7.4 Reviewing process improvement activities

5.4 Reviewing quality improvement activities

Consolidated product

Software Process Assessment –
Part 8 : Guide for use in determining supplier process capability

Version 0.03

(Formerly PCDG 1.00)

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

These terms and conditions apply to the set of documents developed by the SPICE Project, and published within the Project as Version 1.0, with the following titles:

– Part 1 : *Concepts and introductory guide*
– Part 2 : *A model for process management*
– Part 3 : *Rating processes*
– Part 4 : *Guide to conducting assessment*
– Part 5 : *Construction, selection and use of assessment instruments and tools*
– Part 6 : *Qualification and training of assessors*
– Part 7 : *Guide for use in process improvement*
– Part 8 : *Guide for use in determining supplier process capability*
– Part 9 : *Vocabulary*

1. You may copy and distribute verbatim copies of any or all of the Documents as you receive them, in any medium, provided that you conspicuously and appropriately publish with each copy a copy of these Terms and Conditions. You may charge a fee for the physical act of transferring a copy.
2. You may copy extracts from these documents in materials for internal or public use, providing you provide clear acknowledgment of the source of the material, by citation or other appropriate means.
3. You may not copy, modify, sub-license, or distribute the Documents except as expressly provided under these Terms and Conditions.

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout
Members:	Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : ***Concepts and introductory guide***

Product Manager: Terry Rout

– Part 2 : ***A model for process management***

Product Managers: Al Graydon, Mark Paulk

– Part 3 : ***Rating processes***

Product Manager: Harry Barker

– Part 4 : ***Guide to conducting assessment***

Product Manager: Harry Barker

– Part 5 : ***Construction, selection and use of assessment instruments and tools***

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : ***Qualification and training of assessors***

Product Manager: Ron Meegoda

– Part 7 : ***Guide for use in process improvement***

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : ***Guide for use in determining supplier process capability***

Product Manager: John Hamilton

– Part 9 : ***Vocabulary***

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards)

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 8

Guide for use in determining supplier process capability

Contents

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialing to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being co-ordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

- Part 1 : *Concepts and introductory guide*
- Part 2 : *A model for process management*
- Part 3 : *Rating processes*
- Part 4 : *Guide to conducting assessment*
- Part 5 : *Construction, selection and use of assessment instruments and tools*
- Part 6 : *Qualification and training of assessors*
- Part 7 : *Guide for use in process improvement*
- Part 8 : *Guide for use in determining supplier process capability*
- Part 9 : *Vocabulary*

This part of the standard (Part 8) is for guidance only.

1. Scope

This part of the International Standard provides guidance on how to utilize process assessment for the purposes of process capability determination

A process capability determination (PCD) is a systematic assessment and analysis of selected software processes within an organization, carried out with the aim of identifying the strengths, weaknesses and risks associated with deploying the processes to meet a particular specified requirement.

Process capability determination is applicable in a variety of situations; the specified requirement may involve a new or an existing task, a contract or an internal undertaking, a product or a service, or any other requirement which is to be met by deploying an organization's software processes.

This guidance is intended to be applicable across all software application domains, over all software organizational structures, within any software customer-supplier relationship, and to any organization wishing to determine the process capability of its own software processes.

This guide is primarily aimed at:

- the sponsor who initiates the process capability determination;
- the organization whose process capability is to be determined;
- the assessment team;
- tool and method developers.

2. Normative references

There are no normative references in this part of the International Standard.

3. Definitions

For the purposes of this part of this International Standard, the definitions in *Software Process Assessment - Part 9 : Vocabulary* apply.

4. Introduction to process capability determination

4.1. Overview

4.1.1. Purpose

A process capability determination is a systematic assessment and analysis of selected software processes within an organization, carried out with the aim of identifying the strengths, weaknesses and risks associated with deploying the processes to meet a particular specified requirement.

One of the main reasons for carrying out a process capability determination is to obtain information upon which to base a procurement-related decision. A procurer may initiate a process capability determination to assess the risk of entering into a contract with a particular supplier. The procurer may carry out process capability determinations on a number of competing suppliers as one element of a pre-contract selection activity. Conversely, suppliers may wish to carry out a process capability determination on their own processes before deciding whether to bid for a contract, as part of their own assessment of the business risks involved. A process capability determination may also be initiated for a number of other reasons; for example by a supplier during the course of a project to establish what the risks are to completing the work.

Process capability determination may be applied to a variety of situations: the specified requirement may involve a new or an existing task, a contract or an internal undertaking, a product or a service, or any other requirement which is to be met by deploying an organization's software processes.

4.1.2. Core and extended process capability determination

This guide presents two alternative approaches to process capability determination described below.

Core process capability determination is a minimum, streamlined set of activities applicable whenever a single organization proposes to meet a specified requirement by deploying its current process capability, without any partners or sub-contractors being involved.

Extended process capability determination is applicable when an enhanced capability is proposed, or when consortia or sub-contractors are involved.

In either case the conduct of process capability determination is described in three separate stages, as set out in clause 5 of this guide.

4.1.3. Basis of process capability determination

The output of a process assessment conducted according to the provisions of this International Standard is a process profile. This profile represents an organization's process capability in a particular assessment context and is reusable for both process capability determination and process improvement in that particular context or a similar context.

4.1.4. Assessment approaches

Part 4 of this International Standard describes two main approaches to process assessment: *self-assessment* and *independent assessment*. Either or both may be used during a process capability determination. In a two-party contractual situation, a procurer may wish to invite potential suppliers to provide a self-assessment profile when submitting a proposal for a contract. Such an approach offers the benefit of sharing both the cost and the benefit of the process assessment, since suppliers may also use the assessment results within their own process improvement programmes.

The procurer may choose to accept a self-assessment at face value, or alternately may reserve the right to initiate an independent assessment to verify that the self-assessment is a true representation of the supplier’s process capability. Alternatively, the procurer may decide to rely entirely upon an independent assessment and make this a condition of contract award.

This International Standard thus offers the benefit of reducing disruption to suppliers’ business activities caused by multiple process assessments, since the same assessment results may be offered to many procurers. It also provides procurers with a rigorous and defensible approach to supplier capability determination, and promises to reduce assessment costs through the reuse of results and the utilization of self-assessments.

4.1.5. Process-oriented risk

During a process capability determination, a selection of an organization's software processes are assessed, and the results analysed to identify strengths, weaknesses and risks. Process capability determination does not address all aspects of risk, which may include strategic, organizational, financial, personnel and many other factors. The output from a process capability determination feeds into this wider risk analysis, but confines itself to *process-oriented risk*.

The process architecture of this International Standard rests on a reference model contained within Part 2. This model sets out 35 processes defining the software engineering or management base practices of each, as well as a set of generic practices which apply to all processes. The generic practices are concerned with process management and are grouped into ordered capability levels, which progressively describe major enhancements to process capability. The single generic practice in the *Performed Informally* capability level summarizes the overall adequacy of the base practices of a process. Additional, user-defined processes can also be added if required.

During a process assessment, individual practices are rated by qualified and trained assessors against a four-point adequacy scale using an appropriate assessment instrument. Practices may be rated *fully, largely, partially* or *not* adequate with respect to the process purpose statement set out in the process model in part 2 of this International Standard. These ratings are then aggregated into a process capability profile that indicates, for each process assessed, and for each capability level for each process, how well the generic practices are achieving their intended purpose. Rating and the aggregation of ratings is described in parts 3 and 4 of this International Standard.

The key to process-oriented risk lies in the process model, the good process management practices it reflects through the generic practices, and the benefits that arise from deploying them. Process-oriented risk arises from inappropriate process management - i.e. not deploying appropriate generic practices, or from deploying them in a way which is assessed in the particular context as less adequate than required.

4.1.6. Target capability

Within this guide, the capability of a process is expressed in terms of the adequacy of its generic practices.

The process capability determination sponsor[1], who initiates the process capability determination, produces a target capability statement that defines which of the 35 processes in the process model are key to the specified requirement, which generic practices should be applied to each of the key processes, and what degrees of adequacy are required.

For example, the target capability statement could specify six or seven key processes, and indicate that all of the generic practices up to and including the *Well-Defined* capability level should be fully adequate for three of them. It could also indicate that for the remaining processes, all of the generic practices up to and including the *Planned And Tracked* level should be fully adequate.

The target capability is chosen to be that capability which the process capability determination sponsor judges will represent a minimal process risk to the successful implementation of the specified requirement.

4.1.7. Process-oriented risk analysis

Within this guide, process-oriented risk is assessed firstly from the *probability* of a particular problem occurring, and secondly from its potential *impact*, should it occur.

If the process capability determination sponsor's target capability statement indicates that a particular generic practice should be fully adequate for a particular process, while the assessed adequacy of the generic practice is less than fully adequate, then there is a gap between target and assessed capability which increases the probability that the process will not contribute satisfactorily towards meeting the specified requirement. For example, if the process capability determination sponsor believes that for a particular process, all of the generic practices up to and including the *Planned And Tracked* capability level should be fully adequate, and if the assessed process profile shows that capability level 1 is not fully adequate, then a major gap exists and there is a high probability of a problem occurring.

The potential impact of the problem depends upon the capability level within which it occurs. For example, if the base practices of a key process are assessed less than fully adequate, as reflected by the rating for the single generic practice in the *Performed Informally* capability level, then the process is incomplete and this may lead to missing work products, or unacceptable product quality, or both.

4.1.8. Output

The output of a process capability determination is the process capability report, which summarizes the strengths and weaknesses, expressed in terms of capability level gaps, and the risks associated with each key process included within the target capability statement.

[1] The process capability determination sponsor may be a procurer initiating a process capability determination to determine whether a potential supplier's processes are suitable for a particular requirement, or an organisation initiating a process capability determination to determine whether its own processes are suitable.

4.2. Target capability

Process capability determination sponsors may wish to develop or purchase an appropriate method or tool for defining target capability. A number of approaches are possible, but most will be based on the following principles.

The target capability is chosen to be that capability which the process capability determination sponsor judges will represent a minimal process risk to the successful implementation of the specified requirement.

Target capability is expressed within a target capability statement, which lists processes key to meeting the specified requirements and states, for each such key process, the required adequacy of each generic practice.

Only generic practice adequacy targets of *fully*, or *largely*, or *not required* should be set.

For each key process, process capability determination sponsors should identify which generic practices are required, and set the degree of adequacy for each. Generic practice adequacy may be set in several ways; for example the same degree of adequacy may be allocated to:

- all of the generic practices within a capability level;
- all of the generic practices within a common feature.

4.2.1. Setting target capability

A number of approaches to setting target capability are possible. One approach is to:

- identify an initial set of key processes;
- set default generic practice adequacy targets for the initial set of key processes;
- review and adjust the default generic practice adequacy targets;
- add further processes, and set adequacy targets for the further processes.

These steps are described in the following paragraphs.

4.2.2. Initial key processes

The processes in the process model which contribute most directly to the delivery of products and services are those within the *Customer-Supplier* and *Engineering* process categories. Processes from the *Project, Support* and *Organization* process categories provide a more indirect contribution.

Key processes are identified starting with the processes in the *Customer-Supplier* and *Engineering* process areas. Any processes in these categories which are not relevant to the specified requirement should be eliminated, and the remainder designated as the initial set of key processes.

4.2.3. Default generic practice adequacy targets

A good starting position is to state, for each key process, that all of the generic practices in the first three capability levels - *Performed Informally, Planned and Tracked,* and *Well Defined* - should be fully adequate; all of the other generic practices will not be required.

This approach ensures firstly that processes are complete with fully adequate base practices; secondly that generic practices are in place to eliminate unpredictability, missed deadlines, budget overspend and reduced output quality; and thirdly that processes are deployed following organization-wide standard process definitions, thus providing confidence that future performance will be consistent with past accomplishments.

4.2.4. Adjusting generic practice adequacy targets

Requiring that generic practices in the *Quantitatively Controlled* capability level should also be fully or largely adequate for a given process may reduce performance risks. For instance, a particular specified requirement may demand that some processes be controlled quantitatively. Generic practices within the *Continuously Improving* capability level may occasionally also be needed, but for many organizations, this degree of process management may not yet be practical. Alternatively, process capability determination sponsors may feel that for a particular key process, only generic practices within the first two capability levels are appropriate.

4.2.5. Adding further processes

Many generic practices are related to processes within the *Project, Support* and *Organization* process categories.

For example, if the generic practice *2.2.2 '...Do configuration management..'* has been included for a process within the *Engineering* process category, then the *Configuration Management* process within the *Support* process category may also be included as a key process.

The target capability for processes in the *Project, Support* and *Organization* process categories is determined by the extent to which they support generic practices applying to the initial set of key processes. Other processes from the *Project, Support* and *Organization* process categories may also be included in the target capability statement where they are relevant to the specified requirement.

Note that the specified requirement may be for an *organizational capability*, rather than a product or service. The specified requirement may be to establish a strong configuration management process as an end in itself. This class of specified requirement would arise from an organization's business goals and priorities.

4.3. Process-oriented risk analysis

Within this guide, process-oriented risk is inferred from the existence of gaps between target capability and assessed capability. Such gaps are identified at the practice level: if the target capability statement indicates that a particular generic practice should be fully adequate, while the assessed practice adequacy rating is less than fully adequate, then a gap is said to exist.

Process-oriented risk is assessed firstly from the probability of a particular problem occurring, and secondly from the nature of its impact. The probability is derived from the extent of any gaps between an assessed capability profile and a target capability statement. The nature of the impact depends upon the capability level within which the gap occurs.

4.3.1. Assessed capability profile

The assessed capability profile will be in the form of an output from a process assessment conducted according to the provisions of this International Standard. This will show, for each process assessed and for each capability level, the proportions of practices which have been assessed fully, largely, partially or not adequate. Figure 1 illustrates how an example assessed capability might be illustrated.

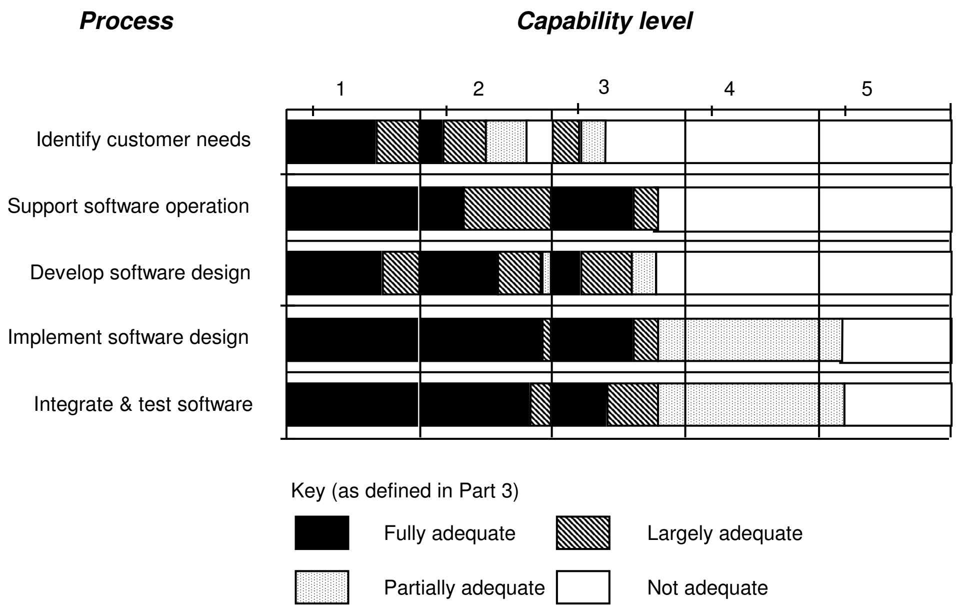


Figure 1 - Assessed capability profile

Generic practice adequacy is defined as a judgement, within the process context, of the extent to which the implemented generic practice satisfies its purpose.

Because practice adequacy is defined in this way, process assessment is highly context-sensitive. For example, an organization developing a large, complex and safety-critical software system would need to deploy a highly refined process in order to be assessed fully adequate at the *Performed Informally* level. In contrast an organization working on straightforward, non-critical applications would need far less sophistication to attain a similar assessment result. Therefore process capability ratings are meaningful only within their stated process context.

4.3.2. Target capability statement

Figure 2 shows one way that a target capability statement might be illustrated, along with the example assessed capability profile from figure 1.

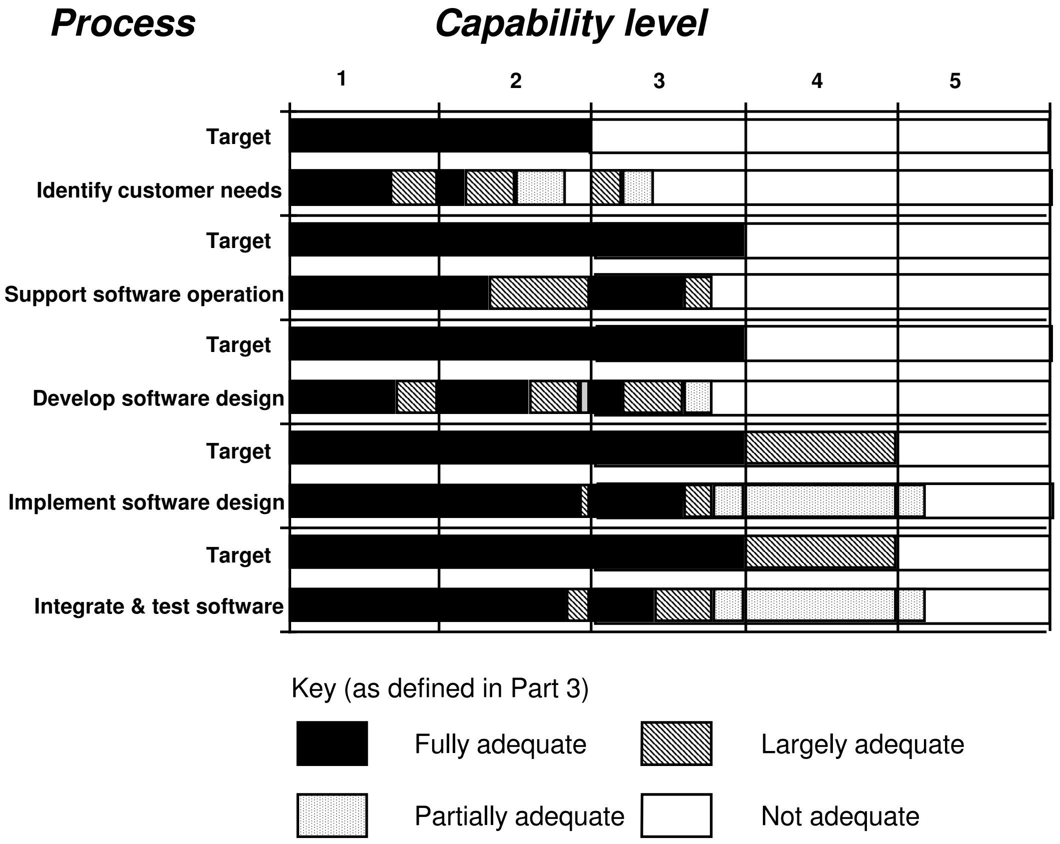


Figure 2 - Target capability with assessed capability

In this example the process capability determination sponsor has deemed that for the first process, *Identify Customer Needs,* all of the generic practices up to and including the *Planned and Tracked* level should be fully adequate. For the next two processes, all generic practices up to an including the *Well-Defined* level should be fully adequate. For the final two processes, not only should the generic practices up to and including the *Well-Defined* level be fully adequate, but in addition those of the *Qualitatively Controlled* level should also be largely adequate.

4.3.3. Probability

The probability of problems occurring is inferred from the extent of gaps between the target capability statement and the assessed capability profile.

For a particular process, individual gaps are identified by comparing individual assessed practice adequacy ratings to the corresponding adequacy targets specified in the target capability statement. Gaps are designated as shown in table 1.

Table 1 - Gaps associated with individual practices

Target	Assessed	Gap
Fully Adequate	Fully Adequate	None
	Largely Adequate	Minor
	Partially Adequate	Major
	Not Adequate	Major
Largely Adequate	Fully Adequate	None
	Largely Adequate	None
	Partially Adequate	Major
	Not Adequate	Major

Gaps within a capability level, and the corresponding probability of a problem occurring, are designated as shown in table 2.

Table 2 - Gaps associated with capability levels

Number of Individual Gaps within Capability Level	Capability Level Gap	Probability of problems occurring
No major or minor gaps	None	Very Low
Minor gaps only	Slight	Low
A single major gap concerning a generic practice	Significant	Medium
A single major gap concerning a base practice or more than one major gap concerning a generic practice	Substantial	High

Note that because of the way *largely* adequate is defined (see part 3), any number of *minor* gaps within a single capability level constitute only a *slight* gap at the capability level.

4.3.4. Impact

The previous section showed how the probability of problems occurring is inferred from the extent of the gap at a capability level.

The nature of the potential *impact* of a particular problem depends only upon the *capability level* within which it occurs, as shown in table 3.

Table 3 - Nature of potential impact of problems at each capability level

Capability Level where problem occurs	**Nature of Impact**				
	Missing work products, unacceptable product quality	Cost or time overruns	Reduced cost effectiveness, reduced spatial and temporal uniformity of performance	Inability to predict performance or timely detect problems	Reduced cost/time optimization-reduced ability to cope with changes in technology.
Continuously Improving	**No identifiable Impact**	**No identifiable Impact**	**Low Impact**	**Medium Impact**	**High Impact**
Quantitatively Controlled	**No identifiable Impact**	**Low Impact**	**Medium Impact**	**High Impact**	**High Impact**
Well-Defined	**Low Impact**	**Medium Impact**	**High Impact**	**High Impact**	**High Impact**
Planned and Tracked	**Medium Impact**	**High Impact**	**High Impact**	**High Impact**	**High Impact**
Performed Informally	**High Impact**	**High Impact**	**High Impact**	**High Impact**	**High Impact**

4.3.5. Overall risk

The overall process-oriented risk associated with a single process may be summarized as shown in table 4.

Table 4 - Overall process-oriented risk

	Extent of Capability Level Gap			
Capability Level	None	Slight	Significant	Substantial
Continuously Improving	**No Identifiable Risk**	**Low Risk**	**Low Risk**	**Low Risk**
Quantitatively Controlled	**No Identifiable Risk**	**Low Risk**	**Low Risk**	**Medium Risk**
Well-Defined	**No Identifiable Risk**	**Low risk**	**Medium risk**	**Medium Risk**
Planned and Tracked	**No Identifiable Risk**	**Medium Risk**	**Medium Risk**	**High Risk**
Performed Informally	**No Identifiable Risk**	**Medium Risk**	**High Risk**	**High Risk**

To use tables 1 to 4, consider each key process in turn, and then, for each process, consider each capability level in turn. Categorize individual gaps at the practice level using table 1, and then determine the capability level gap and probability of a problem occurring using table 2. The potential impact of the problem is then obtained from table 3. For example, a substantial gap within the *Planned and Tracked* level implies a high probability of problems occurring, which, should they occur, will have potentially high impact on product quality and a medium impact on budget and schedule. According to table 4, this then constitutes a *high risk.*

If gaps exist at more than one capability level, then the overall risk is determined from whichever row of table 4 which shows the greater risk.

It is emphasized that table 4 is merely a guide to overall risk; nominal risk levels should always be confirmed by a critical review against experience and reality.

It should be noted that a particular row from table 4 is relevant only if the generic practices of the particular capability level have been included in the target capability statement.

4.4. The process capability report

The process capability report is the final output of process capability determination. It consists of a summary and a detailed report. The summary consists of three parts:

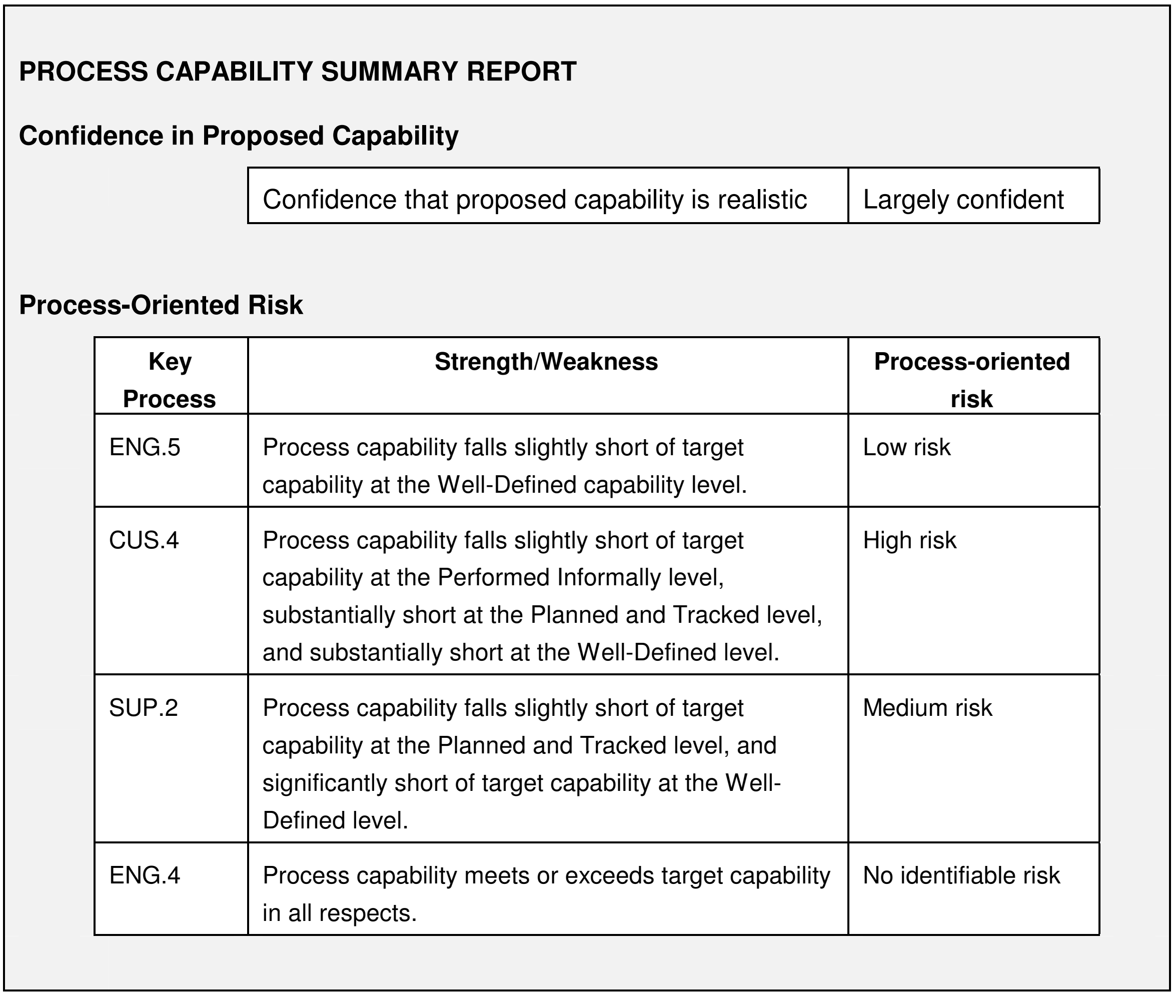
(i) an introduction that describes the context of the process capability determination, who carried it out, and where, when and why it took place;

(ii) a statement of the process capability determination sponsor's confidence that the proposed capability is realistic and likely to be brought to bear in meeting the specified requirement. This confidence may be derived from the results of an independent process assessment, or from some other aspect of the process capability determination sponsor’s relationship with the organization;

(iii) a report, for each key process, of any gap between target capability and proposed capability, and of the process-oriented risk arising from this gap.

Figure 3 illustrates how a summary process capability report might be presented showing the assessed overall risk associated with each process.

The summary report should be supported by a detailed report showing, for each process within the target capability statement, the target and proposed adequacy of every generic practice, listing individual gaps (designated according to table 1) and summarizing capability level gaps (designated according to table 2).



PROCESS CAPABILITY SUMMARY REPORT

Confidence in Proposed Capability

Confidence that proposed capability is realistic	Largely confident

Process-Oriented Risk

Key Process	Strength/Weakness	Process-oriented risk
ENG.5	Process capability falls slightly short of target capability at the Well-Defined capability level.	Low risk
CUS.4	Process capability falls slightly short of target capability at the Performed Informally level, substantially short at the Planned and Tracked level, and substantially short at the Well-Defined level.	High risk
SUP.2	Process capability falls slightly short of target capability at the Planned and Tracked level, and significantly short of target capability at the Well-Defined level.	Medium risk
ENG.4	Process capability meets or exceeds target capability in all respects.	No identifiable risk

Figure 3 - Illustration of process capability summary report

5. Conducting a process capability determination

Process capability determination sponsors may wish to develop or purchase an appropriate method or tool to support the conduct of a process capability determination. A number of approaches are possible, but most will be based on either core or extended process capability determination as explained in the following sections.

5.1. Core process capability determination

Core process capability determination is a minimum, streamlined set of activities applicable whenever a single organization proposes to meet a specified requirement by deploying its current process capability, without any partners or sub-contractors being involved.

Core process capability determination comprises three stages as illustrated in figure 4. The ovals in figure 4 represent activities, the arrows represent information being passed between activities, and the clouds represent comment.

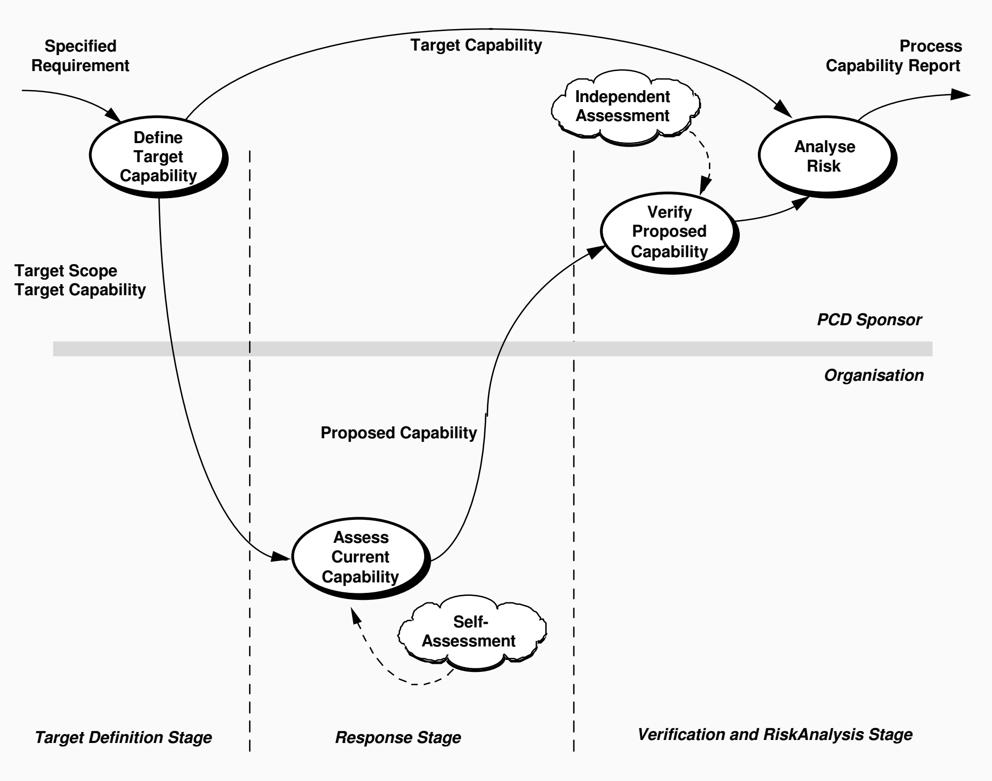


Figure 4 - Core process capability determination

Throughout clause 4 of this guide the term *assessed capability* was used to refer to the output of a process assessment. This clause introduces the term *proposed capability* to represent that process capability which the organization proposes to bring to bear in meeting the specified requirement. For core process capability determination, the proposed capability is the organization’s current assessed capability, represented as the output of a recent, relevant process assessment conducted according to the provisions of this International Standard.

5.1.1. The target definition stage

The process capability determination sponsor is responsible for the target definition stage. The process capability determination is carried out with respect to a specified requirement, which may be expressed in a high-level or detailed form, and may involve a new or existing task, a contract or class of contracts, an internal undertaking, a product or a service, or any other requirement which is to be met by the organization's proposed processes.

During the target definition stage, the process capability determination sponsor:

- plans and initiates the process capability determination;
- develops the target capability statement;
- defines the target scope – i.e. the process assessment context implied by the specified requirement. This may include the minimum number of separate process instances which should be included to represent overall organizational capability. It may also include any extended processes, as described in part 2 of this International Standard, which the process capability determination sponsor wishes to include;
- passes the target scope and, optionally, the target capability statement to potential suppliers.

When initiating the process capability determination, sponsors may wish to request supporting details of current similar projects undertaken by the organization.

Process capability determination sponsors may choose to disclose the target capability statement to potential suppliers, or not as they see fit.

5.1.2. The response stage

During the response stage, the organization assesses its current capability with respect to the target scope. The proposed capability profile is aggregated from assessments of a number of current or recent projects, as described in parts 3 and 4 of this International Standard. This capability profile:

- should be based on a number of process assessments, conducted according to the provisions of this International Standard;
- should correspond to the target scope;
- should be a true representation of the organization's current process capability;
- should be owned by the organization;
- will most likely have been the product of self-assessment, but could also have been produced by a previous independent assessment.

A key feature of this International Standard is that process assessment results are re-usable. Many organizations will have a repository of process assessment outputs generated as part of a process improvement programme. If a number of suitable process assessments are available, then the organization may use the outputs as the basis of the proposed capability. If not, then the organization carries out a self-assessment in accordance with parts 3 and 4 of this International Standard.

5.1.3. The verification and risk analysis stage

5.1.3.1. Verification

The process capability determination sponsor reviews the proposed capability to establish how much credibility it merits, and decides what further action is needed to establish confidence in it. This will typically involve:

- checking that the assessed capability is the result of an assessment conducted according to the provisions of this International Standard;
- checking that the context of the proposed capability matches the target scope;
- carrying out an independent assessment of one or more processes.

A process capability determination sponsor may accept the proposed capability or may wish to initiate an appropriate degree of independent assessment, bearing in mind the nature, cost and importance of the specified requirement. This independent assessment may involve, for example, a sample of key processes, or a comprehensive independent assessment of all key processes specified in the target capability statement. Having carried out the independent assessment, the process capability determination sponsor will be able to compare this independent output with the organization's proposed capability and record the level of confidence in the organization's proposed capability in the terms shown in table 5.

Table 5 - Terminology for expressing confidence in proposed capability

Correspondence of independent assessment to proposed capability	Degree of confidence
The process capability determination sponsor has no reason to doubt the proposed capability, or the results of an independent assessment correspond to the proposed capability.	Fully confident
The results of an independent assessment have fallen slightly short of the organization's proposed capability	Largely confident
The results of an independent assessment have fallen significantly short of the organization's proposed capability	Partially confident
The results of an independent assessment have fallen substantially short of the organization's proposed capability	Not confident

The terms *slightly, significantly* and *substantially* are used here as defined in table 2.

If a process capability determination sponsor is carrying out process capability determinations on a number of competing suppliers, then to ensure consistency, the same independent assessment team should be used to verify each supplier's proposed capability.

Following appropriate verification, the proposed capability becomes an input to risk analysis.

5.1.3.2. Risk analysis

Risk analysis is carried out by the process capability determination sponsor as described in clause 4 of this guide:

For each key process within the target capability statement, the following steps are followed:

- examine the practice adequacy rating for each generic practice within the target capability statement, and designate any individual gaps according to table 1.
- consider each capability level in turn and designate any capability level gaps according to table 2.
- identify the risk corresponding to each capability level gap by referring to table 4.
- record this risk in the process capability report.

5.2. Extended process capability determination

This section provides outline guidance on the additional activities covered within extended process capability determination.

Extended process capability determination is applicable whenever:

- the proposed capability is greater than currently assessed capability; or
- the proposed capability involves a constructed capability (as explained below) with partners or sub-contractors.

Extended process capability determination comprises three stages as illustrated in figure 6. The target definition stage is the same for both core and extended process capability determination. Hence, the following descriptions relate to the *Response* stage and the *Verification And Risk Analysis* stage only.

5.2.1. The response stage

5.2.1.1. Proposing an enhanced capability

The organization's assessed capability may meet or exceed the target capability, but if not, the organization may wish to develop a proposed capability which lies somewhere between the assessed capability and the target capability as illustrated in figure 5.

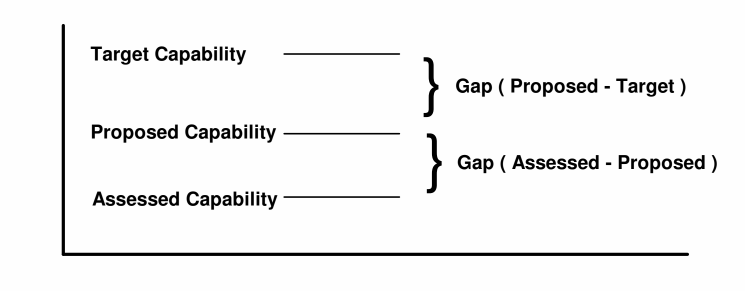


Figure 5 - Target, proposed and assessed capability

Since the specified requirement relates to work to be undertaken in the future, the organization may wish to propose an enhanced capability, justified by a currently assessed capability and a process improvement plan. The process improvement plan may in turn be supported by a process improvement track record if the organization already has a process improvement programme in place.

If the organization's proposed capability falls short of the target capability the organization may wish to submit a shortfall plan, addressing each area where process capability falls short of the target capability, setting out the organization's assessment of the shortfall, and proposing measures to mitigate it.

The proposed capability may be derived by examining the gap between the current and target capabilities and interacting with a process improvement process. The process improvement process will balance previously planned improvements with those necessary to close the gap between the current and target capabilities, but may be constrained by available resources.

The process improvement process may return a process improvement plan, setting out details of what has to be done and what resources are required (see part 7 of this International Standard). If they exist, any process improvement records which add credibility to the plan may also be included, showing what has been achieved in the past.

The organization may therefore wish to pass to the process capability determination sponsor a proposed capability, justified by:

- an assessed capability;
- a process improvement plan;
- a process improvement track record;
- a capability shortfall plan.

This additional information is illustrated in figure 6.

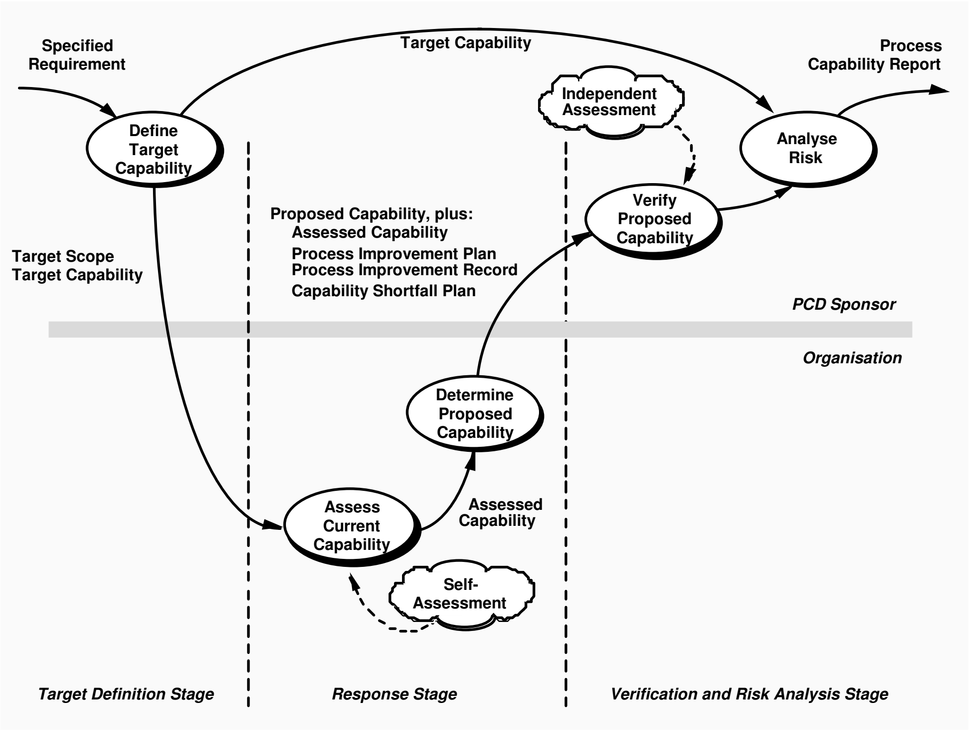


Figure 6 - Extended process capability determination

5.2.1.2. Proposing a constructed capability

The process capability determination will be carried out with respect to a specified requirement which will be worked on in the future. Although the process capability determination will be firmly based on one or more current or recent process assessments, the organization may wish to - or have to - propose a capability which has not yet been constructed. The organization which will undertake the work may not yet exist, and may have to be constructed from existing organizational elements plus sub-contractors, consultants, partners etc. A typical example is illustrated in figure 7.

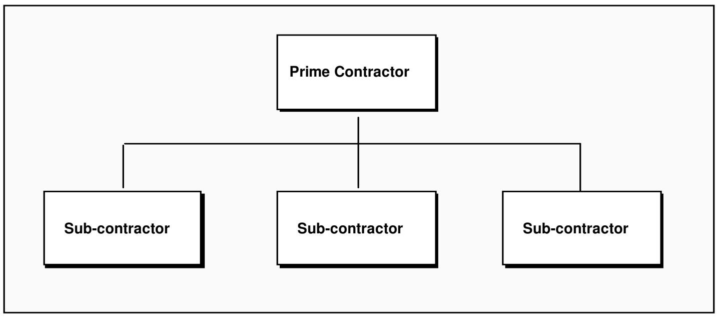


Figure 7 - Constructed capability

There are two different modes that need to be considered when generating a constructed capability from a number of sources.

- **Disjoint mode.** Each key process is deployed uniquely by an individual organization and the constructed capability simply consists of a set of processes that are selected from two or more organizations.
- **Conjoint mode.** A number of organizations deploy the same process or processes in parallel e.g. several organizations developing different sub-systems of an overall requirement.

It is also possible in large or complex contracts to have a mixture of both modes at the same time.

Disjoint mode is used to construct a capability by mixing and matching two or more key processes (processes from the CUS and ENG process categories) to meet (or come as close as is possible/deemed appropriate) to a target capability. Each process is performed uniquely by one organization, and the supporting processes that support it are also provided by that organization.

Under these circumstances, each key process is operating in its own environment, and although for instance, project planning might be performed differently in each organization, it should not affect the ability of each key process to continue to perform to its assessed capability.

It is not possible to construct a capability, however, for the supporting processes. One organization could not provide the 'Plan the Project' process for a number of key processes from different organizations, unless the process was identical in its implementation (not just identical in capability) across those different organizations, which is improbable.

Conjoint mode covers the more complex situation where two or more organizations are deploying the same key process(es) in parallel. It is not valid to average the practice ratings across different organizations. Hence either the worst capability, representing the weakest link in the chain, may be proposed, or, if this is deemed to be inappropriate, then alternatives such as providing all of the ratings or a representation of minimum, maximum and median may be used to provide a more informative representation of capability.

Once again, the inclusion of processes from the Project, Support and Organization process categories, other than those needed to support generic practices applying to individual key processes, is likely to be confusing unless each is clearly identified as to which instance of a key process each is intended to relate to.

Although the constructed capability generated should be representative of the capability of each process in isolation, because two or more organizations are involved, this may lead to unexpected interface issues. Both the organization proposing a constructed capability and the process capability determination sponsor should ensure that suitable mechanisms have been identified to ensure that these issues can be addressed. The more complex the constructed capability and the more disparate the implementation of the processes within the organizations, the more probable that interfacing problems will occur.

5.2.2. The verification and risk analysis stage

Verification within the core process capability determination model is concerned merely with checking that the assessed capability is a true representation of the organization’s processes. Within the extended process capability determination model, extended verification also involves checking:

- the credibility of the process improvement plans upon which the proposed capability is based;
- the integrity of the constructed capability.

SPICE

Consolidated product

Software Process Assessment – Part 9 : Vocabulary

Version 1.00

PREAMBLE

In January 1993 a program of work was approved by ISO/IEC JTC1 for the development of an international standard for software process assessment. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage by developing initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the Technical Report prior to publication as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. The SPICE user trials commenced in January 1995. The working drafts have now been handed over to JTC1/SC7 for the normal process of standards development, commencing in July 1995.

So far as can be determined, intellectual property rights for these documents reside with the individuals and organisations that contributed to their development. In agreeing to take part in the Project, participants agreed to abide by decisions of the Management Board in relation to the conduct of the Project. It is in accordance with this understanding that the Management Board has now agreed to release the baseline set of documents. This introductory statement sets out the terms and conditions under which this release is permitted.

The documents as released are available freely from the SPICE Project File Server, sisyphus.cit.gu.edu.au, by anonymous ftp, or from approved mirrors of the server. A hypertext version of the documents is also available on the World Wide Web at URL http://www-sqi.cit.gu.edu.au/spice/

TERMS AND CONDITIONS

These terms and conditions apply to the set of documents developed by the SPICE Project, and published within the Project as Version 1.0, with the following titles:

– Part 1 : *Concepts and introductory guide*
– Part 2 : *A model for process management*
– Part 3 : *Rating processes*
– Part 4 : *Guide to conducting assessment*
– Part 5 : *Construction, selection and use of assessment instruments and tools*
– Part 6 : *Qualification and training of assessors*
– Part 7 : *Guide for use in process improvement*
– Part 8 : *Guide for use in determining supplier process capability*
– Part 9 : *Vocabulary*

1. You may copy and distribute verbatim copies of any or all of the Documents as you receive them, in any medium, provided that you conspicuously and appropriately publish with each copy a copy of these Terms and Conditions. You may charge a fee for the physical act of transferring a copy.
2. You may copy extracts from these documents in materials for internal or public use, providing you provide clear acknowledgment of the source of the material, by citation or other appropriate means.
3. You may not copy, modify, sub-license, or distribute the Documents except as expressly provided under these Terms and Conditions.

Released on the Authority of the SPICE Management Board:

Project Manager	Alec Dorling
Technical Centre Managers:	
Europe	Harry Barker
Canada, Central and South America	Jean-Normand Drouin
USA	Mark Paulk / Mike Konrad / Dave Kitson
Asia Pacific	Terry Rout
Members:	Catriona Mackie, Bob Smith, Emmanuel Lazinier, Jerome Pesant, Bob Rand, Arnoldo Diaz, Yossi Winograd, Mary Campbell, Carrie Buchman, Ali Azimi, Bruce Hodgen, Katsumi Shintani

Product Managers:

– Part 1 : *Concepts and introductory guide*

Product Manager: Terry Rout

– Part 2 : *A model for process management*

Product Managers: Al Graydon, Mark Paulk

– Part 3 : *Rating processes*

Product Manager: Harry Barker

– Part 4 : *Guide to conducting assessment*

Product Manager: Harry Barker

– Part 5 : *Construction, selection and use of assessment instruments and tools*

Product Managers: Mary Campbell, Peter Hitchcock, Arnoldo Diaz

– Part 6 : *Qualification and training of assessors*

Product Manager: Ron Meegoda

– Part 7 : *Guide for use in process improvement*

Product Managers: Adriana Bicego, Pasi Kuvaja

– Part 8 : *Guide for use in determining supplier process capability*

Product Manager: John Hamilton

– Part 9 : *Vocabulary*

Product Manager: Terry Rout

Acknowledgment:

Acknowledgment is made to all contributors of the SPICE project without whom the project could not have been conceived and carried through successfully.

Note on document formatting

Use the following margins for equivalent printing on A4 or US letter paper (these are NOT the SPICE standards)

Paper size	**A4**	**US letter (imperial)**
Top margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Bottom margin	34.1 mm or 1.34 inches	25.4 mm or 1.0 inches
Left margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches
Right margin	25.4 mm or 1.0 inches	28.4 mm or 1.12 inches

SOFTWARE PROCESS ASSESSMENT Part 9

Vocabulary

Contents

Foreword

In June 1991, the fourth plenary meeting of ISO/IEC JTC1/SC7 approved a study period (resolution 144) to investigate the needs and requirements for a standard for software process assessment.

The results, which are documented in a Study Report (JTC1/SC7 N944R, 11 June 1992), came to the following major conclusions:

- there is international consensus on the needs and requirements for a standard for process assessment;
- there is international consensus on the need for a rapid route to development and trialing to provide usable output in an acceptable timescale and to ensure the standard fully meets the needs of its users;
- there is international commitment to resource the project with an international project team staffed by full time resource, with development being co-ordinated through four technical development centres in Europe, N America (2) and Asia Pacific;
- the standard should initially be published as a Technical Report Type 2 to enable the developing standard to stabilise during the period of the user trials, prior to its issuing as a full International Standard.

The new work item was approved in January 1993 by JTC1. In June 1993 the SPICE Project Organisation was established with a mandate from JTC1/SC7 to:

- assist the standardisation project in its preparatory stage to develop initial working drafts;
- undertake user trials in order to gain early experience data which will form the basis for revision of the published Technical Report prior to review as a full International Standard;
- create market awareness and take-up of the evolving standard.

The SPICE Project Organisation completed its task of producing the set of working drafts in June 1995. These working drafts have formed the basis for this Technical Report Type 2. The period of SPICE user trials commenced in January 1995 and is synchronised in phases to allow feedback to the stages of the technical work.

ISO/IEC Directives state that a Technical Report Type 2 may be used to publish a prospective standard for provisional application so that information and experience of its practical use may be gathered.

This Technical Report Type 2 consists of the following parts, under the general title *Software Process Assessment:*

– Part 1 : *Concepts and introductory guide*

– Part 2 : *A model for process management*

– Part 3 : *Rating processes*

– Part 4 : *Guide to conducting assessment*

– Part 5 : *Construction, selection and use of assessment instruments and tools*

– Part 6 : *Qualification and training of assessors*

– Part 7 : *Guide for use in process improvement*

– Part 8 : *Guide for use in determining supplier process capability*

– Part 9 : *Vocabulary*

1 Scope

This part of the International Standard defines the terms used throughout this International Standard.

The terms are first defined in logical groupings as an aid to understanding. The groupings are arranged to bring together terms which are related to each other. The same terms are then presented as an alphabetically ordered list for ease of reference.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated are current. All standards are subject to revision, and parties entering into agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of the IEC and ISO maintain registers of currently valid International Standards.

ISO 8402: 1993, *Quality assurance - Vocabulary.*

ISO 2382-1: 1982, *Data processing - Vocabulary. Part 1 - Fundamental terms.*

ISO 2382-20: 1990, *Data processing - Vocabulary. Part 20 - System development.*

ISO/IEC 12207-1: 1994, *Software life cycle processes.*

3 Definitions

For the purposes of this International Standard, the definitions given in ISO 8402, ISO 2382-1, ISO 2382-20 and ISO/IEC 12207-1 apply, together with the following definitions.

3.1 General assessment concepts

3.1.1 **software process**: The process or set of processes used by an organization or project to plan, manage, execute, monitor, control and improve its software related activities.

3.1.2 **process assessment**: A disciplined evaluation of an organization's software processes against the process model or variant model described in this International Standard.

3.1.3 **process improvement**: Action taken to change an organization's processes so that they meet the organization's business needs and achieve its business goals more effectively.

3.1.4 **process capability determination**: A systematic assessment and analysis of selected software processes within an organization against a target capability, carried out with the aim of identifying the strengths, weaknesses and risks associated with deploying the processes to meet a particular specified requirement.

3.2 Process architecture concepts

3.2.1 **process**: A set of activities [ISO/IEC12207-1]

3.2.2 **process** (in this International Standard): A statement of purpose and an essential set of practices (activities) that address that purpose.

NOTE 1: The processes described in part 2 of this International Standard are not full, formal process definitions. Rather, the statements express high level, abstract concepts without constraining how a process may be implemented.

3.2.3 **practice**: A software engineering or management activity that contributes to the creation of the output (work products) of a process or enhances the capability of a process.

3.2.4 **base practice**: A software engineering or management activity that directly addresses the purpose of a particular process and contributes to the creation of its output. A base practice is an essential activity of a particular process.

3.2.5 **generic practice**: A process management activity that enhances the capability to perform a process. A generic practice supports the implementation or management of a process and may be applied to any process.

3.2.6 **common feature**: A set of generic practices that address an aspect of process implementation or management.

3.2.7 **capability level**: A set of common features (i.e. generic practices) that works together to provide a major enhancement in the capability to perform a process.

3.2.8 **process category**: A set of processes addressing the same general area of activity. The process categories address five general areas of activity: customer-supplier, engineering, project, support, and organization.

3.2.9 **process purpose**: A summary description of the intent or functional objectives of a process and its base practices.

3.2.10 **extended process**: A process which differs from any process contained in part 2 of this International Standard, either by having additional base practices defined for an existing process or being an entirely new process. An extended process should conform to the requirements for laid down in Annex A in part 2 of this International Standard.

3.3 Process management terms associated with generic practices

3.3.1 **defined process**: The operational definition of a set of activities for achieving a specific purpose. A defined process is characterized by standards, procedures, training, tools, and methods.

NOTE 2: A defined process is developed by tailoring the organization's standard process to fit the specific characteristics of its intended use. (See *standard process.*)

3.3.2 **well-defined process**: A process with inputs, entry criteria, tasks, validation, outputs, and exit criteria that are documented, consistent, and complete.

3.3.3 **standard process**: The operational definition of the basic process that guides the establishment of a common process in an organization. It describes the fundamental process elements that is expected to be incorporate into any defined process. It also describes the relationships (e.g., ordering and interfaces) between these process elements. (See *defined process.*)

3.3.4 **process capability**: The range of expected results that can be achieved by following a process. (See *process performance* for contrast.) [CMM Version 1.1 - CMU/SEI-93-TR-25].

3.3.5 **process performance**: A measure of the actual results achieved by following a process. (See *process capability* for contrast.) [CMM Version 1.1 - CMU/SEI-93-TR-25].

3.4 Process assessment terms

3.4.1 (assessment) sponsor: The individual, internal or external to the organization being assessed, who requires the assessment to be performed, and provides financial or other resources to carry it out.

3.4.2 (assessment) owner: The management role that takes ownership of the assessment and the assessment output, and has the authority to make the assessment happen.

3.4.3 organizational unit: That part of an organization that is the subject of an assessment. An organizational unit deploys one or more processes that have a coherent process context (q.v.) and operates within a coherent set of business goals.

NOTE 3:

An organizational unit is typically part of a larger organization, although in a small organization, the organizational unit may be the whole organization. An organizational unit may be, for example:

- a specific project or set of (related) projects;
- a unit within an organization focused on a specific lifecycle phase (or phases) such as acquisition, development, maintenance or support;
- a part of an organization responsible for all aspects of a particular product or product set.

3.4.4 assessment input: The collection of information required before a process assessment can commence. This includes:

- the assessment purpose;
- the assessment scope;
- the assessment constraints;
- the assessment responsibilities, including at a minimum the identity of the qualified assessor;
- the definition of any extended processes identified in the assessment scope;
- the identification of any additional information required to be collected to support process improvement or process capability determination.

3.4.4.1 assessment purpose: A statement, to be provided before an assessment is commenced, which defines the purpose of the assessment. The purpose may include:

- to promote an understanding of the software process;
- to support process improvement;
- to support process capability determination.

3.4.4.2 assessment scope: A statement, to be provided before an assessment is commenced, which defines:

- which organizational unit processes are to be investigated;
- the mapping from the organizational unit processes to the processes of this International Standard and extended processes that are to be assessed;
- the identification of, and justification for, the process instance(s) selected;
- the organizational unit that deploys these processes;
- the process context.

3.4.4.3 process context: Those factors that influence the judgement, comprehension and comparability of process ratings. These factors include at a minimum:

- the application domain of the products or services;
- the size, criticality and complexity of the products or services;
- the quality characteristics of the products or services (see, for example, ISO 9126);
- the size of the organizational unit;
- the demographics of the organization unit.

NOTE 4: All of these factors influence the judgement of process ratings. However, it is largely the product related factors that influence the comparability of process ratings.

3.4.4.4 assessment constraints: Restrictions placed on the freedom of choice of the assessment team regarding the conduct of the assessment and the use of the assessment outputs. Such restrictions may be positive (e.g. requiring that a specific group or individual provides information), or negative (e.g. requiring that a specific group or individual be excluded from providing information) and may include:

- specific process instances to be included or excluded from the assessment;
- the minimum, maximum or specific sample size or coverage that is required for the assessment;
- ownership of the assessment outputs and restrictions on how they may be used;
- controls on information resulting from a confidentiality agreement.

3.4.5 assessment output: The formal output from an assessment consisting of the process profile and the assessment record.

3.4.5.1 process profile: The actual and derived generic practice adequacy ratings, and the process capability level ratings for each process identified in the assessment scope.

3.4.5.2 assessment record: Any information which is pertinent to the assessment. This includes at a minimum:

- the assessment input;
- the assessment approach;
- the assessment instrument used;
- the base practice ratings for each process instance assessed;
- the date of the assessment;
- the names of team who conducted the assessment;
- any additional information required that was identified in the assessment input to support process improvement or process capability determination;
- any assessment assumptions and limitations.

3.5 Process rating concepts

3.5.1 base practice adequacy: A judgement, within the process context, of the extent to which the implemented base practice contributes to satisfying the process purpose.

3.5.2 base practice existence: A judgement, within the process context, of whether a base practice is implemented and produces some output.

3.5.3 generic practice adequacy: A judgement, within the process context, of the extent to which the implemented generic practice satisfies its purpose.

3.5.4 process instance: A single instantiation of a process, where its purpose is fulfilled in terms of taking the process inputs, performing the set of base practices and producing a set of process outputs.

3.5.5 actual rating: A rating that has been determined by assessing a specific process instance.

3.5.6 derived rating: A rating that has been determined by aggregating two or more actual ratings to derive an aggregate or average rating.

3.5.7 process capability level rating: A representation of the extent to which a process achieves the set of capabilities represented by that capability level. A process capability level rating consists of an aggregation of generic practice adequacy ratings of the generic practices within a particular capability level.

3.6 Assessment instrument concepts

3.6.1 assessment instrument: A tool or set of tools that is used throughout an assessment to support the evaluation of the existence or adequacy of practices within the scope of the assessment. It may provide assistance in collecting, recording, formalizing, processing, using, storing or retrieving information gathered during an assessment.

3.6.2 artefact: A tangible output, such as a work product, produced from the execution of an implemented process.

3.6.3 work product: An artefact associated with the execution of a practice (e.g., a test case, a requirement specification, code, or a work breakdown structure). The existence of the work product indicates that the practice is performed.

3.6.4 work product characteristic: An attribute of a type of work product that indicates the adequacy of an implementation of a practice.

3.6.5 assessment indicator: A key word or phrase that guides an assessor in recognizing characteristics of practice adequacy.

3.6.5.1 process indicator: An assessment indicator that highlights base practices or work product characteristics. Process indicators help in substantiating the rating of base practice adequacy or base practice existence and are associated with the performance of a process.

3.6.5.2 process management indicator: An assessment indicator that highlights characteristics of a particular generic practice. Process management indicators help in substantiating the rating of generic practice adequacy and are associated with the organization's ability to manage a process.

3.7 Assessors and assessor competence

3.7.1 competence: The work performance that results from effectively applying skills, knowledge and personal attributes.

3.7.2 competency: The skills, knowledge and personal attributes that enable effective work performance.

3.7.3 provisional assessor: An assessor who has not yet demonstrated competence or obtained validation of the skills, education and training appropriate to conducting assessments in accordance with the provisions in part 6 of this International Standard.

3.7.4 qualified assessor: An individual who has attained the qualifications for carrying out process assessments, as defined in part 6 of this International Standard.

3.8 Process improvement concepts

3.8.1 process improvement programme: All the strategies, policies, goals, responsibilities and activities concerned with the achievement of specified improvement goals. A process improvement programme can span more than one complete cycle of process improvement.

3.8.2 process improvement project: Any subset of the process improvement programme that forms a coherent set of actions to achieve a specific improvement.

3.8.3 process improvement action: An action planned and executed to improve all or part of the software process. A process improvement action can contribute to the achievement of more than one process goal.

3.9 Process capability determination concepts

3.9.1 (process capability determination) sponsor: The organization, part of an organization or person initiating a process capability determination.

3.9.2 target capability: That process capability which the process capability determination sponsor judges will represent an acceptable process risk to the successful implementation of the specified requirement.

3.9.3 assessed capability: The output of one or more recent, relevant process assessments conducted in accordance with the provisions of this International Standard.

3.9.4 constructed capability: A capability constructed from existing organizational elements plus sub-contractors, consultants, partners etc.

3.9.5 enhanced capability: A capability greater than current assessed capability, justified by a credible process improvement programme.

3.9.6 proposed capability: The process capability that the organization proposes to bring to bear in meeting the specified requirement. For core process capability determination, the proposed capability is the organization’s current assessed capability, whereas for extended process capability determination, the proposed capability is either an enhanced capability or a constructed capability.

4 Definitions arranged alphabetically

4.1 **actual rating**: A rating that has been determined by assessing a specific process instance.

4.2 **artefact**: A tangible output, such as a work product, produced from the execution of an implemented process.

4.3 **assessed capability**: The output of one or more recent, relevant process assessments conducted in accordance with the provisions of this International Standard.

4.4 **assessment constraints**: Restrictions placed on the freedom of choice of the assessment team regarding the conduct of the assessment and the use of the assessment outputs. Such restrictions may be positive (e.g. requiring that a specific group or individual provides information), or negative (e.g. requiring that a specific group or individual be excluded from providing information) and may include:

- specific process instances to be included or excluded from the assessment;
- the minimum, maximum or specific sample size or coverage that is required for the assessment;
- ownership of the assessment outputs and restrictions on how they may be used;
- controls on information resulting from a confidentiality agreement.

4.5 **assessment indicator**: A key word or phrase that guides an assessor in recognizing characteristics of practice adequacy.

4.6 **assessment input**: The collection of information required before a process assessment can commence. This includes:

- the assessment purpose;
- the assessment scope;
- the assessment constraints;
- the assessment responsibilities, including at a minimum the identity of the qualified assessor;
- the definition of any extended processes identified in the assessment scope;
- the identification of any additional information required to be collected to support process improvement or process capability determination.

4.7 **assessment instrument**: A tool or set of tools that is used throughout an assessment to support the evaluation of the existence or adequacy of practices within the scope of the assessment. It may provide assistance in collecting, recording, formalizing, processing, using, storing or retrieving information gathered during an assessment.

4.8 **assessment output**: The formal output from an assessment consisting of the process profile and the assessment record.

4.9 **(assessment) owner**: The management role that takes ownership of the assessment and the assessment output, and has the authority to make the assessment happen.

4.10 assessment purpose: A statement, to be provided before an assessment is commenced, which defines the purpose of the assessment. The purpose may include:

- to promote an understanding of the software process;
- to support process improvement;
- to support process capability determination.

4.11 assessment record: Any information which is pertinent to the assessment. This includes at a minimum:

- the assessment input;
- the assessment approach;
- the assessment instrument used;
- the base practice ratings for each process instance assessed;
- the date of the assessment;
- the names of team who conducted the assessment;
- any additional information required that was identified in the assessment input to support process improvement or process capability determination;
- any assessment assumptions and limitations.

4.12 assessment scope: A statement, to be provided before an assessment is commenced, which defines:

- which organizational unit processes are to be investigated;
- the mapping from the organizational unit processes to the processes of this International Standard and extended processes that are to be assessed;
- the identification of, and justification for, the process instance(s) selected;
- the organizational unit that deploys these processes;
- the process context.

4.13 (assessment) sponsor: The individual, internal or external to the organization being assessed, who requires the assessment to be performed, and provides financial or other resources to carry it out.

4.14 base practice: A software engineering or management activity that directly addresses the purpose of a particular process and contributes to the creation of its output. A base practice is an essential activity of a particular process.

4.15 base practice adequacy: A judgement, within the process context, of the extent to which the implemented base practice contributes to satisfying the process purpose.

4.16 base practice existence: A judgement, within the process context, of whether a base practice is implemented and produces some output.

4.17 capability level: A set of common features (I.e. generic practices) that works together to provide a major enhancement in the capability to perform a process.

4.18 common feature: A set of generic practices that address an aspect of process implementation or management.

4.19 **competence**: The work performance that results from effectively applying skills, knowledge and personal attributes.

4.20 **competency**: The skills, knowledge and personal attributes that enable effective work performance.

4.21 **constructed capability**: A capability constructed from existing organizational elements plus sub-contractors, consultants, partners etc.

4.22 **defined process**: The operational definition of a set of activities for achieving a specific purpose. A defined process is characterized by standards, procedures, training, tools, and methods.

NOTE 5: A defined process is developed by tailoring the organization's standard process to fit the specific characteristics of its intended use. (See *standard process*.)

4.23 **derived rating**: A rating that has been determined by aggregating two or more actual ratings to derive an aggregate or average rating.

4.24 **enhanced capability**: A capability greater than current assessed capability, justified by a credible process improvement programme.

4.25 **extended process**: A process which differs from any process contained in part 2 of this International Standard, either by having additional base practices defined for an existing process or being an entirely new process. An extended process should conform to the requirements for laid down in Annex A in part 2 of this International Standard.

4.26 **generic practice adequacy**: A judgement, within the process context, of the extent to which the implemented generic practice satisfies its purpose.

4.27 **generic practice**: A process management activity that enhances the capability to perform a process. A generic practice supports the implementation or management of a process and may be applied to any process.

4.28 **organizational unit**: That part of an organization that is the subject of an assessment. An organizational unit deploys one or more processes that have a coherent process context (q.v.) and operates within a coherent set of business goals.

NOTE 6:

An organizational unit is typically part of a larger organization, although in a small organization, the organizational unit may be the whole organization. An organizational unit may be, for example:

- a specific project or set of (related) projects;
- a unit within an organization focused on a specific lifecycle phase (or phases) such as acquisition, development, maintenance or support;
- a part of an organization responsible for all aspects of a particular product or product set.

4.29 **owner**: (See assessment owner).

4.30 **practice**: A software engineering or management activity that contributes to the creation of the output (work products) of a process or enhances the capability of a process.

4.31 process: A set of activities [ISO/IEC12207-1]

4.32 process (in this International Standard): A statement of purpose and an essential set of practices (activities) that address that purpose.

NOTE 7: The processes described in part 2 of this International Standard are not full, formal process definitions. Rather, the statements express high level, abstract concepts without constraining how a process may be implemented.

4.33 process assessment: A disciplined evaluation of an organization's software processes against the process model or variant model described in this International Standard.

4.34 process capability: The range of expected results that can be achieved by following a process. (See *process performance* for contrast.) [CMM Version 1.1 - CMU/SEI-93-TR-25].

4.35 process capability determination: A systematic assessment and analysis of selected software processes within an organization against a target capability, carried out with the aim of identifying the strengths, weaknesses and risks associated with deploying the processes to meet a particular specified requirement.

4.36 (process capability determination) sponsor: The organization, part of an organization or person initiating a process capability determination.

4.37 process capability level rating: A representation of the extent to which a process achieves the set of capabilities represented by that capability level. A process capability level rating consists of an aggregation of generic practice adequacy ratings of the generic practices within a particular capability level.

4.38 process category: A set of processes addressing the same general area of activity. The process categories address five general areas of activity: customer-supplier, engineering, project, support, and organization.

4.39 process context: Those factors that influence the judgement, comprehension and comparability of process ratings. These factors include at a minimum:

- the application domain of the products or services;
- the size, criticality and complexity of the products or services;
- the quality characteristics of the products or services (see, for example, ISO 9126);
- the size of the organizational unit;
- the demographics of the organization unit.

NOTE 8: All of these factors influence the judgement of process ratings. However, it is largely the product related factors that influence the comparability of process ratings.

4.40 process improvement: Action taken to change an organization's processes so that they meet the organization's business needs and achieve its business goals more effectively.

4.41 process improvement action: An action planned and executed to improve all or part of the software process. A process improvement action can contribute to the achievement of more than one process goal.

4.42 process improvement programme: All the strategies, policies, goals, responsibilities and activities concerned with the achievement of specified improvement goals. A process improvement programme can span more than one complete cycle of process improvement.

4.43 process improvement project: Any subset of the process improvement programme that forms a coherent set of actions to achieve a specific improvement.

4.44 process indicator: An assessment indicator that highlights base practices or work product characteristics. Process indicators help in substantiating the rating of base practice adequacy or base practice existence and are associated with the performance of a process.

4.45 process instance: A single instantiation of a process, where its purpose is fulfilled in terms of taking the process inputs, performing the set of base practices and producing a set of process outputs.

4.46 process management indicator: An assessment indicator that highlights characteristics of a particular generic practice. Process management indicators help in substantiating the rating of generic practice adequacy and are associated with the organization's ability to manage a process.

4.47 process performance: A measure of the actual results achieved by following a process. (See *process capability* for contrast.) [CMM Version 1.1 - CMU/SEI-93-TR-25].

4.48 process profile: The actual and derived generic practice adequacy ratings, and the process capability level ratings for each process identified in the assessment scope.

4.49 process purpose: A summary description of the intent or functional objectives of a process and its base practices.

4.50 proposed capability: The process capability that the organization proposes to bring to bear in meeting the specified requirement. For core process capability determination, the proposed capability is the organization's current assessed capability, whereas for extended process capability determination, the proposed capability is either an enhanced capability or a constructed capability.

4.51 provisional assessor: An assessor who has not yet demonstrated competence or obtained validation of the skills, education and training appropriate to conducting assessments in accordance with the provisions in part 6 of this International Standard.

4.52 qualified assessor: An individual who has attained the qualifications for carrying out process assessments, as defined in part 6 of this International Standard.

4.53 software process: The process or set of processes used by an organization or project to plan, manage, execute, monitor, control and improve its software related activities.

4.54 sponsor: (See assessment sponsor and process capability determination sponsor).

4.55 **standard process**: The operational definition of the basic process that guides the establishment of a common process in an organization. It describes the fundamental process elements that is expected to be incorporate into any defined process. It also describes the relationships (e.g., ordering and interfaces) between these process elements. (See *defined process*.)

4.56 **target capability**: That process capability which the process capability determination sponsor judges will represent an acceptable process risk to the successful implementation of the specified requirement.

4.57 **well-defined process**: A process with inputs, entry criteria, tasks, validation, outputs, and exit criteria that are documented, consistent, and complete.

4.58 **work product**: An artefact associated with the execution of a practice (e.g., a test case, a requirement specification, code, or a work breakdown structure). The existence of the work product indicates that the practice is performed.

4.59 **work product characteristic**: An attribute of a type of work product that indicates the adequacy of an implementation of a practice.