ISO/IEC JTC 1 N 9483 01-23-09

ISO/IEC JTC 1
Information Technology

Document Type: **Text for DTR Ballot**

Document Title: **DTR 24766 - Information Technology – Requirement Engineering Tool Requirements**

Document Source: **JTC 1 Secretariat**

Reference:

Document Status: **This document is circulated to JTC 1 National Bodies for a 3 month DTR ballot. National Bodies are asked to vote and submit their comments via the on-line balloting system by the due date indicated.**

Action ID: **VOTE**

Due Date: **2009-04-24**

No. of Pages: **51**

ISO/IEC JTC1/SC7
Software and Systems Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC7 /N4211

2009-01-23

Document Type	DTR
Title	DTR 24766 - Information Technology – Requirement Engineering Tool Requirements
Source	WG4
Project	24766
Status	Final
Reference	**N3888, N4012, Resolution 1041, N4210**
Action ID	FYI
Distribution	AG
No. of Pages	29
Note	Sent to JTC 1 for DTR processing

Address reply to: ISO/IEC JTC1/SC7 Secretariat
École de technologie supérieure – Department of Software and IT Engineering
1100 Notre Dame Ouest, Montréal, Québec Canada H3C 1K3
secretariat@jtc1-sc7.org

www.jtc1-sc7.org

ISO/IEC JTC1/SC 7/WG4 N 440

2008-11-7

ISO/IEC FCD 24766

ISO/IEC JTC1/SC 7/WG 4

Secretariat: Standards Council of Canada

Information Technology – Guide for Requirement Engineering Tool Capabilities

Warning

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: Technical Report
Document subtype:
Document stage: DTR
Document language: E

Contents

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

In exceptional circumstances, the joint technical committee may propose the publication of a Technical Report of one of the following types:

- type 1, when the required support cannot be obtained for the publication of an International Standard, despite repeated efforts;
- type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;
- type 3, when the joint technical committee has collected data of a different kind from that which is normally published as an International Standard ("state of the art", for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC TR 24766, which is a Technical Report of type 2, was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 7,***Software and Systems*** *Engineering*.

Introduction

Requirements Engineering (RE) is a major activity within the systems and software engineering life cycles. This activity must be carried out in a comprehensive manner to ensure that a complete set of user needs and requirements are captured. These user needs and requirements are transformed into a validated set of technical requirements and managed throughout the life cycle using the requirements engineering process activities. Requirements engineering tools are used to support many requirements engineering and related life cycle activities. Requirements engineering processes are identified in ISO/IEC 15288: 2008 – Systems Engineering – System life cycle processes and ISO/IEC 12207:2008 - Information Technology – Software life cycle processes.

IS 15288 and IS 12207 describe a set of requirements engineering processes, activities and tasks to be performed when acquiring or developing systems and software. However, these documents do not address the RE tool capabilities users can expect in order to support a RE process and other related life cycle activities.

Many requirements engineering processes are human activities that in the current state of the practice tools cannot perform and may never be able to perform by a tool. But wherever possible, a tool should support these human activities through the facilitation of documentation capture, content management, distribution, discussion forums, and decision support tools.

This guide describes capabilities of requirement engineering tool to benefit the groups of people that acquire, supply, develop, operate, and maintain a RE process. The outcomes of this guide are to:

Support RE Personnel involved in the execution of one or more RE activities to:

- Obtain a better understanding of the relationship between the activities in which they are involved and RE tool capabilities
- Identify processes or activities that can be improved through better support by a RE tool
- Have an objective basis for a better comparison, evaluation and assessment of RE tools

Support People involved in the purchase of RE tools to:

- Review RE services that can contribute to RE process improvement
- Identify criteria for selecting RE tools

Support RE Tool Vendors to:

- Provide RE tools consistent with the International Standards IS 15288, IS 12207, IS 15940, and IS 14102

Information Technology – Guide for Requirement Engineering Tool Capabilities

1 Scope

Requirements Engineering (RE) is an essential process of the systems and software engineering life cycles. RE has been established as an ISO/IEC standard life cycle process in both ISO/IEC 15288: 2008 – Systems Engineering – System life cycle processes and ISO/IEC 12207:2008 - Information Technology – Software life cycle processes.

A method for evaluating general CASE tools has been adopted as an ISO/IEC standard in ISO/IEC 14102:1995 - Information Technology - Guideline for the evaluation and selection of CASE tools, which details a set of evaluation criteria for tools without referencing a specific activity or service area.

It is the scope of this International Standard to supplement ISO/IEC 14102:1995 by providing guidance on desirable capabilities of requirements engineering tools.

2 Normative references

- ISO/IEC 9126-1:2001 – Software Engineering – Product quality – Part1: Quality model
- ISO/IEC 15288:2008 – Systems Engineering – System life cycle processes
- ISO/IEC 12207:2008 – Information Technology - Software life cycle processes
- ISO/IEC 14102:2008 – Information Technology - Guideline for the evaluation and selection of CASE tools
- ISO/IEC TR 19759 – Software Engineering – Guide to the software engineering body of knowledge
- ISO/IEC 25000:2005 – Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE
- ISO/IEC 25030:2007 - Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Quality requirements

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1

Activity

A set of actions that consume time and resources and whose performance is necessary to achieve, or contribute to, the realization of one or more outcomes

3.2

Manage or Management

To provide storing, editing capabilities, track history of edition, versioning control, author identification, change management, time stamping, user notification for content changes, security rights control.

3.3

Functional requirements

Requirement that specifies a function that a system or system component must be able to perform

[ISO/IEC 25000]

3.4

Non-functional requirements

See quality requirements

3.5

Quality requirement

Quality requirement characterizes the capability of a product to satisfy the stated and implied needs when used under specific conditions.

NOTE: the term non-functional requirement is often used instead of quality requirement. We choose to use the quality requirement here to align with other current standards, particularly IS 25000 series.

3.6

Performance requirements

Performance requirements define how well the system must perform its functions. Performance requirements must be measurable and quantitative. E.g. any transaction shall be performed within 2 seconds from initiation of the transaction.

3.7

Process

A set of interrelated or interacting activities which transforms inputs into outputs

[ISO/IEC 12207:2008]

3.8

Requirements attributes

A set of properties associated with requirements

3.9

Stakeholder

A party having a right, share, or claim in a system or in its possession of characteristics that meet that party's needs and expectations

[ISO/IEC 15288:2008]

3.10

Stakeholder equity

The degree of the share or claim a stakeholder has in the system of interest or a portion of the system of interest.

3.11

User requirements

Expression of a perceived need from user that something be accomplished or realized

[ISO/IEC 25000]

NOTE ISO/IEC 25000 defines the term, requirements. User requirements is a requirements issued by user

4 Requirements engineering process

4.1 Overview

A requirements engineering tool should facilitate and support the systematic managing of requirements throughout the project life cycle. The tool should also support the related activities in the context of the requirements engineering process. The following sub-topics characterize the processes that a RE tool would need to address.

4.1.1 Requirements elicitation

Requirements elicitation is the process of seeking, uncovering, acquiring, and elaborating requirements. Requirements are elicited rather than just captured or collected. This implies there are discovery, emergence, and development elements to the elicitation process. Requirements elicitation is an iterative activity and benefits from continuous communication and validation with stakeholders.

4.1.2 Requirements analysis

Requirements analysis involves refining the requirements by decomposing high level requirements into details, building prototypes, evaluating feasibility, analyzing overlaps or conflicts between requirements, and negotiating priorities. The goal is to develop requirements of sufficient quality and detail to reflect the stakeholders' needs.

4.1.3 Requirements specifications

Requirements specification deals with documenting the requirements in a consistent and reviewable way. Documentation includes the functions and capabilities that a software system must provide and the constraints that it must respect. Requirements specification is the basis for all subsequent software life cycle activities including project planning, design, and coding, as well as the foundation for system testing and user documentation [IS 19759].

4.1.4 Requirements and product validation

Validation is the process of evaluating a system or software to determine whether it meets stakeholder requirements. It is performed by examination and through the provision of objective evidence that the requirements for a specific intended use or application have been fulfilled [IS 19759].

4.1.5 Requirements and product verification

Verification is the process of evaluating a system or software to determine whether it properly reflects the specified requirements. It is performed by examination and through the provision of objective evidence that specified requirements have been fulfilled [IS 19759].

4.1.6 Requirements management

Requirements management in conjunction with change management ensures that the requirements remain aligned with the developed product. Requirements management concerns the collection, analysis, and validation and verification of requirements with all the communications and negotiations inherent in working with people.

5 Requirements engineering tool capabilities

5.1 Overview

RE tool allow requirement engineering and management actions to be automated, reducing the cognitive load on the stakeholder. This section provides a list of required capabilities for an RE tool. The capabilities are organized according to the software requirement activities listed in IS 12207, IS 15288, TR 19759 (SWEBOK).

This list can be used for:

- Evaluating and choosing an RE tool
- Matching process to specific RE tool capabilities

5.2 Requirements elicitation

5.2.1 Overview

The requirements expressed in the project scope must address all essential business and user needs. The RE tool should be able to support in identifying stakeholder, capturing and tracing of the business/user requirements, functional requirements, and the quality requirements.

5.2.2 Requirements capture

The RE tool should support requirements capture by allowing the user to:

- Storing and managing the documentation from interviews, workshops, and observation
- Storing and managing stakeholder contact lists, voting, comments, user security, group notifications
- Tracing requirements and generating trace reports
- Creating hierarchical relationships between requirements
- Importing text and graphics from applications (e.g., Rich Text Format (RTF), MS word, MS excel)
- Updating existing linked documents from new or changed versions of the source documents without having to re-establish traceability links
- Storing and managing attributes for classifying or categorizing requirements during identification
- Storing and managing attributes in a variety of formats (e.g., text, enumerated, binary, graphics, descriptions, attachments), that can be associated with each requirements
- Using tool generated inherent attributes (e.g., unique requirement identification, author, time date, requirements change history)
- Using flexible search options for requirements by word or attributes (e.g., requirements identifier, words in text files, user and tool generated attributes)
- Managing the replacement or updating requirements by manual or electronic import
- Using flexible user programming language to develop reports for display or generating documents from the tool

5.2.3 Capturing “as-is” and “to-be” system elements

Requirements expressed in the project scope must not exclude any essential business, user, functional, and quality requirements. The RE tool should support capturing "as-is" and the "to-be system elements as follows:

- Storing and managing graphics and text (e.g., architecture, functional decomposition, Work Breakdown Structure (WBS))
- Storing and managing user definable attributes and additional information associated with a given requirement (e.g., stakeholders, business process, activity, tasks, policy, constraints)

- Flexible tracing (e.g. forward and backward tracing, one to many and many to one, bi-directional tracing of text to text, text to graphics, graphics to graphics, elements within graphics, tables and cells within a table)
- Bi-directional tracing of additional requirements and link between them such as requirements to requirements, or requirements to text (representing trade studies) to derived requirements
- Bi-directional tracing of performance requirements to system elements (e.g., weight, cost, throughput)
- Bi-directional tracing of the allocation of portions of that performance requirements to system elements
- Bi-directional tracing of requirements to system elements (e.g., from the implementation back to the requirements or from the requirements down to the system element)
- Bi-directional tracing of rationale, assignments, criticality, test and validation and many other issues to the requirements, allocation, and the system elements

5.2.4 Stakeholder and requirements traceability

All functional and quality requirements should be traceable back to specific user, stakeholder, and business requirements. The RE tool should support traceability between them as follows:

- Storing and managing the identification and documentation of stakeholders and their roles and responsibilities
- Tracing and managing documentation for the equity of the stakeholders in the system of interest or portions of the system of interest
- Flexible searching and reporting of inconsistencies such as unlinked requirements or system elements (orphans)
- Bi-directional tracing of source needs and requirements
- Tracing user defined attributes for requirements that was fulfilled, how it was done, and who was responsible
- Displaying of traceability in graphical and textual form
- Flexible exporting of traceability matrix in both textual and graphical forms (e.g. Comma Separated Value (CSV), Excel, Rich Text Format (RTF), eXtensible Markup Language (XML))

5.2.5 Goal-oriented scenarios and high-level modeling

Scenarios, models, and simulations can be used to describe the specific interaction between a user and a system to accomplish some goal. The RE tool should support goal-oriented scenarios and modeling as follows:

- Storing and managing user defined or tool provided templates for goal-oriented scenarios (e.g., simulations and modeling business scenarios, strategic issues)
- Storing and managing user defined scenarios
- Storing and managing for the evaluation requirements based on business goals

5.2.6 Elicitation templates and checklists

Templates and checklists provide a consistent structure for recording the requirements descriptions and other requirements related information. The RE tool should support elicitation templates and checklists as follows:

- Storing and managing user defined or tool provided templates for elicitation (e.g. Quality Function Deployment (QFD) or Goal Question Metric (GQM))
- Storing and managing user defined or tool provided elicitation checklists
- Storing and managing user defined or tool provided prioritization forms

5.2.7 Prototyping

Prototyping can be used to explore and validate requirements. The RE tool should support prototyping as follows:

- Presenting information in a graphical user interface (GUI)

5.2.8 Importing and exporting to and from other sources

Requirements should be imported from, or interfaced to users, hardware, and other software systems. The RE tool should support Importing and exporting to and from other sources as follows:

- Importing and exporting to and from other tools (e.g. verification, design, spreadsheets, project management, documents)
- Importing and exporting to and from various standard file formats (e.g., Comma Separated Value (CSV), eXtensible Markup Language (XML))

5.2.9 Elicitation documentation

The output from the entire requirements elicitation tasks should be stored, retrieved, and edited in various formats. The RE tool should support elicitation documentation as follows:

- Storing and managing non-textual requirements in the specified format (e.g., bit-mapped graphics, vector graphics, tables, equations, or formal logic notations)
- Storing and managing textual requirements statements using basic text processor and spell checker

5.3 Requirements analysis

5.3.1 Overview

Requirements analysis includes decomposing high-level requirements into details by building prototypes, evaluating feasibility, and negotiating priorities. The RE tool should be able to support in decomposing requirements into functional and quality requirements, and in analyzing requirements feasibility and risk.

5.3.2 Functional requirements analysis

Functional requirements are a statement of required functionality or a behavior that a system will exhibit under specific conditions. The RE tool should support functional requirements analysis as follows:

- Storing and bi-directional tracing of identified user requirements to functional requirements
- Hierarchical structuring and identification scheme for the elaboration of requirements
- Checking spells and grammars on requirements statements (e.g. word spelling check, passive vs. active voice)
- Bi-directional tracing of requirements analysis to system implementation (e.g., architecture design, Work Breakdown Structure (WBS))

5.3.3 Quality requirements analysis

Quality requirements analysis involves significant architectural and design decisions. The RE tool should support quality requirements analysis as follows:

- Storing and managing quality requirements in quality attributes, policies, or constraints
- Hierarchical structuring and identification scheme for the elaboration of requirements.
- Bi-directional tracing of quality requirements to source requirements or user requirement.
- Checking spells and grammars on requirements statements (e.g. word spelling check, passive vs. active voice)
- Bi-directional tracing of requirements analysis to system implementation (e.g., architecture design, Work Breakdown Structure (WBS))
- Storing and managing the results or the rationale of quality attributes trade-off.

5.3.4 Feasibility analysis

Feasibility analysis evaluates the feasibility of implementing each requirement at acceptable cost and performance. It also identifies technical obstacles. The RE tool should support feasibility analysis as follows:

- Storing and generating user defined or tool provided checklists or templates for various analysis, (e.g., technical, economical, and operational analysis)
- Storing and managing rationale of feasibility analysis

5.3.5 Modelling

Modeling analysis depicts the requirements at a high level of abstraction. Such models include data flow diagrams, entity relationship diagrams, or UML diagrams. The RE tool should support modeling analysis as follows:

- Importing and exporting to and from modeling tools and displaying the results
- Storing and displaying context diagrams, conceptual domain models, and other high level models (e.g. Goal models, Object models, Task models)
- Storing and displaying analysis of requirements in graphical form (e.g., Unified Modeling Language (UML), Data Flow Diagram (DFD))

5.3.6 Prototyping

When developers or users are not certain about the requirements, constructing a prototype make the concepts and possibilities more tangible. The RE tool should support prototyping as follows:

- Presenting information in a graphical user interface (GUI)
- Demonstrating algorithm

5.3.7 Attribute analysis

User defined attributes such as risk, priority, and cost provides metrics for tracking requirements based on project needs. These attributes are assignable to each requirement. The RE tool should support tracking of any attribute(s) user defined or tool provided as follows:

- Storing and managing attributes in various formats (e.g. text, numeric, graphics, attachments)
- Detecting and flagging missing attributes
- Storing, sorting, grouping and ordering of attributes
- Managing changes to attributes

5.3.8 Requirements refinement

Requirements refinement identifies the hierarchical relation of requirements and allocates the requirements to subsystems. The RE tool should support requirements refinement as follows:

- Hierarchical structuring and identification scheme for the elaboration of requirements
- Bi-directional tracing of requirements to parent requirements or user requirement
- Bi-directional tracing of requirements to child requirements or design elements
- Bi-directional tracing of requirements to verification and validation case element
- Storing and managing the allocation rationale (e.g., how it was done, who was responsible)
- Supporting a flexible set of user defined or tool provided queries on requirements

5.3.9 Risk analysis

Risk analysis provides a standard approach to identify and document potential risks, and propose strategies for mitigating them. The RE tool should support risk analysis as follows:

- Exchanging information pertinent to risk analysis tools with external risk analysis tools
- Associating requirements with risks as a means of mitigation

5.3.10 Decision methods

Multiple stakeholders decide how to manage and resolve conflicting requirements. The RE tool should support decision methods as follows:

- Supporting an interface to a possibly separate terminology(glossary) or domain knowledge repository
- Checking requirements' conformity to predefined templates
- Storing and managing the list of conflicting requirements
- Storing and managing the information and rationale for resolved results

5.3.11 Requirements analysis artifacts

The output from the entire requirements analysis task should be stored, retrieved, and edited in various formats. The RE tool should support requirements analysis documentation as follows:

- Storing requirements analysis results in both text and graphical formats (e.g., Rich Text Format (RTF), eXtensible Markup Language (XML),JPEG)
- Storing, managing, and exporting requirements analysis results using basic text processor and spell checker
- Storing and managing non-requirements items, such as constraints on resources (e.g., time, budget, people) and considerations on design

5.4 Requirements specification

5.4.1 Overview

The specification states the functions and capabilities that a software or a system must provide and the constraints that it must respect. The RE tool should be able to support for all requirements be specified and documented in a consistent, accessible, and reviewable way.

5.4.2 Developing specification

The specification should describe as completely as necessary the software or the system's behavior under various conditions. The RE tool should support developing specification as follows:

- Generating traceability reports the identify exceptions in user requirements in the analysis phases to functional requirements
- Generating reports that identifies the exceptions for each functional requirement
- Generating and exporting templates in a variety of standard formats (e.g., text, Rich Text Format (RTF), eXtensible Markup Language (XML))

5.4.3 Traceability analysis

During requirements analysis, changes must be documented. Traceability establishes and tracks the linkage between each requirement and its source, proving insight into the overall project status. The RE tool should support traceability analysis as follows:

- Generating bi-directional traceability of requirements to requirement sources or other artifacts.
- Displaying a graphical representation of the requirements hierarchy
- Generating a report for missing and/or orphan requirements
- Generating a report that identifies changes from/to a requirements throughout the development life cycle

- Generating reports of traceability attributes (e.g., category, number of approval or unapproved, number of changes, number of pending)
- Generating and exporting traceability matrix tables
- Generating a status report for the status of each requirement based on user defined or tool provided attributes.

5.4.4 Requirements specification documentation

The output from the entire requirements specification task should be stored, retrieved, and edited in various formats. The RE tool should support requirements specification documentation as follows:

- Exporting document in standard formats (e.g. Rich Text Format (RTF), MS-word)
- Generating user defined or tool provided templates. (e.g. military standard formats)
- Checking the document through spell checking, grammar checking, data dictionaries, and acronym tables.
- Generating presentation-quality charts and graphs in standard formats (e.g. PIC, PNG, JPEG, GIF)
- Generating the output of the specification in a finished form, including: page security markings, graphics or figures, user definable tables, and indexes.
- Synchronizing changes between the RE tool and formatted document (complete loop between RE tool and formatted document)

5.5 Requirements verification and validation

5.5.1 Overview

Requirements verification and validation ensure that the requirements are correct and that the stakeholders agree to these requirements. Verification and validation activities are not a single discrete phase but an iterative process performed during elicitation, analysis, and specification. The RE tool should be able to support the various tests and evaluation means in verifying and validating the requirements.

5.5.2 Review and inspection

Reviewing and inspecting requirements is used for identifying ambiguous requirements and for validating requirements. The RE tool should support review and inspection as follows:

- Accessing to large numbers of stakeholders and users for review and comment of requirements
- Providing user and group security for each requirement (e.g., access, read only, read-write)
- Generating reports on requirements by user defined sort criteria (e.g., attributes, key words)
- Managing user comments and review edit.
- Storing and managing discussion threads linked to a specific requirement
- Maintaining history of all changes for each requirement
- Storing and managing the review and inspection results **(**e.g., how it was done and who was responsible)

5.5.3 Verification

Verification is a more formal and rigorous way of ensuring that the final product meets the specified requirements. The RE tool should support verification as follows:

- Tracing of verification cases to the requirements
- Generating exception reports on requirements that do not have verification plan cases and verification plan cases that are not linked to requirements
- Supporting review and inspection of verification plans and procedures
- Tracing of verification cases to the requirements

- Tracing of the verification cases to verification procedures
- Storing and managing verification plans and verification procedures
- Generating exception reports on verification plan cases that do not have verification procedures and verification procedures that are not linked to verification plan cases
- Providing standard format for interfacing to verification tools

5.5.4 Validation

Validation is a way of ensuring that the final product meets the users or stakeholder needs under expected usage conditions. The RE tool should support validation as follows:

- Tracing of validation cases to the user requirements
- Generating exception reports on user requirements that do not have validation plan cases and validation plan cases that are not linked to user requirements
- Supporting review and inspection of validation plans and procedures
- Tracing of validation cases to the user requirements
- Tracing of the validation cases to validation procedures
- Storing and managing validation plans and validation procedures
- Generating exception reports on validation plan cases that do not have validation procedures and validation procedures that are not linked to validation plan cases
- Providing standard format for interfacing to validation tools

5.5.5 Defining acceptance criteria

Acceptance verification and end to end validation evaluates whether the product satisfies its documented requirements (verification) and whether it is fit for use in the intended operating environment (validation). As part of the acceptance verification plan, the acceptance criteria must be established. The RE tool should support defining acceptance criteria as follows:

- Storing and managing the acceptance criteria
- Supporting review and inspection of acceptance criteria
- Generating exception reports on requirements that do not trace to verification acceptance criteria and verification acceptance criteria that do not trace to requirements
- Generating exception reports on goals that do not trace to acceptance criteria and acceptance criteria that do not trace to goals
- Generating exception reports on system interfaces or requirements that do not trace to standards and standards that do not trace to system interfaces or requirements
- Generating exception reports on requirements that do not trace to a detailed design element and detailed design elements that to not trace to a requirement
- Generating exception reports on detailed design elements that do not trace to an implementation element and an implementation element that is not linked to a design element

5.5.6 Baseline of the requirements

Once the requirements artifacts are in- process validated and approved by the user and/or stakeholders, each artifact is deemed a baseline. A baseline is the agreed upon and approved set of requirements. The RE tool should support baseline of the requirements as follows:

- Storing and managing a baseline document
- Tracing the baseline requirements to a specific product release (specification tree)
- Version controlling of baseline requirements

- Generating reports that compare and contrast baselines versions
- Providing read and write protection of baseline products

5.6 Requirements management

5.6.1 Overview

Requirements management is a life cycle activity (lasting beyond the development phase to retirement and disposal). The RE tool should be able to support for monitoring the changes and the maintenance of the requirements in insuring that the requirements accurately reflect the product.

5.6.2 Identification of configuration items of requirements

After the baseline requirements have been established, changes may be made only through the defined change process. Every version of the requirements needs to have a unique identifier to avoid confusion between drafts and baselines and between previous and current versions. The RE tool should support Identification of configuration items of requirements as follows:

- Managing version identification (e.g., version number, date, time of creation or revision)
- Searching and tracing of specific versions of requirements

5.6.3 Requirements change management

Change management tracks changes to requirements and ensures that approved changes are communicated to all affected stakeholders. The RE tool should support requirements change management as follows:

- Providing check in and out capabilities of the baseline to add, change, update requirements
- Providing user defined or tool provided flexible search criteria for all requirements artifacts affected by the changed requirements
- Managing the change status (e.g., acceptance or approval, for pending changes)
- Storing and generating electronic change request form and templates
- Accessing to and distributing electronic forms and electronic mail notification
- Generating requirements modification history reports that should include, change author, change id, date of change, change status such as accepted, rejected or pending

5.6.4 Traceability management

Traceability management allows one to follow the life of requirements both forward and backward, and from origin through implementation. The RE tool should support traceability management as follows:

- Storing and managing references and links to source documents
- Tracing requirements to other sources (e.g., contractor's requirements, design, and verification procedure documents)
- Generating reports that compare current and previous versions when a source document is updated
- Storing historic information on the requirements by various requirements attributes (e.g., time, owner)
- Maintaining a history of requirements changes, who changed it, when it was done, why it was done.

5.6.5 Risk management

Risk management is an approach to identifying and documenting risk factors, evaluating their potential severity, and proposing strategies for mitigating them. Risk management relates to requirement engineering in two ways: Risks may be raised against requirements, and requirements may mitigate risks.

While it is not the role of the RE tool set to manage risks, it is important that the relationship between requirements and risks be maintained. The RE tool should support risk management as follows:

- Exchanging information pertinent to risk management with external risk management tools
- Maintaining relationship of requirements to risks raised and risks mitigated

5.6.6 Project management

Project management applies to managing resources, schedules, and commitment of the requirements. While it is not the role of the RE tool set to perform project management, project management needs to keep track of the status of requirements. The RE tool should support project management as follows:

- Recording, tracking, and reporting the status of the overall requirements management process
- Exchanging information pertinent to project management with external project management tools

5.7 Other tool capabilities

5.7.1 Overview

The output of each RE activity and task is the basis for all subsequent systems and software life cycle activities including project planning, design, and coding, as well as the foundation for testing and user documentation. Therefore, integrating RE tools into systems and software development environment is important for automated RE support.

5.7.2 Collaborative access

In a distributed development environment, the synchronization of developers and access permissions to the requirement artifacts are key functions for ensuring integrity. The RE tool should support role-based collaborative access as follows:

- Maintaining data integrity under simultaneous multi-user access
- Assigning roles to users and groups of users
- Assigning role-based access to sets of requirements
- Assigning role-based access to attributes of requirements
- Assigning role-based access to tasks
- Maintaining the role-based access schema

5.7.3 RE tool administrative information

In many projects, versions can be distributed on different hardware platforms. In this case, administrative information will be needed to monitor system wide use. The RE tool should support this information as follows:

- Providing project Information (e.g., project size, concurrent users, and number of analyst)
- Providing user information indicating single user or multiple concurrent user environment
- Providing platform information indicating which platforms and operating systems support the tool
- Providing database information indicating which database it uses (proprietary or commercial)
- Providing hardware and/or software information indicating hardware and/or software configuration requirements (e.g., memory requirements, CPU requirements, disk space requirements, browsers)

5.7.4 Graphical user interface

To improve the usability of the tool, various capabilities of GUI should be supported. The RE tool should support graphical user interface as follows:

- Providing standard windows, Unix, Mac environment capabilities (e.g., multi-widows, open multiple files concurrently)

- Providing multiple windows or views into the tool (e.g., a change in one view will automatically reflect in all other views)
- Providing an interactive graphical input and manipulation of data
- Enabling web browser interface
- Providing edit, undo, cut, and paste function
- Providing an ability for doing one thing while users are looking at another (e.g., run a report and look at a requirements at the same time)

6 Quality characteristics of requirements

6.1 Overview

Setting up a target quality criteria (or quality characteristics) can help in the early detection and correction of problems. The quality characteristics can also be used as a metric in evaluating the requirements.

This section describes the sets of quality characteristics of requirements and how the RE tool can support in defining and utilizing them.

6.2 Quality characteristics of requirements artifacts

The associated quality characteristics for individual requirement statements are denoted as follows:

- Singularity – each requirement statement addresses a single concern at an agreed granularity
- Correctness – requirement statement corresponds to customer request
- Conciseness – brief meanings, no unnecessary information
- Non-ambiguity – requirement statement has a single interpretation
- Traceability – requirement statement is upwards and downwards traceable to surrounding artifacts
- Compliancy – whether or not the requirement is mandatory
- Priority – the urgency of the requirement is determined
- Verifiability – requirement statement is testable
- Abstraction – requirement statement is expressed at the proper level of detail
- Feasibility – requirement statement is technically feasible
- Stability – requirement statement is not recently subject to frequent changes
- Sufficiency – requirement statement is sufficiently covered in each dimension of downwards traceability (e.g. satisfaction traceability down to lower requirements, and verification traceability across to tests)
- Necessity – requirement statement is necessary in each dimension of upwards traceability in which it engages (e.g. the requirement is necessary to satisfy higher requirements)

The associated quality characteristics for sets of requirement statements are denoted as follows:

- Consistency – absence of conflict among a set of requirements
- Uniformity – consistent use of terms across a set of requirements
- Modularity –cohesion among a set of requirements (i.e. requirements that belong together are together)
- Scoped – requirement set within the defined scope
- Uniqueness – all requirement statements in set are significantly different (i.e. no similar or identical requirements)

- Completeness – the set of requirements represents a complete definition of the product
- Understandability - the set of requirements is comprehensible as a document
- Affordability – the set of requirements is attainable with respect to schedule and cost

6.3 Tool capabilities for quality characteristics

6.3.1 Overview

The RE tool should support the effective use of quality characteristics in classifying and assessing the requirements as follows:

- Supporting stakeholders in identifying, specifying and quantifying key quality characteristics of requirements
- Providing services to specify, measure and review requirements using the identified quality characteristics
- Allowing the user to ensure that the requirements have the necessary quality characteristics needed by the stakeholders
- Allowing the user to record and report the status of each requirement with respect to each quality characteristic
- Allowing the user to record and report rationale associated with the assessment of a requirement statement with respect to each quality characteristic

In addition to the above capabilities, the following attribute-specific capabilities should be provided:

6.3.2 Singularity

A requirement statement should address a single aspect. A single requirement may consist of multiple elements, such as a statement of capability, one or associated modes of operation and one or more associated levels of performance. The RE tool should support detection of multiple elements as follows:

- Providing a numbering or naming scheme for uniquely identifying a requirement or element of a requirement
- Indicating the presence of multiple statements that combines two or more elements

NOTE: Completeness and singularity need to be balanced, and modularity has to be used to ensure that related requirements are grouped.

6.3.3 Correctness

The correctness of requirement with respect to stakeholder needs is addressed by the requirements validation activity. The RE tool should support validation as follows:

- Supporting a validation process in which requirements can be reviewed with respect to stakeholder needs

NOTE: The use of mathematical formalisms can address certain issues of correctness of specifications against requirements. There is no known method of automatic correctness checking of requirements against stakeholder needs.

6.3.4 Conciseness

To achieve conciseness, requirements should be expressed as single sentences with a small number of sub-clauses. The RE tool should support validation as follows:

- Identifying requirements that break the conciseness rule.

6.3.5 Non-ambiguity

The RE tool should support the detection of ambiguity as follows;

- Providing a user-manageable list of inherently ambiguous terms, expressions, and syntactic structures
- Highlighting to the user occurrences of ambiguous terms, expressions and syntactic structures in requirements statements
- Allowing the user to record exceptions in the use of listed ambiguous terms, expressions and syntactic structures (e.g. state the rationale for allowing the use of such terms in particular circumstances.)

NOTE: Some requirements specification languages would allow a tool to support automatic detection of lexical, syntactic, and semantic errors over and above those detectable by searching for words and phrases.

6.3.6 Traceability

The primary purpose of traceability is to enable impact analysis. It is not only requirements that engage in such traceability; it may be other development artifacts, such as test cases, issues, and risks. It should be possible to discover what artifacts may be impacted if a selected artifact were to change. The RE tool should support both backward and forward traceability. Backward traceability links each artifact explicitly to what impacts it in previous stages of development; for instance a system requirement back to the stakeholder requirements it is intended to satisfy. Forward traceability links each artifact to what it impacts in subsequent stages of development; for instance, a system requirement to the tests that are designed to validate it. The RE tool should support backwards and forwards traceability as follows:

- Allowing the user to define valid traceability relationships (i.e. which artifacts in which documents may be linked)
- Allowing the user to characterise traceability relationships by name and cardinality
- Allowing the user to create valid backwards traceability from an artefact to those that impact it
- Allowing the user to create valid forwards traceability from an artefact to those that it impacts
- Allowing the user to report on multi-level backwards and forwards impacts from selected artifacts

6.3.7 Compliancy

Compliance refers to an attributes of a requirement which expresses the level or degree of compliance to a standard. There are two levels of compliance characteristics to standards:

Mandatory: The standards are applicable to all requirements defined in sentences containing the word "shall". These requirements must be verified.

Goal or objective: The standards are not mandatory but need rationale and documentation if requirements are not met by the standards. The RE tool should support for:

- Allowing the user to define a number of compliance levels (e.g. mandatory or desirable)
- Allowing the user to classify each requirement with a compliance level

6.3.8 Priority

Establishing the relative priority of each requirement enables users to develop against the highest priority requirements first. The RE tool should support requirements prioritization as follows:

- Providing a capability for specifying the importance of individual requirements

6.3.9 Verifiability

A requirements is verifiable if there exists method by one can test that the product meets the requirements. If a method cannot be devised, the requirement should be removed or revised. The RE tool should support verifiability as follows:

- Providing a user-manageable list of inherently vague terms, expressions, and syntactic structures that render a requirement unverifiable
- Highlighting to the user occurrences of vague terms, expressions and syntactic structures in requirements statements
- Allowing the user to record exceptions in the use of listed vague terms, expressions and syntactic structures (e.g. state the rationale for allowing the use of such terms in particular circumstances.)
- Allowing the user to associate a selection of verification methods with each requirement
- Allowing the user to associate a rationale against the set of test methods associated with each requirement

6.3.10 Abstraction

Each requirement should be positioned at an appropriate level within the system hierarchy. For example, level I may indicate a top or system level requirements. Level II may be the segment level requirements. The RE tool should support abstraction as follows:

- Identifying levels within the system hierarchy
- Allowing the positioning of requirements within appropriate levels
- Supporting a review process that checks the appropriate level of abstraction of a requirement

6.3.11 Feasibility

The RE tool should support feasibility as follows:

- Supporting a requirements review process that assesses the feasibility of each requirement

6.3.12 Stability

The earlier a requirements stabilizes, the less the test plans change. An unstable requirement is one that is changed frequently. The RE tool should support abstraction as follows:

- Identifying and monitoring the requirements with frequent changes and outdated test cases

6.3.13 Sufficiency

The RE tool should support sufficiency as follows:

- Allowing the user to capture a rationale for the sufficiency of the set of downwardly linked requirements that satisfy each requirement
- Supporting a review process that determines the sufficiency of the set of downwardly linked requirements that satisfy each requirement

6.3.14 Necessity

The RE tool should support necessity as follows:

- Allowing the user to capture a rationale for the necessity of each requirement for each upwardly linked requirement it serves to satisfy
- Supporting a review process that determines the necessity of each requirement for each upwardly linked requirement it serves to satisfy

6.3.15 Consistency

The challenge here is identifying conflict among a large number of requirements. Consistency is easier to check among small sets of related requirements. Thus, knowing which requirements are related in some way is useful. The RE toolset should support the detection of conflicting requirements as follows:

- Allowing the user to identify related requirements through detection of keywords
- Allowing the user to identify related requirements through classification of requirements in multiple ways

- Presenting sets of related requirements for conflict review
- Allowing the user to identify conflicts traced to the sets of requirements concerned
- Allowing the user to track the resolution of conflicts

6.3.16 Uniformity

The RE tool should support uniformity of language as follows:

- Allowing the user to define and manage a glossary of terms
- Indicating terms used in requirements that are and are not in the glossary

NOTE: There are many dimensions to uniformity; For instance, uniformity in the way in which requirements are expressed in terms of grammar; uniformity in the level of detail expressed in requirement statements. One indication of non-uniformity in the level of detail of requirements is the expansion factor for each requirement as it is decomposed into requirements to the level below. If the expansion factor is very different across requirements, it perhaps indicates an imbalance.

6.3.17 Modularity

The RE tool should support modularity as follows:

- Allowing requirement sets to be organized hierarchically
- Organizing the requirements by level

6.3.18 Scoped

The RE tool should help ensuring that the requirements are within the project scope. The project scope is monitored whenever requirements are identified, analyzed, and baselined. For example, where a requirements is refined to have external interfaces, scope decisions need to be made about the system boundary or any services to be performed related to the interface. The RE tool should support scoping as follows;

- Allowing the user to identify external interface and system boundary
- Allowing the user to check the requirements are within system boundary

6.3.19 Uniqueness

The RE tool should assist by associating more than one product with the same requirements in a traceability matrix. If there are duplicate requirements, a tool should trace them to see if they are really the same. If requirements are related, but located in separate areas of the repository or document, they should be cross-referenced. The RE tool should support uniqueness as follows;

- Allowing the user to trace the requirements along requirements hierarchy
- Allowing the user to search the requirements by requirements attributes
- Allowing the user to report on same requirements

6.3.20 Completeness

Each requirement must fully describe the functionality to be delivered. It must contain all the necessary information for the developer to design and implement the functionality. The RE tool should support as follows:

- Highlighting any unresolved or incomplete requirements and tracking such requirements

6.3.21 Understandability

Understandable requirements are presented in a style allowing the users to review them without any misconception or misinterpretation. The RE tool should support understandability as follows:

- Separating quality requirements from functional requirements
- Organizing the user requirements by business process or scenario
- Organizing the requirements by level

6.3.22 Affordability

The affordable requirements can be technically and operationally implemented within existing constraints such as time and money. The RE tool should support affordability as follows;

- Allowing the user to trace to the requirements constraints
- Storing the infeasible requirements into a repository for future usage

ANNEX A Requirements quality characteristics and software quality characteristics

Users tend to focus on specifying their functional or behavioral requirements. But users also have expectations about how well the product would work. Characteristics that fall into this category include how easy it is to use, how quickly it runs, how often it fails, and how it handles unexpected conditions. Such characteristics are known as software quality attributes or simply quality factors and are often considered as a part of the system's nonfunctional requirements.

IS 25000 specifies the software quality attributes and provides a framework for the evaluation of software product quality. IS 25000 describes a two-part model: a) internal and external quality, and b) quality in use. The first part of the model defines 6 quality attributes and 28 sub-attributes. The second part of the model specifies 4 quality-in-use attributes. In addition, Data quality model specified in IS 25012 has 6 internal and external data quality which are further subdivided into sub-characteristics and four data quality in use.

Idealistically, every system exhibits the maximum possible value for all its attributes. The system would be available at all times, would never fail, would supply instantaneous results that are always correct, and would be intuitively obvious to use. Since this situation is not possible in real world and different parts of the product need different combinations of quality attributes, users have to trade off the most important attribute to the project's success.

This subtopic maps the effects of using the requirements quality characteristics into the software quality characteristics (IS 25000) and data quality (IS 25012).

Requirements quality characteristics	Software quality characteristics (IS 9126, IS 25000)	Data quality characteristics (IS 25012)
singularity	Functionality (e.g., accuracy) Maintainability (e.g., analyzability)	Understandability
correctness	Functionality (e.g., accuracy)	Accuracy, Precision, Creditability
conciseness	Maintainability (e.g., analyzability)	Understandability
non-ambiguity	Functionality(e.g., accuracy)	Precision, Understandability
traceability	Maintainability (e.g., changeability)	Traceability, Accessibility
compliancy	Functionality (e.g., security)	Compliance, Portability
priority	Functionality (e.g., functionality compliance)	Compliance
verifiability	Maintainability (e.g., testability)	Accuracy, Completeness, Traceability
abstraction	Maintainability (e.g., analyzability)	Consistency
feasibility	Maintainability (e.g., testability)	Completeness
stability	Maintainability (e.g., stability)	Credibility, Currentness
sufficiency	Functionality (e.g., suitability)	Credibility
necessity	Functionality (e.g., functionality compliance)	Completeness
consistency	Maintainability (e.g., analyzability)	Consistency
uniformity	Functionality (e.g., interoperability)	Precision, Consistency
modularity	Reliability (e.g., maturity)	Confidentiality, Accessibility
scoped	Portability (e.g., installability)	Portability
uniqueness	Maintainability (e.g., changeability)	Traceability, Accessibility
completeness	Functionality (e.g., suitability)	Completeness

understandability	Usability (e.g., understandability)	Understandability
affordability	Functionality (e.g., suitability)	Availability, Portability

ISO/IEC JTC1/SC7
Software and Systems Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC7 /N4210

2009-01-23

Document Type	Comments Disposition Report
Title	Comments Disposition Report, N3888, FCD 24766 - Information Technology – Requirement Engineering Tool Requirements
Source	WG4
Project	10746-3
Status	Final
Reference	**N3888, N4012, Resolution 1041**
Action ID	FYI
Distribution	AG
No. of Pages	16
Note	

Address reply to: ISO/IEC JTC1/SC7 Secretariat
École de technologie supérieure – Department of Software and IT Engineering
1100 Notre Dame Ouest, Montréal, Québec Canada H3C 1K3
secretariat@jtc1-sc7.org

www.jtc1-sc7.org

ISO/IEC JTC1/SC7/WG4 4/N - 446

ISO/IEC JTC1/SC7 4/N-446

2007-12-8

Document Type	Comment Disposition Report
Title	Comment Disposition Report, DTR 24766, Information Technology – Guide for Requirement Engineering Tool Capabilities
Source	SC7/WG4
Project	
Status	Final
Reference	
Action ID	FYI or ACT
Due Date	
Mailing Date	2008-12-8
Distribution	AG
No. of Pages	15
Note	

Address reply to: ISO/IEC JTC1/SC7 Secretariat
École de technologie supérieure – Department of Software and IT Engineering
1100 Notre Dame Quest, Montréal, Québec Canada H3C 1K3
secretariat@jtc1-sc7.org

www.jtc1-sc7.org

				document:	CD ballot comment disposition for 24766 - Information Technology . Guideline to Requirement Engineering Tool Capabilities			
					Appr: 13, Appr Comm: 1 (Italy), Disappr:3 (AUS, Czech, JP)			
NB	**No.**	**Category**	**Clause, Sub-clause**	**Paragraph, Figure, Table**	**Comment and rationale**	**Proposed new text**	**Outcome**	**Disposition**
JP	1	GE	Title		The title is not conform to the desposition 973 at Moscow prenaly.	Fix the title as "Guide for Requirements Enginnering Tool Capabilities".	Accept	changed the title to "Guide to requirement engineering tool capabilities"
JP	2	TL	introduction	1	The meaning of "cycle" is ambiguous. "many requirements engineering and cycle activities."	"many requirements engineering and related life cycle activities."	Accept	changed as suggested
JP	3	TH	introduction	4	This standard does not define "requirements". Thus, rewite the sataetment.	"This guide describes capabilities of Requirements engineering tool for …"	OBE IT 3	
JP	4	TH	Scope	3	This standard does not define "requirements". "providing a set of minimum requirements for requirements engineering tools."	"provciding descriptions of capabilities for requirements engineering tools."	Accept	changed as suggested
JP	5	TH	Scope	4	The descripotion on complience is not contents to put on this standard.	Remove this paragraph.	Accept	changed as suggested
JP	6	TH	Term and definitions		Terms such as "non-functioanl requirements" and "user requirements" are undefined.	Add deifinitions of those terms.	Accept	the term, non-functional requirements, is replaced by the 'quality requirements', user requirements is defined as in IS 25000
JP	7	TL	4.1.2		This subclause does not explain on "Requirements analysis and completeness". It explains on "Requirements analysis and negotiation".	Change the title of this subclause to "Requirements analysis and negotiation".	Accept in principle	changed the title to "Requirements analysis". Negotiation is included as a task under analysis
JP	8	TL	4. 1.2		Activity of Analysis includes "analyse overlaps and conflicts between requirements".	Add "analyse overlaps or conflicts between requirments".	Accept	changed as suggested
JP	9	E	4.1.3		Usually Requirements are defined by statements and not always structured. The phrase "structre statements" is "structured statements".	Change "structre statemants" to "statements".	Accept	changed as suggested
JP	10	E	5.2	Overview	Explanation of "Goal-oriented scenario" is not clause 5.2. Explanation of "Goal-oriented scenario and modeling" suits to "Goal-oriented scenario".	Change "Goal-oriented scenario" to "Goal-oriented scenario and modeling".	Accept in principle	changed "Goal-oriented scenario" to "Goal-oriented scenario and high-level modeling".
JP	11	E	5.2	Overview	Explanation of "Import from other sources"is not clause 5.2. Explanation of "Import and Export to and from other sources" suits to "Import from other sources".	Change "Import from other sources" to "Import and Export to and from other sources".	Accept	changed as suggested

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
JP	12	E	5.3		Wrong numbering.	Wrong numbering. 5.3 becomes 5.2.1 5.3.x becomes 5.2.x+1, 5.4 becomes 5.3, 5.5 becomes 5.4,5.6 becomes 5.5,5.7 becomes 5.6,5.8 becomes 5.7	Accept	changed as suggested
JP	13	TH	5.2,5.3,5.4.1.1,5.4.1.2,5.4.1.7,5.6.1.1,5.6.1.6,		The term "class" is undefined and is used in many contexts. We can not understand following sentences. "flexible user definable data models and classes for importing of information (text and graphics)" "information content (inter and intra class hierarchical, document generation (templates and user definable)", "Tracing requirements/needs within a class and between classes and generating trace reports" "Create hierarchical relationships within and between classes" "Provide a hierarchical (parent-child) structuring and identification scheme for the elaboration of requirements within a class." "Provide a hierarchical (parent-child) structuring and identification scheme for the elaboration of requirements within a class." "Provide a hierarchical (parent-child) structuring and identification scheme for the elaboration of requirements within a class" "Provide user and group security for each class of requirements (e.g., access, read only, read-write)" "Training classes covering all aspects of the specific tool"	Define "class" or change "class" other terms for understandability.	Accept	deleted the term class

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
JP	14	GE	5.3,5.3.1,5.3.2,5.3.3,5.3.5,5.4.1,5.4.1.1,5.4.1.2,5.4.1.3,5.4.1.4,5.4.1.7,5.5.1,5.5.1.1,5.5.1.2,5.5.1.3,5.5.1.4,5.6.1,5.6.1.1,5.6.1.3,5.6.1.4,5.6.1.5,5.7.1.2,5.8.1.2,5.8.1.3,5.8.1.4,ANNEX A,ANNEX B		Following terms are ambiguous. 5.3 requirements/needs new/changed versions classifying/categorizing requirements 5.3.1 attributes/fields test/validation 5.3.2 user/stakeholder 5.3.3 User definable/tool provided 5.3.5 user defined/tool provided 5.4.1 Information/Data analysis 5.4.1.1 User defined/tool provided 5.4.1.2 system/software need/requirement User defined/tool provided 5.4.1.3 cost/performance user defined/tool provided 5.4.1.4 user defined/tool provided Import user defined/Tool provided 5.4.1.7 requirements/user need/requirement requirements/design elements verification/validation case element	Change "/" to "and","or","and/or".	Accept	changed as suggested
JP	15	E	5.4.1	Overview	Missing the explanation of "Information/Data analysis". And "Information/Data analysis" is ambiguous.	Change "Information/Data analysis" to "Information and/or Data analysis". And add an explanation of this title to this clause.	Accept in principle	Since information and data analysis can be a part of other types of analysis (e.g., Context Diagram, UML, DFD), it is deleted.
JP	16	E	5.4.1.1 Functional requirements analysis		"hierarchical structuring" implies "has_a" relationship between elements. "(parent-child)" in "Provide a hierarchical (parent-child) structuring" is redundant.	Remove "(parent-child)".	Accept	changed as suggested

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
JP	17	E	5.6.1 Overview		"Review/inspection" is ambiguous. Explanation of "Review/inspection" is not clause 5.4. Explanation of "Review and inspection" suits to "Review/inspection".	Change "Review/inspection" to "Review and inspection".	Accept	changed as suggested
JP	18	TH	ANNEX A	Table	Fllowing terms are not 6. Quaity characteristics of requirements, accuracy bounded simplicity logically closed orthogonally Awareness of risk	Remove those undefined terms form table.	Accept	changed, 'bounded' to 'scoped', 'logically closed' and 'accuracy' to 'abstraction', and deleted 'simplicity' (same as conciseness), 'orthogonally', and 'awareness of risk'
JP	19	TH	ANNEX A	Table	Following terms are not refered form the annex A. Abstraction Feasibility Abstraction Stability Sufficiency Necessity Uniformity Modularity Scoped Uniqueness	Add the reason why those termes are omiited.	Accept	changed as suggested
JP	20	TH	5.3.3 Decision methods		This subclause fits the clause 5.4 Requirements analysis. Beause the objectives of this activity is resolving conflicts between requirements and Requirements analysis includes this activity.	Move this subclause from clause 5.2 to clause 5.3.	Accept	changed as suggested
AUS	1		3.7		Requirement source is the origin of requirements and not who wrote the requirement. Also the distinction and definition of "inherent"and "ascribed" attributes are ambiguous. For example: Risk (of what?), sevice level (for what?)		Accept	improved the definition by removing unambiguous terms

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
AUS	2	TH	3.8		Definition of stakeholder in 3.8. is inaccurate and does not comply neither with IEEE template nor with what is generally acceptable in most well known RE books. See for example Robertson, S., Robertson, J. (1999) Mastering the Requirements Process, AddisonWesley. Alexander I.F., Stevens R. (2002) Writing Better Requirements, Addison Wesley		Accept	changed the terms as defined in IS 15288:2008
AUS	3	TH	4.1.1		Requirements Elicitation definition in 4.1.1. is inadequate to compare with what is covered in the literature	Requirements elicitation is the process of seeking, uncovering, acquiring, and elaborating requirements for computer based systems. It is generally understood that requirements are elicited rather than just captured or collected. This implies there are discovery, emergence, and development elements to the elicitation process.	Accept	changed as suggested with minor change
AUS	4	TH	4.1.2		There is no such activity in RE process called “completeness”, this should be removed. If the intent is to include analyzing requirements for completeness, then other requirements quality attributes should also be included such as consistency, clarity, conciseness, etc etc		OBE JP 7	
AUS	5	TH	4.1.2		There is also a set of modeling activities as part of RE process that is missing. These include goal modeling, task modeling, requirements modeling. Object modeling, domain modeling etc etc		Accept	this section is about an overview. Therefore the suggest comments are included in 5.3.5

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
AUS	6	TH	4.1.3		In 4.1.3 it is stated: "requirements typically are represented in the form of use cases, event-response tables or other models as well as structure statements." This statement is inaccurate and not factual. There is no evidence to suggest that event-response tables are used typically or even use cases for that matter. What does "structure statement" mean? In the same section it is stated: "System or software requirements also contain graphical models such as use cases". This is again incorrect, use cases are not graphical models, they are textual narrative descriptions. In the same section it is stated: "These system and software requirements are traced to user requirements." No distinction is made prior to this between different types of requirements		Accept	changed as following; Requirements specification deals with documenting the requirements in a consistent and reviewable way. Documentation includes the functions and capabilities that a software system must provide and the constraints that it must respect. Requirements specification is the basis for all subsequent software life cycle activities including project planning, design, and coding, as well as the foundation for system testing and user documentation.
AUS	7	TH	4.1.4 & 4.1.5		Definitions given for verification and validation of requirements do not comply with IEEE standard definitions nor with any of those given in the well known RE books.		Accept	changed by following ISO 9000 definition
AUS	8	TH	4.1.6		Definition for requirements management is mixed up with configuration management. This is a crucial definition and needs accuracy and rigour especially because most of the commercially available RE tools are in fact requirements management tools.		Accept	changed the definition following INCOSE system engineering handbook v. 3.1
AUS	9	GT	Entire document		There are no references identified in this document which is a clear indication that the authors are not aware of over two decades of work in this area. In particular, the work undertaken by INCOSE on requirements management tool evaluation available free of charge on INCOSE web site.		Accept in principle	this document has been prepared by collaborating with INCOSE requirements working group experts

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
AUS	10		5.2		"The following activities 5.2.1-5.2.9", where are these sections in the document? Furthermore, the bullet points under this section are a mix bag of activities and methods, they must be separated. E.g. "Goal oriented scenario" and "elicitation template and checklists" are not activities, they are metods.		OBE IE3 #11	
AUS	11		5.2		3rd bullet: "elicitation traceability", what is this? It is not clear what this means.		Accept	changed 'elicitation' to 'requirements'
AUS	12		5.3		references made to "classes" in this section assumes an object oriented approach	The standard should not make such assumption because there are many other paradims deployed in Requirements analysis and modelling such as "goal oriented","Aspect Oriented", and "Feature Oriented" to name but a few.	OBE JP 13	
Aus	13		5.3		Again these bullet points are a mixture of functions and data, they must be separated because they are meant to be functions that support the user in requirements capture. E.g. bullets 7,8 and 9 are data and not functions		Accept in principle	Improved those to be meant to be functions
Aus	14		5.3.3		First bullet: Assuming domain knowledg can be captured and specified in a particualr format? Second bullet: How does a tool analyse readability? Third bullet: Assums that a list of conflicting requirements could be identified.		Accept	The whole sub-section is rephrased for more clarification.
AUS	15		5.3.5		What does user defined/tool mean?		Accept	clarified the bullet
AUS	16		5.3.6		The bullet is ambiguous and does not make sense.		Accept	clarified the bullet
AUS	17		5.3.8		Third bullet: This is not part of SRS and based on all acceptable and well respected RE texts and standards this kind of info should not be recorded in the SRS.		Accept	deleted
AUS	18		5.4.1		7th bullet: Attributes - this is not an activity	remove it	Accept in principle	clarified the bullet to be an activity
AUS	19		5.4.1.1		What is word malignity check mean?		Accept	changed 'word malignity' to 'word spelling'
AUS	20		5.4.1.4		Modelling and Analysis are two distinct activities,	remove analysis from section title and text	Accept	changed as suggested

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
AUS	21		5.4.1.9		second bullet: this belongs to 5.5	remove: full text of requirements statement	Accept partially	requirement specification and documentation are different activities. The artifacts of requirement analysis should be kept and managed for tracking history of analysis activites
AUS	22		5.5.1.2		Third bullet: traceability cannot identify missing requirements	remove	Accept	clarified the bullet
AUS	23		5.5.1.3		this whole section belongs to 5.6		Accept	moved to requirements verification and validation section
AUS	24		5.5.1.4		Third bullet: this is not what "consistency" defined as. The 6th bullet is superfulous	remove consistency, remove 6th bullet	Accept	changed as suggested
AUS	25		5.7.1.5		first bullet: there is no such a thing as a requirement lifecycle, second bullet: already covered in requirements management	remove	Accept	changed as suggested
AUS	26		5.8.1		What is a Role based collaborative access?		Accept	clarified the title by removing 'role based'
AUS	27		6.2.1		This list is again a mix bag of stuff, many of them redundant and make no sense. Authors MUST check this list against the well respected, well referenced and used list of Alan Davis in his book: Davis, A., Software Requirements: Objects, Functions and States (2nd edition), Englewood Cliffs, New Jersey: Prentice-Hall, 1993 (U.S. ISBN 0-13-805763-X and International ISBN 0-13-562174-7).		Accept	refering to the suggested reference, some inconsistent or ambiguous terms were clarified.
AUS	28				ANNEX B is redundant and adds nothing new to this document		Accept	removed

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
CZ	1	GE			The standard refers for the standards of the ISO/IEC 9126 and ISO/IEC 14598 series for software quality model and software quality evaluation. These standards are behind the times and now are replaced by the standards of ISO/IEC 250xx series developed in SQuaRE project. It is necessary to wait for the finishing of corresponding standard of this series and arrange the terminology to be consistent with the new standards. It is not reasonable to downline the different terminology. For example attribute is defined as a measurable property of an entity. It is diametrically different concept as the concept in the draft: “attributes are entities that define properties of ... “. The new concept is fully addle-brained.		Accept	the term has been redefined by following IS 25000 and IS 9000
IT	1	E	Intro		Reference to 12207 and 15288 should be updated to latest release (2008)		Accept	changed as suggested
IT	2	E	Scope		Reference to 12207 and 15288 should be updated to latest release (2008)		Accept	changed as suggested
IT	3	TH	Scope	3rd parag. 1st statement	This is a guidance document it does not state requirements	It is the scope of this International Standard to supplement ISO/IEC 14102:1995 by providing guidance on desirable features of requirements engineering tools.	Accept in principle	changed as suggested except changing 'features' to 'capabilities' for consistency purpose.
IT	4	TH	Scope	4th parag. 1st sentence	It is incorrect to claim compliance to a guidance document.	Delete statement	OBE JP 5	
IT	5	E	Normative ref		Reference to 12207 and 15288 should be updated to latest release (2008)		Accept	changed as suggested
IT	6	E	Normative ref	all	Whenever a definition comes from another standard (eg. Process from ISO 9001), indicate source		Accept	changed as suggested
IT	7	TH	5.6.1.3	1st paragraph	Definition of Validation is not correct. It is not jus a more formal and rigorous way of ensuring meeting needs, but a way of ensuring needs are met under expected usage condition.	Validation is ~~a more formal and rigorous~~ way of ensuring that the final product meets the users/stakeholder needs under expected usage conditions.	Accept	changed as suggested

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
IE3	1	E	introduction	paragraph 3	"manage content" is awkward in this sentence	replace with "content management'	Accept	changed as suggested
IE3	2	E	introduction	paragraph 4	"minimum requirements for an RE tool". This is inconsistent with several statements in earlier paragraphs, where the point is made that several tools may be used to perform RE tasks and/or support RE tasks.	replace with "minimum capabilities for tool(s) associated with RE tasks and activities".	OBE IT 3	
IE3	3	E	Clause 1	paragraph 3	This standard is to "be used as the set of criteria by a potential user during an acquisition process, or by a RE tool supplier to help identify desirable tool capabilities.". Much of the content in clauses		Accept	removed the sentence
IE3	4	TL	Clause 1	para 4, sentence 1	"Compliance" is defined in a confusing and vague way. This wording seems to indicate that the *processes* defined in this standard will form the basis for compliance. However this is unworkable, in the sense that the wording in many sub-clauses of clause 5 deal largely with RE processes, which themselves are the subjects of other standards. As stated in the Introduction para 4, sentence one, this standard is about the supporting RE toolset. So, IF compliance to this standard is to be focused on acceptable characteristics of an RE toolset, then compliance should (ideally) be tightly defined based on RE toolset characteristics elaborated in Clause 5. Or, IF compliance to this standard is to be focused on a process (undertaken by a RE toolset customer) of matching his specific RE processes with a generic set of RE toolset features/capabilities, then this should be clearly stated (and Clause 5 needs extended beyond the generic RE toolset features, into the decision making process of the customer). See also comment #9 regarding sub-clauses 5.x and the implication of "comp	Determine whether the scope of this standard is to be only a set of features/capabilities of an RE toolset, or to be a process (presumably the evaluation / matching process used in the evaluation / acquisition of an RE toolset, not actual RE processes themselves). If the intended scope is the former, re-define "compliance" here, and review all sub-clauses in clause 5 to ensure that compliance is clear from the point of specific RE tool features and criteria. Also "compliance" is an incorrect word in this context because it is reserved by the directives to mean agreement with controlling documents of ISO/IEC and JTC 1. The correct word is "conformance". (This mistake was also made in 12207:1995 and has been corrected in 12207:2008.)	Accept	the scope of this standard is to be a set of tool capabilities. It is nothing to do with compliance. Thus the compliance was removed.
IE3	5	E	Clause 3.7	para 1, sentence 1	"attached to a requirement" is slightly awkward. Suggest "associated with a requirement".	Use "associated with a requirement".	Accept	changed as suggested
IE3	6	E	Clause 3.7	para 1, sentence 2	"Examples of inherent attributes" could be made more precise by adding an appropriate qualifier, one which makes it clear that the attributes being discussed are related to requirements.	Replace with "Examples of inherent attributes relating to a requirement"	Accept	changed as suggested
IE3	7	E	Clause 3.7	para 1, bullet 1	A requirements "source" in most software engineering standards can be either a person or a source document.	Replace "Author" with "Author or document where the requirement originated".	Accept	changed as suggested

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
IE3	8	E	Clause 3.7	para 1, bullet 3	As written this bullet does not describe an inherent attribute. It seems to hint at the attribute of "status" (being new/changed/deleted in this case). If so, re-word the bullet to express this as an attribute. [Note this is not a comment on the appropriateness of the term "status", or the choice of new/changed/deleted as values for this status attribute]	Re-word as an attribute, and make it clear that the choices of possible value are merely examples: "Status - e.g. new/changed/deleted".	Accept	changed as suggested
IE3	9a	E	Clause 5.1, 5.x	paragraph 1	The introductory sub-clause to Section 5 states that "This section provides a list of required capabilities", however in almost all of the following sub-clauses, the descriptions use the term "should" to express these minimum requirements of the tool / toolset.	For all the truly required toolset capabilities described in Clause 5, consider use of the terms "shall".	Accept in principle	since this standard is a TR document, using 'should' would be more appropriate.
IE3	9	GT	Clause 5.x		"Conformance" is (see comment #4 above concerning Clause 1) defined vaguely for this standard, and if compliance / conformance is to remain, the sub-clauses 5.x should be worded so that conformance is specifically defined, and easily ascertained by the reader.	Define "conformance" in Clause 1 (see comment #4 above) to be specifically about either the RE Toolset features / capabilities or the evaluation process to be used by the RE toolset customer, then adjust the wording in each of the sub clauses 5.x to make it very clear what conformance means for each RE toolset feature / capability (or each step in an evaluation process).	OBE IE3 4	
IE3	10	TL	Clause 5.1, 5.x	para 2	This paragraph / sentence refers to the RE processes and activities defined in ISO TR 19759. And presumably the sub-clauses following in clause 5 are organized following the process definitions / activities in ISO TR 19759. However, ISO TR 19759 is not listed as a normative reference in Clause 2. Further, ISO 12207 *is* specified as a normative reference in Clause 2. It is appropriate to ISO 12207 here and mention both standards as appropriate sources for RE processes and activities. Mentioning both with help to ensure harmonization and consistency in the use of RE process terms within this standard.	Insert a mention of ISO 12207 here in this section as a source of RE process / activities definition.	Accept	inserted IS 12207 and IS 15288 in the section, and included TR 19759 as a normative reference

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
IE3	11	E	Clause 5.2	paragraph 2	This paragraph contains a lengthy list of direct (and indirect) requirements elicitation activities. This lengthy list is not very useful here since: a) the "activities" do not necessarily add insight into the requirements for the RE toolset; and b) these "activities" themselves are (better) defined in RE process standards anyway.	Eliminate this list of "activities" entirely, or reduce the descriptions here by inserting a reference to 12207 or another standard listing these activities. Consider the entire list of activities and abstract (re-word) each of them into a tool requirement. A suitable reference might be the Stakeholder Requirements Definition process of ISO/IEC 12207:2008.	Accept	removed the listings of activities.
IE3	12	E	Clause 5.3		In terms of actual toolset requirement, there is much overlap between clause 5.2 and 5.3. See above comment #11 regarding clause 5.2. In this clause, similarly, eliminate or reduce textual descriptions of RE activities, and abstract down to the practical toolset requirement(s), which in this paragraph are largely concerned with document and information management.	Eliminate this list of "activities" entirely, or reduce the descriptions here by inserting a reference to 12207 or another standard listing these activities. Consider the entire list of activities and abstract (re-word) each of them into a tool requirement. A suitable reference might be the System Requirments Analysis process or the Software Requirements Analysis process of ISO/IEC 12207:2008.	Accept	removed the listings of activities.
IE3	13	E	Clause 5.3.1	para 1, bullet 3	The items enumerated in the example list (stakeholders, business process, activity, policy, constraints) are not "system elements". These seem to be various types / categories of information *associated* with requirements, i.e. the kinds of attributes one might wish to associate in a database with a particular requirement.	Re-word to indicate that these are attributes or types of additional information associated with a given requirement, and that the user may flexibly define any such attribute.	Accept	changed 'system element' to user definable attributes and additional information
IE3	14	E	Clause 5.3.3	para 1, bullet 1	RE toolset should support storing and management of … terminology (glossary)". While a glossary is an important work product, it is not limited to use in RE, and in fact may be satisfied by another (non-RE specific) tool. If a glossary is to be mentioned as a mandatory feature capability of the RE toolset, perhaps the required capability is that of *interfacing with* a glossary or central terminology repository.	Consider removing this reference to a glossary, or re-wording to indicate that the RE toolset required capability is to "support an interface to a possibly separate glossary or terminology repository".	Accept	changed as suggested, while moving this section into the requirement analysis section. See also JP 20 and Aus 14 comments
IE3	15	E	Clause 5.3.3	para 1, bullet 2	"analysis of readability" is vague from the perspective of an RE toolset capability. Also, this capability is more closely related to general documentation and word processing tools than a specific RE toolset.	Consider removing this bullet, or changing the wording to make it clear that this capability is more narrowly focused on support for simple text readability in formatting (which is essentially an editing or documentation support capability, not a RE specific capability).	Accept	removed

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
IE3	16	E	Clause 5.3.3	para 1, bullet 5	User defined decision methods, particularly the mechanical support for voting and decision making, are not specific to an RE toolset. Generic tools and methods for resolving conflicts, making decisions and voting are required during the software project lifecycle, and this capability should not necessarily be assigned to the RE toolset.	Remove this bullet as a required capability of the RE toolset.	Accept	removed
IE3	17	TL	Clause 4	general	This clause (all sub-clauses) presents a summary of requirements engineering, generally from a process perspective (sub-clause 4.1.3 is not written from a process perspective, however). Other ISO (and IEEE) standards exist which more clearly define requirements engineering processes, including their associated inputs and outputs. Since this standard is concerned with a tool (toolset) supporting these processes, there should be as little process description in this standard as possible. Reducing the description in this standard will also reduce the risk of inconsistency, overlap and duplication. A simple listing of RE processes, along with a normative reference, would reduce overlap and these associated risks.	Reduce these sub-clauses to a very short list which are basically the names of RE processes defined elsewhere. Insert wording referring the reader to the accepted RE process definitions contained in the normative reference.	Accept in principle	reduced the clause 4 and rephrased the sentence by following the definitions in other ISO standards such as ISO 12207, 15288, and ISO 19759
IE3	18	TL	Clause 4	general	Clause 4 may contain more RE process description than is warranted (see comment #13 above). However this standard discusses an RE toolset, and the RE toolset characteristics are certainly related to the RE processes (and their inputs and outputs). So a coupling still exists between this standard and other normative RE process standard(s). While minimizing the overlap in terms of actual process descriptions, a cross reference table would be of help to the reader. Such a cross reference would relate an RE process (perhaps even a specific input or output) in the RE process reference (i.e. in some external normative RE standard doc) to a generic RE toolset characteristic (contained in this standard, in a sub-clause of clause 5).	Consider a cross reference table which relates RE processes (even down to inputs and outputs), defined in an external normative reference, to particular RE toolset characteristics, i.e. sub-clauses in clause 5.	Accept in principle	Up to now, there is no IS standards soley dedicated to RE process.

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
IE3	19	GE	Clause 5.x	general	In various sub-clauses (e.g. 5.2, 5.3.5, 5.4.1.3, 5.5.1.1, 5.5.1.3, 5.5.1.4) references are made to "tool provided templates", which are to be provided by/with the RE tool, and which might be overridden by the user. It is not clear whether the provision of such a template is an RE toolset requirement merely for the convenience of the user (e.g. saving the user the effort of formatting a requirements report or document), or whether such a template represents the provision of specific instruction and knowledge with respect to requirements engineering. Is there some deeper knowledge (represented implicitly by the content in the template) to be provided by / with the RE tool, this should be specifically stated, and criteria for this knowledge provision should also be stated.	Consider whether there are in fact "knowledge requirements" associated with the provision of tool supplied templates and checklists, and if so, state which RE knowledge is to be embedded in these templates, and the criteria by which they may be judged for sufficiency / compliance.	Accept in principle	RE knowledge provision could be specific depending on product/service environment. Therefore specifying RE knowledge might not be necessary and feasible. However, tools should have capabilities to support in accumulating knowledge and experience.
IE3	20	E	Clause 5.8.1.4	para 1, bullet 1	The requirement for the RE tool to provide documentation with respect to all standards met (over and above *this* standard or directly related RE standards, is not very useful. Other standards, like SQUARE, are concerned about general quality characteristics of software products, and should be used (referred to) in order to handle non RE specific quality requirements for the tool which is the subject of this standard. See also related comment # 21.	Remove this clause or insert a more general statement referring to the quality requirements for software products described in other standards, such as SQuARE.	Accept	removed
IE3	21	E	Clauses 5.8.1.5 and 5.8.1.6	general	The requirements for support, maintenance and training associated with the RE toolset are not specific to RE, or to a RE tool in particular. These support, maintenance and training services are reasonable for customers to consider with respect to a RE toolset, however this standard should not present them as mandated requirements. There are other standards, such as SQUARE, which cover general quality characteristics of software products and packages. See also similar comment #20 above.	Remove these clauses or insert a more general statement referring to the quality requirements for software products described in other standards, such as SQuARE..	Accept	removed

NB	No.	Category	Clause, Sub-clause	Paragraph, Figure, Table	Comment and rationale	Proposed new text	Outcome	Disposition
IE3	22	E	Clause 6.3.1.6	paras 1,2,3	The title of this sub-clause is slightly misleading and confusing, as the reader might consider that "conformance" refers to the overall topic of conformance of the RE tool to this standard. Instead the subsequent paragraphs indicate that the "compliance" refers to an attribute of a requirement which expresses the "compliance level" (to some standard).	Adjust the title of this sub-clause to more clearly indicate that "compliance attribute" is another attribute associated with a requirement which might be tracked through the use of the RE toolset.	Accept	inserted the following sentence; compliance refers to an attributes of a requirement which expresses the level or degree of compliance to a standard.

ISO/IEC JTC1/SC7
Software and Systems Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC7 /N4212

2009-01-23

Document Type	Abstract
Title	Abstract, DTR 24766 - Information Technology – Requirement Engineering Tool Requirements
Source	WG4
Project	24766
Status	Final
Reference	**N3888, N4012, Resolution 1041, N4210, N4211**
Action ID	FYI
Distribution	AG
No. of Pages	1
Note	

Address reply to: ISO/IEC JTC1/SC7 Secretariat
École de technologie supérieure – Department of Software and IT Engineering
1100 Notre Dame Ouest, Montréal, Québec Canada H3C 1K3
secretariat@jtc1-sc7.org

www.jtc1-sc7.org

ISO/IEC DTR 24766 Abstract

Requirements Engineering (RE) is a major activity within the systems and software engineering life cycles. This activity must be carried out in a comprehensive manner to ensure that a complete set of user needs and requirements are captured. These user needs and requirements are transformed into a validated set of technical requirements and managed throughout the life cycle using the requirements engineering process activities. Requirements engineering tools are used to support many requirements engineering and related life cycle activities. Requirements engineering processes are identified in ISO/IEC 15288: 2008 – Systems Engineering – System life cycle processes and ISO/IEC 12207:2008 - Information Technology – Software life cycle processes.

IS 15288 and IS 12207 describe a set of requirements engineering processes, activities and tasks to be performed when acquiring or developing systems and software. However, these documents do not address the RE tool capabilities users can expect in order to support a RE process and other related life cycle activities.

Many requirements engineering processes are human activities that in the current state of the practice tools cannot perform and may never be able to perform by a tool. But wherever possible, a tool should support these human activities through the facilitation of documentation capture, content management, distribution, discussion forums, and decision support tools.

This guide describes capabilities of requirement engineering tool to benefit the groups of people that acquire, supply, develop, operate, and maintain a RE process.

The outcomes of this guide are to:

Support RE Personnel involved in the execution of one or more RE activities to:

- Obtain a better understanding of the relationship between the activities in which they are involved and RE tool capabilities
- Identify processes or activities that can be improved through better support by a RE tool
- Have an objective basis for a better comparison, evaluation and assessment of RE tools

Support People involved in the purchase of RE tools to:

- Review RE services that can contribute to RE process improvement
- Identify criteria for selecting RE tools

Support RE Tool Vendors to:

- Provide RE tools consistent with the International Standards IS 15288, IS 12207, IS 15940, and IS 14102

ISO/IEC DTR 24766 Keywords

Requirements Engineering (RE), RE tool, RE tool capabilities, RE process, Requirements elicitation, Requirements analysis, Requirements specification, Requirements validation, Requirements verification, Requirements management

G8 Explanatory Report

EXPLANATORY REPORT	DTR 24766 Information Technology – Guide for Requirement Engineering Tool Capabilities
ISO/IEC JTC 1/SC 7 **N 4211**	
Will supersede:	Secretariat: National Body (SCC)

This form should be sent to ITTF, together with the committee draft, by the secretariat of the joint technical committee or sub-committee concerned.

The accompanying document is submitted for circulation to member body vote as an DTR, following consensus of the P-members of the committee obtained on: 2008-05-05
by postal ballot initiated on: 2008-01-02
P-members in favour: 14 P-members voting against: 3 P-members abstaining: 12 P-members who did not vote: 7

Remarks:
Project: JTC 1
I hereby confirm that this draft meets the requirements of part 3 of the IEC/ISO Directives
Date: 2009-01-22
Name and signature of secretary: Witold Suryn