ISO/IEC JTC 1 N 9484

01-23-09

ISO/IEC JTC 1
Information Technology

Document Type: Text for DTR Ballot

Document Title: DTR 15026-1, Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary

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: 116

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

ISO/IEC JTC1/SC7 /N4202

2009-01-21

Document Type	DTR
Title	DTR 15026-1, Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary
Source	WG7
Project	15026-1
Status	Final
Reference	N4110, N4199, N4201, Resolution 1049
Action ID	FYI
Distribution	AG
No. of Pages	104
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 JTC 1/SC 7 N XXXX

Date: 2009-01-02

ISO/IEC PDTR 15026-1.3

ISO/IEC JTC 1/SC 7/WG 7 W07N1189

Secretariat: SCC

Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary

Élément introductif — Élément central — Partie 1: Titre de la partie

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 draft 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: (30) Committee
Document language: E

C:\Documents and Settings\Anatol W. Kark\Desktop\15026\W07N1189_ISO-IEC_TR_15026_Part1_DTR_v126.doc STD Version 2.1c2

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.

Document type: Technical Report
Document subtype:
Document stage: (30) Committee
Document language: E

C:\Documents and Settings\Anatol W. Kark\Desktop\15026\W07N1189_ISO-IEC_TR_15026_Part1_DTR_v126.doc STD Version 2.1c2

ISO/IEC TR 15026-1, which is a Technical Report of type [1/2/3], was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

This second/third/... edition cancels and replaces the first/second/... edition (ISO/IEC 15026:1998), [clause(s) / subclause(s) / table(s) / figure(s) / annex(es)] of which [has / have] been technically revised.

ISO/IEC TR 15026 consists of the following parts, under the general title *Systems and software engineering — Systems and software assurance*:

— *Part 1: Concepts and vocabulary*

— *Part [n]:*

Document type: Technical Report
Document subtype:
Document stage: (30) Committee
Document language: E

Contents

Page

Tables

List of Figures

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 15026-1, 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*.

As a Technical Report, this second edition Part 1 is a successor to first edition (ISO/IEC 15026:1998) but does not cancel it.

ISO/IEC TR 15026 consists of the following parts, under the general title *Systems and software engineering — Systems and software assurance*:

— *Part 1: Concepts and vocabulary*

— *Part 2: Assurance case*

— *Part 3: System integrity levels*

— *Part 4: Assurance in the life cycle*

{Editor: In accordance with IEEE CS’s liaison agreement with SC7, the following statement has been added.}

The IEEE Computer Society collaborated with ISO/IEC JTC 1 in the development of this international standard. *IEEE Std 1228-1994, IEEE Standard for Safety Plans,* was used as a base document in the development of this standard.

Introduction

Within software and systems assurance and closely related areas, many specialties and subspecialties share concepts but have differing vocabularies and perspectives. This Technical Report provides a unifying set of underlying concepts allowing the creation of understandable perspectives and unambiguous usage of terminology across these varying fields. Thus, it provides a basis for elaboration, discussion, and recording agreement and rationale regarding concepts and the vocabulary used uniformly across all parts of ISO/IEC 15026 as well as providing background information and discussion of rationales and issues.

This Technical Report emphasizes concepts needed for understanding the area of software and systems assurance and, in particular, those central to the preparation and use of parts 2-4 of this ISO/IEC 15026 international standard. In addition, it supports intellectual mastery of the area primarily at the level of shared concepts and issues. It emphasizes usage of concepts and terminology suitable across an appreciable range of properties, application domains, and technologies.

The appreciation of the contents of this Technical Report may undergo change as work proceeds on the other parts of this international standard. A revision of this Technical Report reflecting any such changes is expected to be later published as International Standard ISO/IEC 15026-1.

Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary

1 Scope

This Technical Report's purpose is to aid users of the revised International Standard 15026, but it is also applicable to its preparers, reviewers, and balloters. It is relevant across all parts of the ISO/IEC 15026 International Standard. This Technical Report defines terms. It establishes a basis for shared understanding of concepts and principles central to the International Standard including an extensive and organized set of concepts and their relationships.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 12207:2007, *Systems and engineering — life cycle processes*

ISO/IEC 15288:2007, *Systems and engineering — System life cycle processes*

ISO/IEC 15289:2006, Systems and engineering — Content of systems and life cycle process information products (Documentation)

3 Terms and definitions

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

NOTE These are intended to be uniform through all parts of ISO/IEC 15026.

3.1
assurance
Grounds for justified confidence that a claim has been or will be achieved.

NOTE This definition was generalized from that of ISO/IEC 15308-1:2005, Information technology—Security techniques—Evaluation criteria for IT security. The definition in that standard is "grounds for confidence that a TOE [target of evaluation] meets the SFRs [Security Functional Requirements]."

3.2
assurance case
Representation of a claim or claims, and the support for these claims.

NOTE 1 These claims can be the claims in which confidence is needed.

NOTE 2 An assurance case is reasoned, auditable artefact created to support the contention its claim or claims are satisfied. It contains the following and their relationships:

- One or more claims about properties.

- Arguments that logically link the evidence and any assumptions to the claim(s).
- A body of evidence and possibly assumptions supporting these arguments for the claim(s).

3.3
approval authority
Entity with the authority to decide that the assurance case and the extent of assurance it provides are satisfactory.

NOTE 1 The approval authority may include multiple entities – e.g. individuals or organizations. These can include different entitles with different levels of approval and/or different areas of interest.

NOTE 2 In two-party situations, approval authority often rests with the acquirer. In regulatory situations, the approval authority may be a third party such as a governmental organization or its agent. In other situations, for example, purchase of off-the-shelf products developed in a single-party manner, the independence of the approval authority can be an issue of relevance to the acquirer.

3.4
claim
A statement of something to be true including associated conditions and limitations.

NOTE 1 This definition is consistent with the use of the term in ISO/IEC FCD 15308-1:2007(E).

NOTE 2 The statement of a claim does not mean that the only possible intent or desire is to show it is true. Sometimes claims are made for the purpose of evaluating whether they are true or false or the effort is undertaken to establish what is true.

NOTE 3 In its entirety, a claim conforming to this International Standard Part 2 is an unambiguous declaration of an assertion with any associated conditionality giving explicit details including limitations on values and uncertainty. It could be about the future, present, or past.

3.5
design authority
The person or organization that is responsible for producing the design of the product.

3.6
failure
The termination of the ability of an item to perform a required function or its inability to perform within previously specified limits.

3.7
fault isolation
The ability of a subsystem to prevent a fault within the subsystem from causing consequential faults in other subsystems.

3.8
integrity assurance authority
The independent person or organization responsible for assessment of compliance with the integrity-level-related requirements.

NOTE ISO/IEC 15026:1998 defines it as, "The independent person or organization responsible for assessment of compliance with the integrity requirements."

3.9
integrity level
A denotation of limitations on values of a property and on its associated uncertainty and applicability

NOTE 1 Generally, the intension is that meeting these limitations related to the relevant items will result in maintaining system risks within limits.

NOTE 2 This definition generalizes from the definition in ISO/IEC 15026:1998. “A denotation of a range of values of a property of an item necessary to maintain system risks within tolerable limits. For items that perform mitigating functions, the property is the reliability with which the item must perform the mitigating function. For items whose failure can lead to a threat, the property is the limit on the frequency of that failure.”

3.10
organization
A person or a group of people and facilities with an arrangement of responsibilities, authorities and relationships.

NOTE 1 This definition, including its notes, was copied from ISO/IEC 15288:2007. In turn, that International Standard adapted the definition from ISO 9000:2005.

NOTE 2 A body of persons organized for some specific purpose, such as a club, union, corporation, or society, is an organization.

NOTE 3 An identified part of an organization (even as small as a single individual) or an identified group of organizations can be regarded as an organization if it has responsibilities, authorities and relationships.

3.11
process
Set of interrelated or interacting activities which transforms inputs into outputs.

NOTE 1 This definition is adopted from ISO/IEC 12207:2008 *Systems and software engineering--Software life cycle processes*, and ISO/IEC 15288:2008 *Systems and software engineering--System life cycle processes*.

NOTE 2 This definition does not preclude the existence of a null process, activity, or transformation or of null inputs or outputs.

3.12
process view
Description of how a specified purpose and set of outcomes may be achieved by employing the activities and tasks of existing processes.

NOTE This definition is adapted from D.3 of ISO/IEC 15288:2007, *Systems and engineering — System life cycle processes.*

3.13
product
The result of a process.

NOTE 1 This definition is identical to ISO/IEC 15288:2007 and ISO 9000:2005.

NOTE 2 Results could be components, systems, software, services, rules, documents, or of many other kinds.

NOTE 3 "The result" could in some cases be many related individual results. However, claims usually relate to specified versions of a product.

3.14
system
a combination of interacting elements organized to achieve one or more stated purposes

NOTE 1 Definition and Notes 2 and 3 are from ISO/IEC 15288.

NOTE 2 A system may be considered as a product or as the services it provides.

NOTE 3 In practice, the interpretation of its meaning is frequently clarified by the use of an associative noun, e.g. aircraft system. Alternatively, the word "system" may be substituted simply by a context-dependent synonym, e.g. aircraft, though this may then obscure a system principles perspective.

3.14
system element
a member of a set of elements that constitutes a system

NOTE 1 Definition and Note 2 are from ISO/IEC 15288

NOTE 2 A system element is a discrete part of a system that can be implemented to fulfil specified requirements. A system element can be hardware, software, data, humans, processes (e.g. processes for providing service to users), procedures (e.g. operator instructions), facilities, materials, and naturally occurring entities (e.g. water, organisms, minerals), or any combination.

3.15
systematic failure
A failure related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or of the manufacturing process, operational procedures, documentation or other relevant factors.

4 Symbols (and abbreviated terms)

5 Document purpose and audience

This Technical Report's primary purpose is to aid users and potential users of the other parts of ISO/IEC 15026. In each area it first briefly covers what might be needed by serious professionals – primarily engineers and technical mangers – new to the topic of assurance cases or, in one case, integrity levels. In many areas lists of aspects or examples are provided both to provide additional concreteness and as reminders or checklists. To better aid serious users, some areas are more detailed or nuanced. While essential to assurance practice, details regarding exactly how to measure, demonstrate, or analyse particular properties are not covered. These are the subjects of more specialised standards of which a number are referenced. Some material may be well known to some readers but not to others. Therefore, each reader needs to emphasize the material that is useful to him or her.

A variety of potential users of ISO/IEC 15026 exists including developers and maintainers of assurance cases and those who wish to develop, sustain, evaluate, or acquire a product that possesses specific properties of interest in such a way as to be surer of them. Users of this international standard can benefit from knowing the included terms, concepts, and principles. For example, while ISO/IEC 15026 uses terms consistent with ISO/IEC 12207 and ISO/IEC 15288 and generally consistent with the ISO/IEC 25000-series, the users of ISO/IEC 15026 need to know any differences from that to which they are accustomed. The remainder of this Technical Report not only covers many of the concepts of interest to users of ISO/IEC 15026 but helps to clarify issues.

6 Organization of report

Clause 7 of this Technical Report covers basic concepts such as stakeholders, product, assurance, and uncertainty. Clauses 8 covers some issues that users or potential users of ISO/IEC 15026 Parts 2, 3, and 4 need to be aware of initially. Clauses 9, 10, and 11 cover terms, concepts, and topics particularly relevant to users or potential users of ISO/IEC 15026 Parts 2, 3 and 4 respectively although users of one part can also benefit from some of the information in the clauses oriented to other parts. Clause 10 is potentially useful to users of ISO 15026:1998 as well as of the newer ISO/IEC 15026 Part 3. Clause 12 offers a brief conclusion.

Those who share the usual points of curiosity or initial questions about ISO/IEC 15026 could find it useful to take an early look at Annex A on page 58, the Frequently asked questions annex. Other annexes cover pitfalls with terminology Annex B, ISO/IEC 15026's relationships to several other standards (Annex C), phenomena (Annex D) as a way of helping ISO/IEC 15026 users to think about possibilities, security, (Annex E), and some related standards (Annex F). Annex E gives special attention to security because it is an area expected to be relatively new to many initial users of ISO/IEC 15026. However, ISO/IEC 15026 can be used for both positive concerns such as high performance as well as negatively oriented concerns such as security. A bibliography is included at the end.

7 Basic concepts

7.1 Introduction

This clause covers some basic terms and concepts. The topics covered are stakeholders, products, uncertainty, and assurance.

7.2 Stakeholders

7.2.1 Introduction

Across the life cycle systems or software can have a multiple stakeholders who might (or be perceived to) affect or be affected by the product including by product-related activities. They might benefit, lose, impose constraints, or otherwise have a “stake” in a product.

7.2.2 Kinds of stakeholders

A given product (e.g. system) will typically have stakeholders from several of the categories in Table 1.

Table 1 – Examples of Stakeholders

Product's larger environment	
Regulators	Standards bodies
Specific communities (such as government or the banking industry)	National (possibly multi-national) and international laws, regulations, treaties, and agreements
Enforcement personnel and organizations	Competitors
Entities about whom the product contains information (e.g. customers and suppliers)	Evaluators, regulators, certifiers, accreditors, and auditors
Attackers	The general public
Organizational	
Sources of relevant policies (e.g. safety, security, personnel, procurement, and marketing policies)	Decision makers regarding acquisition and usage (including request for proposal writers and issuers as well as makers of decisions to acquire or use)
Authorized units within an organization	
Directly related to product	
Product developers and maintainers	Integrators of the system or software into a larger product (e.g. OEMs or enterprise-wide application developers)
Those involved in product transition (e.g. trainers and installers)	Product operators and administrators
End users	Others involved throughout the product's systems life cycle (e.g. sustainers and disposers)
System into which product is incorporated	Other systems interacting with the product or using the product’s services
Suppliers of services or consumables to product	Product owners and custodians
Project management	Owners and custodians of elements in the product (e.g. data)

In addition, stakeholders can include non-users whose performance, results, or interests might be affected (e.g. entities whose software is executing on the same computer or on computers networked with it or users of the same body of consumables).

A different kind of stakeholder, attackers are important stakeholders who certainly impose constraints or have interests involved as in, “Both we and the enemy have a stake in keeping within the laws of war.” However, some in the security community and elsewhere use the term “stakeholders” in such a way as to exclude them. Attackers can be of many kinds and have a variety of motivations and capabilities. The issue of how hostile or malicious in intention or detrimental in action an entity would need to be to meet such a definition can be unclear. The existence and characteristics of potential or actual attackers can strongly influence decisions.

A given product or project effort might have more or less of these stakeholders in Table 1 involved. Their roles and relative importance can sometimes be difficult to establish, for example who (e.g. system funders, customers, beneficiaries, attackers, or benefit gainers or loss sufferers) are more important or should have more influence on what decisions – including importance to assurance-related decisions and importance as users of assurance-related artefacts.

7.2.3 Stakeholder interests and assets

Many kinds of stakeholder interests are possible – for example one says, “In the national interests” or “not in the interest of the organization” or “not in my interest.”

Assets may also be of many kinds including real estate, facilities, equipment, people, wealth, and information or data, an executing process, or anything else that is of value to stakeholders.[1] These can include any benefit, loss, or advantage – "interests" – of stakeholder – as in "national interests". Assets within the product and its immediate environment do not necessarily include everything that might be relevant. Examples of interests include the wealth and reputations of persons about whom information is kept and those assets about which the contents of the system could facilitate positive or adverse actions of any kind – e.g. shareholder value, facilities, infrastructure, spies, soldiers, and other valued objects, processes, or conditions. The relevant stakeholders whose interests are of concern usually include the product's owners and users, but may also be others such as listed in Table 1 in 7.2.2. Thus, developers and operators need to identify relevant stakeholder interests and assets and their values or relative importance.

7.3 Product

This International Standard ISO/IEC 15026 applicability includes "products" that are the results of processes. These include systems, software, services, system or software elements or components, and other products of processes. While primarily motivated by concern for products produced (at least in part) by human controlled or artificial processes, this is not a restriction on its use. Use in reducing uncertainty about some natural phenomenon that a system depends upon would not be surprising.

This definition of "product" as a result of a process is adopted from ISO/IEC 15288:2007 and ISO 9000:2005. It does not necessarily preclude the existence of a null process, activity, or transformation; or of null inputs or outputs.

7.4 Uncertainty

Uncertainty is used in ISO/IEC 15026 as an inclusive term. It covers lack of certainty whether the uncertainty can be modelled probabilistically or not. This definition allows the term "uncertainty" to be applied to anything. Different communities restrict the application of this term to limited usage, for example to predictions of future events, to physical measurements already made, or to unknowns. While this may be convenient within these communities, ISO/IEC 15026 users span many communities. Whenever the term is used and what it applies to is specific but otherwise might not be clear, what it applies to needs to be indicated.

7.5 Assurance

Generally, one needs grounds for justifiable confidence prior to depending on a product, especially a system involving complexity, novelty, or technology with a history of problems (e.g. software). The greater the dependence, the greater the need exists for strong grounds for confidence. One needs the appropriate valid arguments and evidence to establish a rational basis for justified confidence for the relevant claims made for the product's properties possibly covering such aspects as future costs, behaviour, and consequences. Throughout the life cycle, the need exists for adequate grounds justifying decisions related to ensuring the design and production of an adequate product to be able to place reliance on it.

Uuncertainty and confidence are not paralleling equivalents. There is not necessarily a straightforward mapping of uncertainty to confidence. For many decision makers, a professionally established uncertainty (e.g. "engineering" uncertainty) needs to be transformed into their own degree of confidence for use in their decision making.

NOTE Conceptually, this conversion of an amount of uncertainty into a level of justified confidence might be straightforward where numerical probability values (with low uncertainty about them) are available and decision makers desire to and can utilize them effectively, say using a decision theory approach. However, decision makers generally take broader approaches. When probabilities cannot be established accurately this mapping is even less straightforward and when they are unknowable, it is often not even a well-understood problem.

Nevertheless, decision makers need to obtain sufficient confidence, which is adequately justified. Professionals that use this international standard need to supply adequate grounds for such confidence and have its adequacy correctly judged by decision makers.

[1] The set of stakeholders whose interests are to be preserved or increased excludes adversaries and possibly other whose interests one might desire to limit, hinder, endanger, or harm.

NOTE This need can sometimes lead to including the kinds of evidence that the relevant decision makers find most convincing.

Assurance is a term whose usages vary, but they all relate to placing limitations on or reducing uncertainty (or sometimes more casually, having "low" uncertainty) in such things as measurements, observations, estimations, predictions, information, inferences, or possible effects of unknowns and ultimately in the achievement of a goal or claim. Such a reduction may provide an improved basis for justified confidence. More importantly, the effort spent in reducing uncertainty about the value assigned to a parameter—thereby making that value more assured – can often be cost-effective in that it improves the basis for decision-making even if the estimate of the parameter's value remains unchanged.

Examples of ways in which the word "assurance" is sometimes used elsewhere include:

- Actions taken to provide a basis for justified confidence – these actions may constitute how something is done, or the evaluations of something or how it is/was done.

- Arguments and evidence that reduce uncertainty or provide grounds to justify confidence.

- Degree an individual or organization has of justified confidence in something such as the justifiable confidence that a system exhibits its required properties or satisfies the needs and expectations for it.

While this International Standard uses a specific definition for the term as being grounds for justified confidence, for clarity ISO/IEC 15026 seldom uses the term "assurance" alone.

The term may relate to different scopes – from the world at large to product elements and their constituents – and to any property of products as well as their interactions or consequences. Kinds and examples of properties are covered in sub-clauses 9.2.7 and 9.2.8. In addition, it may relate to (1) would the product (e.g. system or software) as specified meet real-world needs and expectations, to (2) would or does the as built and operated product meet the specifications, or to both (1) and (2).

NOTE Trust is related to confidence. However, confidence can exist without justification, and trust can be bestowed without justified confidence. An entity that is trusted may or may not be trustworthy.

An assurance case is a means to provide grounds for confidence (assurance) and to aid related decision making. It has one or more top-level claims in which presumably confidence is needed and their supporting arguments connecting them with multiple levels of sub-claims that are in turn supported by evidence and where appropriate assumptions. Assurance cases are covered in depth in clause 9.

8 Using ISO/IEC 15026 or parts of ISO/IEC 15026

8.1 Introduction

This clause covers issues regarding use of this international interest that can affect all users or potential users. The topics covered are Initial usage concerns, Internal structure of parts 2-4 of this international standard, Relationships among parts of ISO/IEC 15026, Authorities, and Lack of ambiguity.

8.2 Initial usage concerns

The decision to use one or more parts of ISO/IEC 15026 usually involves understanding their purpose, scope, and contents particularly their requirements as well as considering their fit with related organizations, policies, processes, practices, personnel, and standards and other governing documents. The decision can be the result of risk assessments, needs for information for decision making (e.g. decisions to launch or acquire a product), customer direction, organizational practices, and industry or other regulatory requirements.

When conformance is not required, the decision regarding usage might include deciding to conform but not claim conformance, use the standard as guidance, or conform to or use as guidance only portions.

The properties and/or claims covered when using a part of ISO/IEC 15026 are entirely up to the users of the standard who are responding to their own needs and outside requirements upon them. Any property or claim may be selected regardless of its importance or related risk. However, Part 2, Assurance case, is intended to be used for high assurance situations and not low assurance ones, and the other parts are expected to also find their primary use in among higher assurance situations.

ISO/IEC 15026 or its parts can be used alone or with other standards or guidance both those with broad and narrow scopes. They can be mapped to most life cycle standards, and can utilize any set of well-done definitions for qualities or properties. Annex C begins to address these issues.

Decisions concerning their voluntary use need to analyse the feasibility of doing so including existing organizational readiness (e.g. felt need and relevant competencies), riskiness of situation, cost/benefit (including the amount of value affected by decisions it would support), the advantages of taking a more systematic approach to product-risk-related engineering and management activities and decisions, and the alternative approaches available. For example, on one hand, assurance cases are simply aids for good risk management. On the other, they can involve a significant change in thinking and can influence every product-related activity.

NOTE 1 In part, the use of assurance case has spread because of the occurrences of failures by alternative process-oriented standards to ensure safety or other important properties. However, many more or less process-oriented standards exist that are often quite useful for their specificity and the detail and methods they contain. Many of these are usable in conjunction with parts of ISO/IEC 15026. In addition, Part 4 is process oriented and can be used with the other parts.

While ISO/IEC 15026 Part 3 is generally backwards compatible with ISO/IEC 15026:1998, transitioning from it to ISO/IEC 15026 Part 3 will open up new engineering and decision options but require dealing with some differences. One difference is in perspective. Part 3 takes not only a standalone perspective but one that includes possibly relating integrity levels to an assurance case. It also concerns itself with the creation of integrity levels. Part 3 concentrates more on the product itself and using integrity levels with it than on external risk analysis. Clause 10 discusses integrity levels.

If a decision is made to use any parts ISO/IEC 15026, then understanding certain concepts and terms is essential. This Part 1 provides context, concepts, and explanations to aid users in doing this as well as aiding in the usage of the other Parts.

Several terms are used in a special way within ISO/IEC 15026 including "give attention," "consider," and "address" although the requirements associated with them vary slightly to fit the Part. The terms "adequate" and "appropriate" appear in this International Standard to allow users to accommodate the range of claims, situations, products, technologies, and stakeholder requirements that may occur. They are not intended to be unduly vague but rather to be interpreted or more precisely defined for particular uses. This more precise definition might be done for individual uses or for a set of uses of, for example for uses within a domain where this might be accomplished though the use of a domain standard.

Occasionally user confusion exists concerning "should". Its meaning needs to be clear to all. Within ISO/IEC 15026, "should" is used "to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) ["should not"] a certain possibility or course of action is deprecated but not prohibited" ([128] page 65). No documented justification is required for doing otherwise.

A final sometimes misunderstood point is that maliciousness and subversion are concerns even when no security-related product property is involved – for example, none in the assurance case's top-level claim. For example, malicious developers might have an effect on successful achievement of almost any property.

NOTE This Technical Report often refers to a single top-level claim. However, this is not a comprehensive prescription; an assurance case may have multiple top-level claims.

8.3 Internal structure of parts

For Parts 2-4, the main purpose of their structure and layout is to support their use including separately identifiable individual requirements or small sets of related requirements facilitating traceability regarding conformance. This structure sometimes makes a casual or initial reading less smooth, but these readings are less frequent than the repeated readings and references during use.

Parts 2-4 follow the structure of their subject matter – the assurance case, creation and use of integrity levels, and the life cycle. In addition to relevant artefacts and performing relevant activities, each Part covers some elements of related planning possibly in a separate clause or annex. They have limited introductory and explanatory material but are self-contained and intended to be usable by knowledgeable persons as standalone documents – although this Part 1 can help.

Thus, Parts 2-4 have a number of aspects designed to facilitate their serious use.

8.4 Relationships among parts of ISO/IEC 15026

The parts of ISO/IEC 15026 are:

1) 15026-1: Concepts and vocabulary: initially a Technical Report and then revised to be an international standard and possibly a guidance document.

2) 15026-2: Assurance case: includes requirements on the assurance case content and the life cycle of the assurance case itself as well as an informative clause on planning for the assurance case itself.

3) 15026-3: System integrity levels: relates integrity levels to the assurance case and includes related requirements for their use with and without an assurance case (revises ISO/IEC 15026 1998).

4) 15026-4: Assurance in the life cycle: addresses concurrent development and maintenance of the product and the assurance case including project planning for assurance considerations.

While each of Parts 2, 3, and 4 can be used in a standalone fashion, they form a related set. Part 1 relates to all of them especially Part 2 in clause 8.5, Part 3 in clause 10, and Part 4 in clause 11.

The assurance case is relevant to a greater or lesser extent in all parts. Part 2 concentrates on the assurance case but also has an informative annex on planning related to the development and maintenance of the assurance case. Part 3 relates integrity levels to their role in assurance cases, and Part 4 provides details on integrating the assurance case into the life cycle.

While Part 3 supports its use with a Part 2 conformant assurance case, it also supports use of integrity levels without an assurance case or with an assurance case that is not entirely conformant to Part 2. However, users of this Part 3 require Part 2, Assurance Case, as parts of it are required related to integrity levels. In addition, Part 3 places minimal requirements on any assurance cases used with integrity levels. These are a subset of the requirements in Part 2, and some are also requirements Part 3 places on all risk analyses.

Part 4 addresses concurrent development and maintenance of the product and its assurance case along with integrating the assurance case into the product-oriented life cycle. While more extensive, its requirements are consistent with the assurance case life cycle requirements in Part 2. However, it can be used with an assurance case that is not conformant to Part 2. Part 4 provides many points suggestive of what can be included into an integrity level's imposed requirements on development and maintenance or as evidence within an assurance case. Thus, Part 4 includes assurance-related concerns across the life cycle and concerns that extend beyond those directly related to the assurance case. These include project planning for assurance-related considerations.

Thus, Parts 2, 3, and 4 of this International Standard provide a separation of concerns and may be used separately or together. This Part 1 provides background, concepts, and related material which are, at least in part, applicable to all three.

8.5 Authorities

Parts of this international standard involve "approval authorities" as well as other "authorities." They can call for the identification of an approval authority (possibly comprised of multiple entities or multiple roles) that is ultimately the judge of both the following being true regarding actual use of the part or parts of this international standard:

a) Required decisions and information artefacts are of adequate quality.

b) Other requirements are met.

This approval authority can be inside or outside the organization producing the artefacts involved. However, legal and contractual requirements often identify the approval authority, for example as the acquirer of the product or as a regulatory authority.

However, this "approval authority" is not necessarily the judge of conformance to a Part of this international standard. To the extent possible claims of conformance to Parts are judged on more straightforward and harder to dispute aspects than the quality of artefacts and decisions judged in the context of product or project.

Sometimes in addition to the ultimate approval authority other authorities also exist. Part 3 includes obtaining agreements among authorities (e.g. design authority and integrity assurance authority).

Only in rare special cases are approvals by a contractual acquirer explicitly mentioned and always during the performance of the contract. In practice, contracts can call for the acquirer to be the approval authority or the approver of conformance to Parts of this international standard. This can lead, as can having an internal approval authority, to the possibility of conflicting motivations.

NOTE Whenever the source of funding is also the approval authority or judger of conformance, it can face tradeoffs – for example, between cost or investment and the quality of the information artefacts called for by this international standard.

This and the competence, diligence, and trustworthiness of the approvers of claims of conformance and the approval authority are potential issues. Therefore, parts of this international standard can call for identification of the approval authority and makers and approvers of claims of conformance and possibly for descriptions of their degree of independence. This allows decision makers including potential users of products to consider these (among other things) in deciding the degree of confidence they should have in any claim.

8.6 Lack of ambiguity

Lack of ambiguity is needed for assurance cases, integrity levels, and defining processes. The requirements for its users to be unambiguous within the documents it requires are explicit in this international standard. For example, each portion of the assurance case needs to be clear and unambiguous to its developers, reviewers, and users. However, it need not be necessarily clear and unambiguous to persons lacking the appropriate context, knowledge, and background.

NOTE Preferably, these prerequisites for understanding are made explicit for or within documents or sets of documents.

More generally, in achieving lack of ambiguity and usability, attention is needed to the two basic rules of usage:

- Write so readers can understand.

- Write so readers cannot misunderstand.

For the purposes of terminology usage guidance, one can restate these as in two basic usage rules as:

- Use terminology the audience can understand.

- Use terminology the audience cannot misunderstand.

Plus, two additional rules:

- Conform to governing definitions.

- Make the content usable to those that need to or will use it.

The need to follow these rules combined with the variety of definitions that exist in the relevant audiences in the systems and software communities and related specialties and subspecialties as well as within ISO publications means writers and editors often need solutions beyond giving official single definitions for single words. First, definitions of single words are unlikely to be shared across the audiences and communities of interest, and, second, even within a single audience segment the term may be ambiguous or used in varying ways. Using multiple-word terms or phrases can help.

Definitions need to be clear. In part, this might be accomplished through an internal glossary although this could also be done by reference to outside sources. When definitions are supplied by reference, the definitions provided elsewhere need to be available in a manner that does not unduly slow or hinder the accomplishment of related work or use. Second, even though definitions are given and followed additional usage rules or conventions consistent with these definitions may be needed to ensure clarity and lack of misunderstanding across audiences.

Unambiguous does not necessarily imply precise or deterministic properties or measures, but rather that they be evaluatable. It also does not imply lack of uncertainty in measurement.

Confusion can also be avoided by avoiding duplication, saying each thing only in one place.

Thus, for lack of ambiguity to be effective, three basic areas are important. First, terms need to be adequately defined; second authors, reviewers, and users need to have a shared understanding of the underlying concepts and context; and third, the assurance case needs to meet the rules of usage above.

9 Assurance Case

9.1 Introduction

Most commonly, the purpose of an assurance case is to provide assurance about product – possibly including service – properties to parties not closely involved in the associated technical development or service processes, often for purposes of certification or regulation, acquisition, or audit. More broadly stated the purpose of an assurance case is to inform stakeholders' decision-making whenever these properties are relevant and, preferably, to supply grounds for needed stakeholder confidence. Within their decision-making, different stakeholders seek to achieve their own goals such as developing a suitably trustworthy product or deciding whether to use (or allow the use of) a product in light of the risks.

NOTE In addition, an assurance case can be created simply to ascertain reality or even what claim is (or possibly was or will be) true.

Usually, an assurance case addresses the reasons to expect and confirm successful production of the product including concern for the possibilities and risks identified as difficulties or obstacles to developing and sustaining a product. Assurance cases include claims about a product and supporting arguments for these claims that are in turn supported by evidence or assumptions as well as including the relationships among these. It provides a reasoned, auditable argument supporting a claim – normally that a product satisfies relevant requirements.

The assurance case provides a multi-level structure of claims and sub-claims (or goals and sub-goals) and connecting arguments that are ultimately based on evidence and assumptions. Together they show the truth or achievement of the top-level claim(s) (or possibly their falsehood or non-achievement). Figure 1 — Fragment of structure shows the major kinds of elements in an assurance case. Furthermore, to convince stakeholders successfully, the possibilities and risks they perceive and their doubts must be addressed – whether developers believe them to be merited or not.

NOTE 1 As is discussed elsewhere, work on an assurance case does not always start by specifying its top-level claim.

Best first considered with the initial concept and requirements, the assurance case subsequently reflects experienced or postulated possibilities and risks; avoidance and mitigation strategies related to its claims; and an assurance argument referring to associated and supporting evidence from design and construction activities, verification activities, tests and trials, etc. This may eventually include in-service and field data.

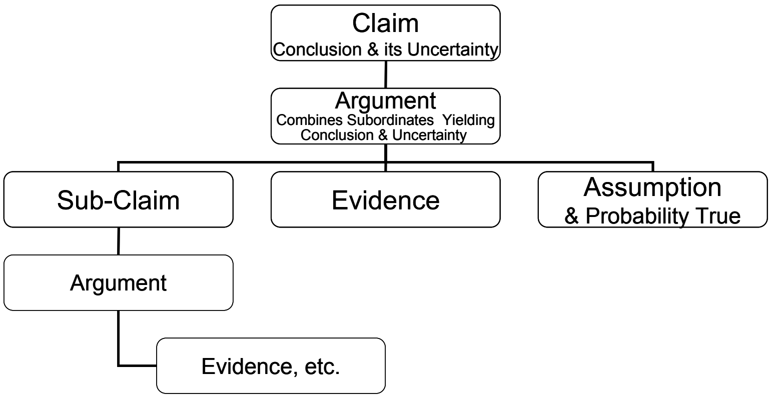

Claim
Conclusion & its Uncertainty
Argument
Combines Subordinates Yielding Conclusion & Uncertainty
Sub-Claim
Evidence
Assumption
& Probability True
Argument
Evidence, etc.

Figure 1 — Fragment of structure

Any substantive modifications in the product or the assurance case's top-level claims will necessitate changes to the assurance case. Such changes can also be needed as a result of changes in the environment. Thus, usually an assurance case contains a progressively expanding body of evidence built up during development and possibly later that responds as required to all relevant changes during development and later [[146], p. 5]. Evidence can include quality results from reviews, analyses, tests, trials, in-service and field experience, process fidelity, standards conformance results, personnel qualification records, and much more. The assurance case also records any changes to it.

NOTE An assurance case is sometimes called an “assurance argument;” in ISO/IEC 15026 “argument” is used for the arguments that connect the evidence, assumptions, and sub-claims to the claims they support (or possibly contradict).

Assurance cases can be used to address either or both verification and validation concerns. The users of Part 2 select its purpose and the product and the claims and their required properties to be covered. The assurance case’s argument must, of course, be supported by evidence and where appropriate assumptions that support each part of the assurance case argument. Such evidence comes in many forms including results from tests, reviews, mathematical proof checkers, and analyses as well as process and personnel quality.

NOTE The property in the claim is often mentioned as being associated with a product, but this is not necessarily the situation.

Although they may be separate, a combined assurance case for multiple properties may be produced. Thus, the claim's property is possibly composed from multiple properties, and these possibly include consequences.

NOTE 1 An assurance case's claim(s) properties required could perhaps include the product’s entire set of property-related requirements for a property of interest. One example might have a top-level claim composed of the combination of required limitations on consequences, functionality, and properties of the product itself (e.g. that this functionality cannot be bypassed). The qualities defined in the ISO/IEC 25000-series include qualities related to functionality and constraints. The Common Criteria v. 3.1 Revision 2 [30] is also interested in both.

NOTE 2 Many sources of information regarding assurance cases are included in the Bibliography. Standards or standards-related entries include [146], [149], [150], [154], [155], [156], [164], [165], [180], [181], [182], [196], [197], and [198].

NOTE 3 Safety cases are a use of assurance cases. Another kind of assurance case used regularly is the Reliability, Availability and Maintainability or RAM Case. RAM assurance cases are the topic of [146] mentioned above. Another approach to RAM cases is documented in the SAE JA1000 Reliability Program Standard [180]. It prescribes three objectives for the assurance of reliability; (1) demonstrate understanding of requirements, (2) develop and execute a plan to achieve those requirements, and (3) provide progressive assurance and evidence of compliance.

Detailed aspects of assurance case content include relationships, specifications, definitions, justifications, real-world consequences, conditionalities, and uncertainties. Additionally, contents may include background information and links providing traceability, and other materials. Contents need to sufficient for reviewers and other relevant stakeholders to be able to comprehend and evaluate the argument or case presented. Depending on the stakeholder and evaluation context that is anticipated, these may need to be scaled up and down accordingly always conforming to this International Standard.

In high consequence situations, an assurance case needs to exist whether written or not – otherwise an essential element needed to provide grounds for confidence (assurance) is missing – most commonly for a rational decision to use the product. Thus, the assurance case is central to rational use of products where uncertainty and consequence are serious concerns. These consequences could be either positive or negative (e.g. risk).

Considered as an artefact, an assurance case has quality issues that concern the overall case, claims, arguments, evidence, and assumptions as well as identifying and other metadata. Among these quality-related aspects are the nature of content, its form or structure (e.g. method of argumentation or modularity), semantic issues such as completeness, creation and maintenance including tool support, usability and presentation, integrity, validity, understandability, and having clearly stated conclusions with explicit degrees of uncertainty. One recent article [175] covers a substantial list of quality-related characteristics for assurance cases.

However, the success of an assurance case depends not just on its characteristics as a standalone artefact, but also on the project process as well as the more particular assurance case methods, practices, techniques, and tools. Important practices include the assurance case being considered from the earliest stage in an effort; being planned, designed, developed, and maintained concurrently with the system; and being used to influence all activities, products and services [147] and [[196], Appendix B].

An approach to assurance or assurance strategy should appear in any feasibility study and be further elaborated to accompany any operational concept document, and a description of the proposed assurance case would normally appear in a proposal document during acquisition.

The assurance case provides an audit trail of the relevant engineering considerations. It provides a justification for why certain activities have been undertaken and how they were judged successful. As a living, top-level control document, its status is continually tracked and typically summarized in Assurance Case Reports at predefined intervals or milestones. The assurance case usually remains with the product throughout its life through disposal.

While certification and regulatory authorities do not always consider everything relevant, every aspect having potential significant consequences for meeting top-level claim or for the confidence of key stakeholders has a potential place in a full assurance case along with its related evidence. It should not only give coherent confidence to developers, sustainers, and acquirers, but also be directly usable by others including certifiers and accreditors.

What are vaguely called “assurance activities” overlap (possibly entirely) with other project activities including those directed towards evaluations of both the product and the processes used to develop and sustain it. More straightforwardly identifiable, are the activities directly creating, maintaining, and evaluating the assurance case, which need to be planned and performed. Activities involved might include:

- Create top-level assurance claim from requirements.
- Establish level of uncertainty needed for information to be used in decision making.
- Establish structure of argument with sub-claims including their relationships.
- Create portions of assurance argument tailored for the desired limitation on uncertainty.
- Compile portions of argument and supporting evidence.
- Verify.
- Validate.
- Use as input to certification.

NOTE Industry and agency standards and guides explicitly about assurance cases include [146], [149], [150], [154], [155], [156], [164], [165], [181], [182], [196], [197], and [198].

This clause's major sub-clauses cover Claims, Arguments, Evidence including assumptions, Management and life cycle of assurance case, and Decision making using the assurance case.

9.2 Claims

9.2.1 Introduction

Selecting the top-level claim and the properties it involves are not restricted by this international standard – although their statement is. Top-level claims are often a portion of the total requirements and specification but may be something internal to product, related to its dependences, or not directly related to the primary product. This sub-clause includes coverage of motivations for claims, their form and scope, and example properties they might involve.

9.2.2 Motivations for a claim

9.2.2.1 General

Producers and other stakeholders may prioritize and perform tradeoff studies involving properties such as between efficiency and reliability. Additionally, achieving a quality such as safety might affect speed or other characteristics making them less desirable. Tradeoffs occur within many activities such as specifying the product’s external behaviour. This last area is a “product design” activity and, as such, can be fraught with tradeoffs, including ones among properties or qualities. A number of techniques have been created for addressing these trades, such as those in [25][69][130][168] and [42]. The specifying of a top-level claim is sometimes the result of analyses including tradeoff studies.

The production of assurance cases can be top-down or bottom-up. In top-down production, the limitations on required property values and associated uncertainties within a top-level claim are first established deriving from analyses and the purposes, uses, expectations, and intentions regarding the assurance case. In practice, stakeholders may tolerate a range of results. The subordinate claims (sub-claims) derive from what is required for the claims and arguments above them. What is required must be met by the established "actual" values and uncertainties derived from below the claim – from the supporting arguments, evidence, sub-claims, and assumptions. “Support” can include contrary as well as supportive aspects.

9.2.2.2 Kinds of questions might desire to answer

While the nominal question usually mentioned that the assurance case is helping to answer is: (was, is, or will) the claim shown within the required uncertainty limitations. In brief form this question might be stated as:

a) Will it be good enough for what we require and are we sure enough about that?

However, several other kinds of questions might readily be asked:

b) What is the chance that the claim's property will meet its limitations?

c) How lenient would limitations on property's value need to be to allow us to be sure enough its values will fall within limitations?

d) What can be shown about the claim's property from the evidence?

In addition, several other more open kinds of questions might be asked, for example:

e) What if anything can we be sure (enough) about regarding this situation or product?

f) What if anything can we find out about this situation or product?

g) Finally, for each of b) through f), one can also ask, for how long under what conditions?

These questions represent different motivations or starting places for using an assurance case.

Assurance cases can be used to address either or both verification and validation concerns. These include answering:

a) If assurance case’s top-level claim is met, will this would result in meeting real-world intention(s), need(s), and expectation(s)?

b) Will or does product as designed and built and possibly as transitioned, operated, etc. meet the top-level claim.

These two aspects need to be dealt with by any approach to the life cycle of a product, where risk, consequences, or uncertainty is an issue. These points may be dealt with separately or together, and assurance cases used for one only, together in single assurance case, or both separately but with as required for meeting requirements for such a combination Part 2.

The parts of the first point and the second point can be addressed by assurance cases – either separately or in combinations - to aid not only in gauging feasibility, suitability, and desirability of development, release, use, etc. but also for corrective action, learning, and improvement. If an assurance cases for one point is combined with one for the other, then they need suitably overlapping claims

To establish feasibility, suitability, and desirability for production, transition, use, etc. knowledge concerning the second point is needed. Risk management combines degrees and uncertainties regarding achievement of points 1) and 2) to establish the comprehensive, net or residual potential consequences or risk. Also relevant here are the benefits to decision making from knowledge in the assurance case and its conclusions.

9.2.2.3 Categories of requirements

While it need not be the case, top-level claims and their properties often derive from product requirements. ISO/IEC 15288 in its sub-clause 6.4.1 Stakeholder Requirements Definition Process lists a number of areas from which requirements can be derived and in 6.4.2.2 Outcomes states one outcome as specification of the required characteristics, attributes, and functional and performance requirements for a product solution. Within Stakeholder Requirements Definition Process in sub-clause 6.4.1.3, it mentions:

- Consequences of existing agreements, management decisions and technical decisions.
- Activities and set of activity sequences.
- Relevant characteristics of the end users of the system.
- The physical environment, social and organizational influences.
- Interaction between users and the system including the areas of:
- Physical, mental, and learned capabilities.
- Work place, environment and facilities, including other equipment in the context of use.
- Normal, unusual, and emergency conditions.
- Operator and user recruitment, training and culture.
- Critical qualities such as health, safety, security, and environmental damage.

ISO/IEC 25030 provides a categorization for requirements that, while expanded only for software, has relevance to other aspects of systems as well – as well as to the system as a whole. These include functionality, quality (both internal and external), and development requirements, and quality in use.

Examples of internal measures are given in ISO/IEC 9126-3 (to be replaced by ISO/IEC 25022). Examples of external measures are given in ISO/IEC 9126-2 (to be replaced by ISO/IEC 25023). Examples of quality in use measures are given in ISO/IEC 25010.

The ISO/IEC 25000 series defines a large number of qualities in use. Many come from life cycle processes that usually follow development. Examples of some such broad sources of requirements (not all from ISO/IEC 25010) include requirements related to manufacturability, marketability, training, maintainability and test equipment, usability, interoperability, use in extreme environments, legal compliance, cost of operation, and mission accomplishment.

The need fulfilled by a top-level claim and its assurance case can be small compared to the concerns related to the total system or product. Will the ground at the site support the planned structure? Will a sub-function on a new airplane be at least as safe as the same sub-function on similar prior airplanes? Is a tool adequately trustworthy? Thus, assurance cases may be used for either large and small requirements or needs.

9.2.3 Form of a claim

A claim takes the form of a true-false statement concerning a property. This property may be some combination of other properties. The term “property” is used quite generally – a property is a descriptive aspect that may have a claim concerning it, at least in principle, evaluated as true or false.

This usage of the term “property” derives from, is consistent with, and subsumes the broad use of the term “property” in ISO/IEC CD 25010 where it is used spanning properties including properties that are inherent or not, internal, external, and in use or context. In principle, one could apply this International Standard to claims regarding any property of any importance.

A claim is a true-false statement which concerns the limitations on the values of an unambiguously defined property – called the claim's property – and limitations on the uncertainty of the property’s values falling within its associated limitations during the claim's duration of applicability under stated conditions with possibly uncertainties associated with the last two as well. Thus, a claim potentially contains:

- True-false statement.
- Claim's property.

- Claim's property value limitations.

- Uncertainty limitations on property value meeting its limitations.

- Limitations on duration of claim's applicability.

- Duration-related uncertainty.

- Limitations on conditions associated with claim.

- Condition-related uncertainty.

NOTE The term "limitations" is used to fit the many situations that can exist. Values can be a single value or multiple single values, a range of values, multiple ranges of values, and be multi-dimensional. Finally, the boundaries are sometimes not sharp but rather involve probability distributions, or are incremental or have other fuzzy aspects.

Of course, each of these may have details within it. In particular, the property might include consequences or possibly their worth – how valuable they are or would be.

In different circumstances or for different purposes, for each of these elements within the claim one can state one or more of the following about an element. What is:

- Required.

- Planned to be established.

- Actually shown or established.

- Contradicted might also be a possibility.

Others kinds of meanings could be possible. For example, that the value has had the limitations on its values fixed (held constant). This might be done for the sake of exploring or analyzing what values of the other elements of the claim (and possibly other properties) would be consistent with the fixed limitations on its value.

NOTE At a given moment of the last six kinds of elements, some subset of them is "definite" within the context of interest – e.g. what is required has been specified, what is planned has been determined, or what has actually been shown is known. This means that $2^6 = 64$ possible combinations could exist.

The term "conclusion" describes the actually shown or established conclusion regarding whether actual property values (did, do, or will) make the true-false statement true or false. The conclusion has its associated uncertainty - that is the uncertainty associated with the claim's property's actual value(s) - meeting it limitations.

The quality of a claim depends on it being fully specified, so the true-false statement can need to be further defined by references to either incorporated or outside material such as definitions of terms or descriptions of context. Aspects possibly needing supplementary definitions include limitations on range of property values (e.g. required agree of achievement or tolerances), its duration of applicability, conditions it requires or on which it depends, unambiguous definitions of the terms (e.g. properties being combined and relations used in the combination or measures) used, and limitations on uncertainty. While many terms may have well-known meanings and even abbreviations, e.g. kilometres per hour, that do not need explicit definition many do, and all terms such as units of measure need to be explicitly stated.

Often the conditions under which the claim is said to be true and its duration of applicability (which may be other than one continuous period) are often treated as unvarying constants. However, they could be treated as variables and as having uncertainties. This can be particularly true when the purpose of the assurance case is to simply establish what is true. For example, uncertainty may exist about the durability of the product or how long it will continue to possess a quality. Figure 2 gives a simplified view of the claims elements and their relationships.

In addition to the required limitations on uncertainty, a claim or one of its aspects can have associated with it as mentioned above several categories of uncertainty, namely the uncertainty that is required to be achieved, the uncertainty that is planned to be achieved by the assurance case, and the uncertainty that it has already achieved. In the end, that last, the uncertainty about the claim achieved by its supporting argument (and sub-claims, evidence, and assumptions), needs to meet any required limitations for its uncertainty. Finally, uncertainties can exist about uncertainties.

Among the forms in which claims may be stated is to do so in terms of placing limitations on events or on the establishment and preservation of conditions. For example, claims might take the forms of:

- For events:
  - Having desired behaviours and events.
  - Limitations on undesirable behaviours and events.
- For conditions:
  - Establishing (and possibly preserving or re-establishing) the preconditions for desired events.
  - Establishing and preserving the conditions that preclude (or limit) undesirable events.

In the last point, to preclude an event the relevant condition must imply the negation of the precondition for such an event. Similarly, one speaks of states and state transitions. A state of a system or product, a possibly relevant condition, can involve many aspects as [[14], p. 13] states, "The total state of a given system is the set of the following states: computation, communication, stored information, interconnection, and physical condition." If a system contains humans, then the relevant portions of their states are also part of the system state.

9.2.4 Scope of concern

In different situations or activities, concerns for properties can vary in time and in extent. In time concerns can exist before and across the life cycle and supply chain, and beyond to ultimate consequences and residual obligations. A product-related claim can have a scope of interest that is on one side of or extends across the product-environment boundary (e.g. system boundary). In extent or nature, concerns vary across several levels of lessening scale from the real world to the individual product element or service, behaviour, or property as well as their relationships, makeup, contents, and governance.

9.2.4.1 Extent

Recognizing these varying scopes of concern ISO/IEC 25010 has a quality model – albeit limited to software – that defines three different kinds of quality:

- Quality in use: the extent to which a product used by specific users meets their needs to achieve specific goals with effectiveness, productivity, safety and satisfaction in specific contexts of use.
- External software quality: capability of a software product to enable the behaviour of a system to satisfy stated and implied needs when the system is used under specified conditions.
- Internal software quality: Capability of a set of static attributes of a software product to satisfy stated and implied needs when the software product is used under specified conditions.

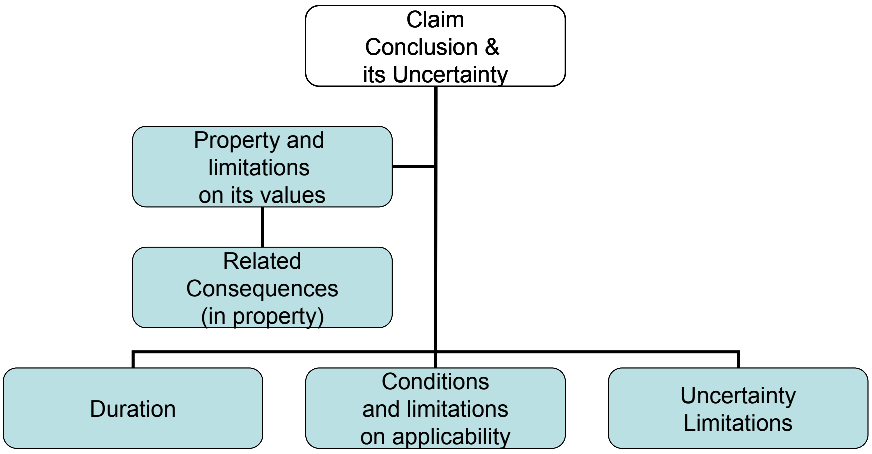

Claim Conclusion & its Uncertainty
Property and limitations on its values
Related Consequences (in property)
Duration
Conditions and limitations on applicability
Uncertainty Limitations

Figure 2 — Claim

A somewhat similar sense of expansiveness exists when conceptualizing where the behaviour, events, or conditions of interest occur. These might relate to concerns in different scopes. These include:

- Real world – e.g. concerns regarding funds, lives, real property, natural environment, and other interests of stakeholders – including allies, adversaries, and neutrals or bystanders.

- System-environment interface – .e.g. user interface, interface with sensor or effector, service offered by system, service depended upon by system, intake of consumable, output of system product (possibly including by-products), physical support, interaction with test environment/equipment.

- Internal to system:

  - System elements, resources, or assets.

  - Non-computing elements, resources, or assets – possibly containing computing or information sub-elements, resources, or assets.

  - Computing or information elements, resources, or assets – e.g. database, stored software, computing hardware.

  - System behaviour – e.g. internal operations or operations viewed from an internal perspective.

  - Software behaviour.

  - Enabling functionality – e.g. logging, automatic recovery.

Generally, from outside-in (top to bottom of list), the scopes form an ordered layering of extent of concern. These can roughly correspond to levels in an assurance case argument that includes consideration of consequences and bases its argument or behaviour, events, and conditions in the environment as affected by behaviour at the product-environment boundary that is in turn the result of behaviour internal to the product. Ultimately, concern is usually driven by affects in the first of these scopes – that is by real-world effects: benefits, costs, and consequences. However, everything down to the detailed internals of the product may be relevant in building an argument.

9.2.4.2 Duration of applicability

The duration a claim applies (that is, it claims to apply) might be stated in calendar time, as a time interval or intervals, as life cycle processes, as being during certain activities, under certain conditions, or some combinations of ways. It might result from what is required or by what is achievable. While the duration of applicability is usually a constant, it may have required, planned, and supported values and should at least eventually be tacitly consistent with the last. It may also have associated uncertainty, but this is not a requirement.

9.2.5 Consequence

In practice claims (and requirements) can extend beyond the boundaries of the product or its behaviours. In particular, it can place limitations on consequences related to a product's behaviour and/or product-related events, activities, and/or conditions – especially on the values of consequences. One may refer to:

- **Consequence:** Any effect (change or non-change), usually associated with an event or condition or with the product and usually allowed, facilitated, caused, prevented, changed, or contributed to by the event, condition, or product. It could yield a benefit, loss, or neither.

- **Desirable (or positive) consequence:** A desirable consequence. Commonly associated with a gain or avoiding an adverse consequence.

- **Adverse consequence:** An undesirable consequence. Commonly associated with a loss.

The value of any consequence is also a consequence. A consequence has value or is desirable or undesirable from a perspective or viewpoint or in terms of stakeholder interests. A consequence may occur anywhere in the product's life cycle or beyond. Certain effects within a product may be treated as consequences such as wear from use and damage from mishaps. For example, "The publishing of the concept for the product induced enquires to start concerning possibilities for both investment and purchase," or, "The product was retired and disposed of long ago but liability remains and new liability claims continue to occur."

9.2.6 Claim violation

9.2.6.1 Violation-related terms

The following three terms used in this international standard are widely used in standards and elsewhere:

- Fault: A defect in a representation of a product or a product that if followed and/or executed/activated could potentially result in an error. It is incorrect and usually thought of in terms of a static representation or a static instance of the product. Faults can occur in specifications when they are not correct. (See sub-clause 9.2.8.1.)

- Error: An erroneous state of the product.

- Failure: An externally visible deviation from the product's specification.

Under the same conditions, exercising a fault might or might not invariably result in an error. Likewise, an error might or might not result in a failure. At a certain time, a fault, error, or failure can be known (recognized) or unknown.

For the next four terms used, usage is less uniform for terms describing the following four concepts:

- External mistake (e.g. human error): External entity's or entities' non-malicious action or inaction, or non-malicious input to or interaction with the product that has the potential to result in a fault or/and error (and thereby possibly in failure) or an adverse consequence either not intended or not intended to be adverse.

- Attack: A malicious action or interaction with the product or its environment that has the potential to result in a fault or and error (and thereby possibly in a failure).

- Adverse consequence**:** An undesirable consequence. Commonly associated with a loss.

- Violation: A behaviour, act, or event deviating from a property particularly a product's desired property or from a claim particularly used in relation to a claim or property of interest. Examples might include violation of a performance standard, a speed limit, limitations on tolerances, confidentiality, laws, or a claim of suitability.

Human errors (including organizational ones) can be intended or unintended, and planned or unplanned, and in agreement or disagreement with a plan reflecting a mistaken plan, cognitive lapse in enacting a plan, or non-cognitive slip.

9.2.6.2 Violation-related concepts

Threatening entities – also referred to as sources of danger, threat agents, and attackers – can possess capabilities, resources, motivations, and intentions that enable them to initiate and carryout non-malicious (e.g. mistaken) or malicious efforts to violate a claim. To perform their efforts or attempts, violators use their capabilities to use or take advantage of product- and/or environment-provided opportunities called vulnerabilities – "weaknesses…that could be exploited or triggered by a threat source" [160].[2] Non-malicious and malicious entities use specific methods (e.g. agents and kinds of attacks) that often fall within recognizable patterns referred to as patterns of abuse, failure patterns, accident patterns, and attack patterns.

Products or their environment often employ countermeasures to limit or reduce the opportunities for and ease of violations and limit or reduce adverse consequences, for example extent and intensity of damage that would result from a violation – say from a successful or partially successful attack. Generally, attackers make

[2] For many purposes, the meaningfulness and need to separate vulnerabilities from other weaknesses can be low or non-existent. In addition, a question always exists about the current and future contexts that are relevant for "could be exploited or triggered".

gains only after further effort and product stakeholders have losses after making efforts to limit them. Their respective gains and losses often differ.

9.2.7 Properties

9.2.7.1 Introduction

The term “property” is used quite generally. In ISO/IEC 15026, a property is a descriptive aspect that may have a claim concerning it, at least in principle, evaluated as true or false. Generally, this implies a property needs to potentially:

- Be true or false, or
- Vary in some other way.

To users of this International Standard, the practicality of evaluation is a central issue. This reflects the issues of how to establish estimates of their values and how sure or uncertain one is of them including predicting, measuring including testing, and analysing. More particularly, this is a concern of assurance cases. They address the question of the uncertainty a property does or does not (or did or did not, or will or will not) meet its related claim (or claims).

Thus, what a property might be includes a condition, characteristic, attribute, quality, trait, measurement, and consequence. Properties are means of description including specification or definition. A property might be variously, for example, invariant; or dependent on time, situation, or history. Although they may be separate, a combined assurance case for multiple properties could be produced.

NOTE 1 This usage derives from, is consistent with, and possibly expands on the broad use of the term “property” in ISO/IEC CD 25010 where it is used spanning properties including properties that are inherent or not, internal, external, and in use or context.

NOTE 2 In the use of this International Standard, a property is usually expected to be relevant directly or indirectly to a product or products.

NOTE 3 In principle, one could apply this International Standard to claims regarding any property of any importance.

They may also be of interest for what they were in the past, what they are presently, or what they will be in the future. Generally, the last is the most important in this International Standard. As this involves predicting the future, it is often also the most difficult and uncertain. Therefore unsurprisingly, products’ future behaviour and consequences often become principal issues in their assurance.

Many of the properties of interest are qualities of the product (e.g. system or software). Several standards and reports provide lists and definitions of qualities including ISO/IEC 9126, ISO/IEC 25010 (and the related series), ISO/IEC 2382-14, ISO 9241, ISO/TR 18529, and ISO/TS 25238. Several standards or reports mention consequences associated with products within a specific domain. Examples include ISO 14620, ISO 19706, and ISO/TS 25238. Risk management standards also address consequences, for example ISO/IEC 16085.

Several general properties have been mentioned and more are listed below. However, a specific product requires specific specified properties within these general properties. Examples of concern for properties include the integrity of a barrier, the maintainability of a piece of equipment, and the availability of a less than three minute response by the fire department, and the early confidentiality of new weapons (e.g. the US F-117 stealth fighter). For information or data, confidentiality may only be relevant to a portion of the product’s data and integrity concerns relevant to certain operations involving certain data. In addition, the limitations on uncertainty desired regarding each engineering application of a property may vary with the level or seriousness and that in turn usually reflects the possible consequences in the real world.

Properties may include (but are not limited to) dependability-related qualities such as reliability, availability, integrity, maintainability, correctness, accuracy, safety, confidentiality, accountability, or usability (e.g. human error proneness); time- and resource-related ones such as processing speed, schedulability, throughput, and storage capacity; and human and organizational ones such as those related to human factors, as well as more global ones such as profit or mission achievement.

9.2.7.2 Specifying properties of behaviours

Often the property to be specified can be understood as a partial specification of behaviour. For example, a property could be that a certain erroneous state cannot be reached, or that a certain sequence of transitions must (or cannot) occur. At the scope system (or software) performed operations, behaviour-related properties might be formally specified as a combination of:

- Safety property: restriction on allowed system states.
- Liveness property: system states that must reached; required progress or accomplishment.
- Constraints on flows or interactions; requirements for separation .

These kinds of properties can be stated as conditions or constraints that must be true of the system.[3] In practice, these are non-trivial and modularized and involve time and starting state(s) as well as state transitions related to interaction with the system's (or software's) environment.

If the product states are adequately known or modelled, this approach can also be taken at the product-environment interface (e.g. system-environment interface or software and its environment). One may also wish to model the environment if affects on and within it or its changes in its state are important to overall consequences. This is one way requirements related to the environment's condition (e.g. a certain condition in the environment would be catastrophic) and combined product-environment behaviour can be addressed. This is not an unusual situation as the situation of interest is often larger than the product.

Many kinds of flows such as of gases, fluids, traffic, or information are of possible interest as well as constraints on them such as non-interference and separations to be maintained. In addition, flow constraints are often convenient or necessary to specify aspects of information security [143] including access control mechanisms and policies, and restrictions on information overtly or covertly communicated,

9.2.7.3 Content of specifications

Specifications may be representations of static and/or dynamic aspects of the product. One may speak of an external specification, a specification related to the product-environment boundary, or a top-level specification that may contain some internal design. Specifications often include descriptions of capability, functionality, behaviour, structure, service, and responsibility including time- and resource-related aspects as well as limitations on frequency or seriousness of deviations by the product and related uncertainties. ISO/IEC 15288 and ISO/IEC 12207 as well as the IEEE standards on requirements divide these concerns into “functional” and “non-functional” ones.

Specifications may be prescriptions and/or constraints (e.g. for and on product behaviours) as well as including measures of merit and directions regarding tradeoffs. Generally, specifications place some limitations on when they apply such as on the environment and its conditions (e.g. temperature) and possibly the conditions of the product (e.g. age or amount of wear).

9.2.8 Examples of properties

9.2.8.1 Correctness

Two major kinds of correctness are relevant:

- Correctness of the specification (or portion thereof) in terms of meeting needs and expectations and for practical purposes such as being feasible.
- Correctness of artefacts and the product in terms of agreement with the specifications – as well as by extension agreement during usage or involvement of the product in activities such as transition, operation, loss or theft, and the rest of the life cycle.

[3] If specified formally, this can allow static analysis of conformity of designs and code potentially adding creditable assurance evidence.

For the latter point the product might be thought of as having two variants of correctness:

- A product is correct at its external boundary if it would always meet its external behaviour specification under the required conditions. That is, it has no failures.

- A product is correct throughout if it contains no faults and therefore is never, under the required conditions, in an error state. In ISO/IEC 25010 terms, this is an internal quality.

Being internally correct (correct throughout) implies being correct at the external boundary, an external quality in ISO/IEC 25010 terms. A system can be externally correct, however, without being internally correct if it can tolerate or recover from internal error states never displaying incorrect externally visible behaviour.

9.2.8.2 Dependability properties

Dependability is a qualitative "umbrella" term [[14], p. 13]. ISO/IEC 25010 notes that "dependability characteristics include availability and its inherent or external influencing factors, such as: reliability, fault tolerance, recoverability, integrity, security, maintainability, durability, and maintenance support." Several standards address dependability (e.g. [69], [70], and [74]), and many more address the qualities within it. IEC 50 (191) offers related definitions [68].

Thus, it includes reliability, safety, maintainability, integrity, availability, plus related survivability; and when addressing security, confidentiality, accountability (knowing who or what did something), non-repudiation (their not being able to deny it), authenticity, security compliance, and immunity (the degree to which the product is resistant to attack), In addition, interfacing with humans or usability, particularly error-prone and inconvenient interfaces, can also have a significant effect on dependability.

Assets may be categorized by attributes related to the dependability property of interest. Examples include confidentiality (e.g., Top Secret, Secret, Confidential, or Unclassified); by their degree of integrity (e.g., accurate and up to date versus old and with unknown accuracy (possibly including corrupted)); or by criticality of availability or acceptable level of unavailability (e.g. outage length 0-1 minute, 1-5 minute, 5-10 minutes, 10-30 minutes, 30 minutes-2 hours, greater than two hours). Ultimately, such categorizations derive from and are surrogates for the values of the stakeholders' real-world benefits and losses (and sometimes uncertainties) potentially associated with the property's (e.g. integrity's) preservation and violation.

9.2.8.3 Time- and resource-related properties

Time- and resource-related properties include such properties as meeting deadlines, efficiency, and storage capacity. These can be not only important alone but also in combination with dependability-related and other aspects.

Table 2 — Some time- and resource-related properties

Timing	Throughput
• On time	• Bandwidth
• Meeting Deadline	Speed
• Schedulability	• Rates of Learning and Use
• Delay or Latency	Size
• Response Time	• Storage Capacity
• Sequence	Productivity
• Serializability	Efficiency
• Order Independent	Cost

Many relationships and potential tradeoffs can exist among dependability properties and speed, efficiency, or other time- or capacity-related properties. An example of a property that is relevant to both areas is computational difficulty. Computational difficulty is an issue when one tries to compute something and when one wants to prevent someone else from computing something. The first is of interest in areas like achieving real time performance and the latter in decryption.

Another example spanning the two areas is availability and timing-out on whatever the deadline is in the particular measurement of availability. In the end, the issue in availability is not whether the system will eventually respond, but will it respond within a specified (e.g. as useful or acceptable) period.

Table 2 lists several properties related to time and resources. These include rates such as throughput or processor clocking, size, and economic ones. An entry such as “storage capacity” could relate to anything stored from gasoline to binary bits. These kinds of properties are often measures of some aspect of performance and often are partial measures of merit for a product.

9.2.8.4 Human and organization-related properties

An almost unlimited number of properties can be associated with humans and organizations. Among common ones relevant to systems are usability, safety, size and reach, strength, mission accomplishment, and benefit and loss.

Normally, a human interface requires some concern in assurance case for human factors because almost every property is affected by it particularly an external quality or a quality in use. Human factors are addresses in several standards, for example [83], [99], and [107].

9.2.8.5 Compliance or conformance

Compliance or conformance to laws, regulations, standards, policies, and other governing documents is an issue, indeed a necessity – for almost every system. These vary by industry, location, and property of interest. Claims concerning them are often a central concern.

9.2.8.6 Other kinds of properties

Properties are everywhere. They include anything objectively measurable and many things that are not. Examples include shape, colour, attractiveness, opportunities made available, reusability, buoyancy, hardness, and strength. ISO/IEC 9126-1, ISO/IEC 26702, and ISO/IEC 25030 list many other product qualities that could be the subject of assurance.

9.3 Arguments

9.3.1 Introduction

Arguments are the glue that holds the assurance case together by relating its immediate underlying support – sub-claims, evidence, or assumptions – to the claim (or claims) it supports. It yields the combined effect of the evidence, sub-claims, and assumptions that it utilizes into a conclusion – usually that the claim it supports is true – and an associated uncertainty for its conclusion.

The uncertainty regarding the conclusion derives from the uncertainties in the argument's immediate underlying support plus the strength or rigor of the argument and its own affect (plus or minus) on uncertainty. For example, several pieces of evidence that individually would leave much uncertainty about the claim might be combined by an argument into support yielding a claim with low uncertainty.

Normally, this uncertainty that the argument has derived needs to be within the limitations for uncertainty that were allocated or budgeted to the claim – meeting its requirement. This required limitation on uncertainty (it could be in terms of limitation on risk) derives from the uncertainty limitations of claims yet higher in the overall argument structure and ultimately from the limitations associated with the top-level claim. Plus the claim in question limitations may be affected if “local” consequences associated with a claim that is sub-claim in n assurance case. For example, this might be the situation if the claim relates to a product element whose misbehaviour could have separable consequences of its own.

Figure 3 — Argument Context provides the context in which an argument shows how its super-ordinate claim (and sometime claims) is implied by the sub-claims, evidence, and assumptions that lie immediately below it and provide its support.

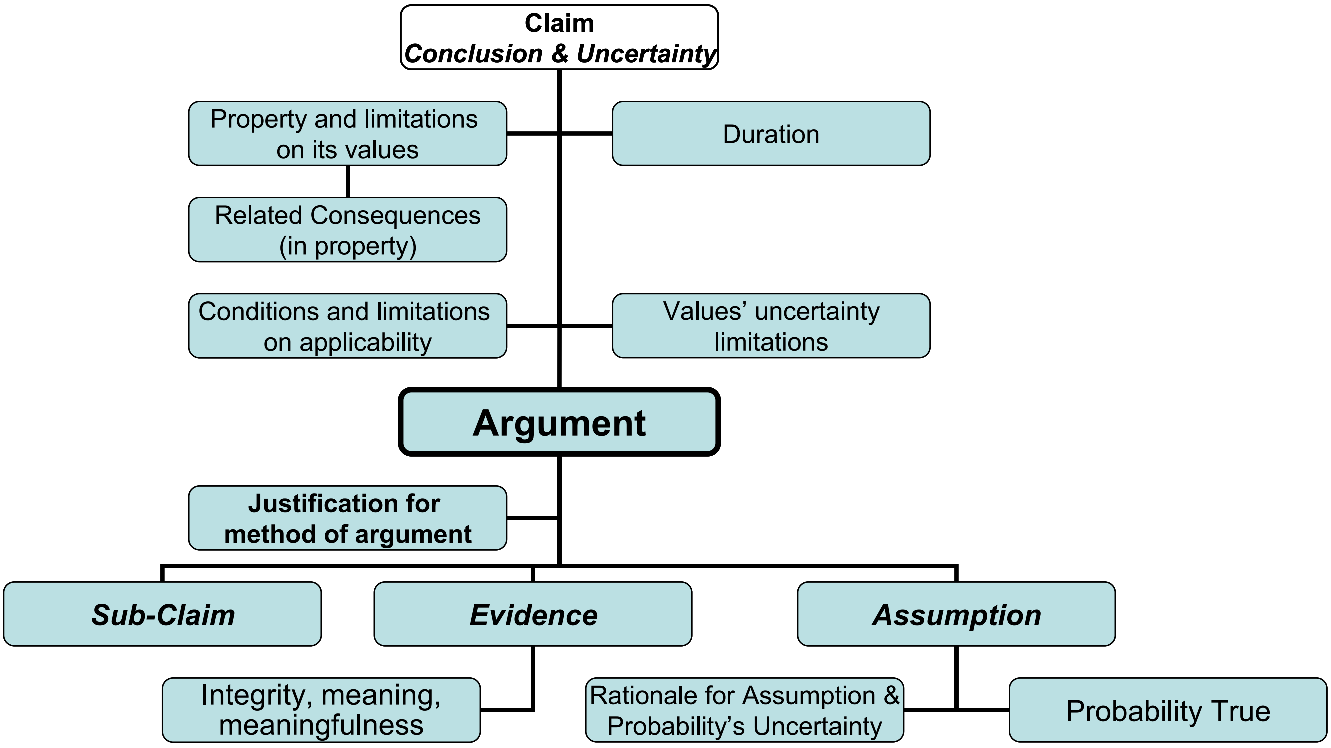

Claim
Conclusion & Uncertainty
Property and limitations on its values
Duration
Related Consequences (in property)
Conditions and limitations on applicability
Values' uncertainty limitations
Argument
Justification for method of argument
Sub-Claim
Evidence
Assumption
Integrity, meaning, meaningfulness
Rationale for Assumption & Probability's Uncertainty
Probability True

Figure 3 — Argument Context

This sub-clause briefly discusses the variety of reasoning methods that one might see or consider. This is followed by covering the roles arguments might play in an assurance case, and several issues concerning the bases for and the structures of arguments.

9.3.2 Roles of arguments

Mainly, arguments derive roles from their place in the structure of the assurance case including their local place within the larger argument. They also may have roles in their use by stakeholders such as communication among and use by decision makers. Roles include yielding the combined effect of the evidence, sub-claims, and assumptions that they utilize, providing a second argument in support of claim, and replicating the same argument with different support – usually different evidence.

In addition, issues arising from the assurance case arguments for the property of interest and different properties or qualities can highlight tradeoffs with other properties or functionality.

9.3.2.1 Deriving support for claim from support for argument

An argument needs to argue that what supports it is relevant to supporting the claim and that it meaningfully combines what supports it into support for the claim whose meaning and uncertainty reflect those of what supports it and the nature of the argument. Many kinds of reasoning are possible. Some produce stronger results than others and different ones may be appropriate for different situations. For example, using probability related to occurrence of a natural event contrasts with the need for concern for the possibility of an intelligent, malicious action. Regardless of the method used to reason and decide, it can be difficult to gain adequate information, understand, and structure or model the situation in a way that results in very low uncertainty. This can call for changes.

9.3.2.2 Combining supports for a claim

When they are used to combine supporting evidence, sub-claims, and assumptions into support for a claim, different systems of reasoning vary in their applicability, power, resulting accuracy and uncertainty, and ease of use. Among rigorous methods, the use of probability-based methods to do this combining has the longest and in many ways the most successful history. As mentioned elsewhere, in some situations its applicability is difficult, questionable, or unsuitable (Sub-clause 9.3.3.2).

The items supporting the argument have uncertainties associated with them and the argument can increase or reduce uncertainty. A method of argument can be an additional source of uncertainty.

As covered in 9.3.2 and elaborated in 9.3.3.2, many systems of reasoning can be applied differing situations. Some are more rigorous than others and some are quantitative and some not. Human judgement can frequently play a role. As the contents of 9.3.3.2 indicate, this combining of supports is often a difficult and important one.

9.3.2.3 Separateness or independence arguments

Three situations might exist with multiple arguments all supporting the same claim:

- Different arguments with same support (e.g. supported by same or essentially overlapping sets of evidence).
- Same argument with different support (e.g. different evidence).
- Different arguments with different support (e.g. different evidence).

Replication of experiments has an important place in science. Replication by someone or organization other than the original can be helpful in confirming a result. Replication can help establish that previous results were not dependent on some unreported aspect and that they were not the result of using an improper method; lack of proper interpretation, fidelity, competence or skill, or care in enacting the method; mistake in recording, analyzing, or communicating (including understanding) description or results; statistical fluke; or maliciousness. Aspects may be unreported or inadequately or mistakenly reported because they are not noticed, recognized as potentially relevant, accurately observed or measured, accurately and completely recorded, or reported in disagreement with records or in an ambiguous or difficult to understand way.

The more independent and different the conductors of replications are from the original conductor (and each other), the less likely unrecognized or unreported aspects may exist that potentially could affect results, or their meaning or meaningfulness (significance).

Multiple arguments reaching similar conclusions using distinctly different methods or based on different conceptual bases generally add credence to the conclusion.

9.3.2.4 Modification of arguments

Modification of arguments can be needed because of changes or because of their weaknesses – unsatisfactory method, execution, conclusions or convincingness (e.g. excessive uncertainty). The structure of the assurance case's argument, particularly modularity and mapping to product design, can make it easier or harder to create, understand, and modify [133]. Automated tools to aid in recording, maintaining, and managing assurance cases can also help.

9.3.3 Reasoning

9.3.3.1 Introduction

Subjects of and approaches to reasoning differ among different communities. They have differing motivations, mindsets, and methods of reasoning – often multiple ones. Systems of reasoning

- "Quantitative":
  - Deterministic (e.g. formal proofs)
  - Non-deterministic formal systems for reasoning:
    - Probabilistic
    - Game theoretic (e.g. minimax)
    - Other uncertainty-based formal systems of reasoning (e.g. fuzzy sets)
- Qualitative:

- E.g. staff performance evaluations, court judgements, and qualitative statements of event causality.

Some examples of ways of systems of reasoning and examples of their ways showing something is true (possibly with some uncertainty) are listed in Table 3 — Examples of ways of showing.

Table 3 — Examples of ways of showing

Logic
Reduction (e.g. infer by laws of logic, definition, substitution, simplification)
Generalize to collection from arbitrary particular member
Contraposition
Contradiction
Induction
Show existence true by an example
(Show false by counterexample)
By Cases
Probability
Inference (e.g. probability theory)
Induction (e.g. statics, observations, experiments, tests)
Models
Simulation
Analysis
Agreement among
Multiple methods or instances of showing
Suitable humans

Generally, quantitative or rigorous reasoning systems and methods are considered preferable within engineering. While this certainly appears to be the case with many engineered artefacts and natural phenomena, and humans are known to make certain kinds of human errors, whether this is universally true is by no means clear [178], [210], and [187]. Historically, the assurance case has often been held together by values, uncertainties, and relationships dealt with using probability-based methods such as statistical confidence, decision theory and Bayesian networks.

Complex products and situations – and any involving humans – may be beyond the current state of the art to "quantitatively" create predictions of the nature needed – much less precise and accurate ones. In addition, supplementing quantitative techniques with expert review – and judgement – is widely used and generally accepted as being a wise necessity.

While sometimes necessary or possibly advantageous, use of subjective judgement within the assurance case can lead to problems or additional uncertainties, so, generally, (just as with assumptions) the less critical they are the better. They are used in the absence of affordable, suitable, more objective methods and techniques or where needed to supplement or evaluate the results of such techniques. As with other forms of argument, subject judgements take the form of a claim and its support. Table 4 lists some of the communities and activities having their own – although sometimes overlapping – mindsets and approaches.

Table 4 — Some viewpoints and approaches to reasoning

Mathematical	Safety	Research
Security	Engineering	Correctness
Project Management	Counterintelligence	Risk Management
Crime	Financial	Regulatory
Executive Management	Industrial competitiveness	Subversion
Political or social activism	Litigation/Liability	Espionage
Marketing	Buyer	Terror
User	Diplomacy	Revolution
Intelligence analysis	War fighting	Attacker
Natural disaster		

Generally, judgements are expected to be of higher quality if they reflect an agreement among multiple persons who cover the full scope of knowledge of the situation and the necessary relevant expertise and experience, and are consistent with known facts. Group decision making has its problems [32], but decisions made in isolation have their own risks. As with other forms of argument, their conclusion needs to be accompanied by an estimate of its uncertainty and be reviewed, recorded, and acceptable to approvers of the assurance case. Finally, if as the result of using human judgement or another form of reasoning a risky amount of uncertainty results, stakeholders relying on the assurance case may need to be warned.

The legal profession has long faced similar problems and has developed methods that many engineers might question, but which reflect what one might think of as a long (and ongoing) history of experiment. [46] Making decisions through conflicts may not be the best way to gain creditable evidence and a good understanding, but it does automatically supply doubt and question the behaviour and possibly the methods being used. This method can lead to categorical thinking, inflexibility ignoring the middle or compromise position, a strong desire for certainty rather than explicit recognition of uncertainty, and living and dealing with it. However, this method has a more moderate parallel in the technique of multiple working hypotheses.

On the other hand, scientific reasoning appears willing to keep the question open and the decision unmade until enough evidence and understanding are achieved and willing to later reopen it. Among others, scientific reasoning tends to use induction, deduction, analogy, experiment, theory formulation, causal reasoning, and problem solving techniques. [44] Generally, theories that are at least theoretically refutable and make testable differing predictions than competing theories are preferred. Lately, experimenting and investigating with "computational" models (e.g. simulations) has gained prominence and a role alongside theory, and real-world experimentation and data collection.

Engineering has been characterized as applied science and mathematics, and to some extent this is true. On the other hand, engineering often has proceeded in many areas on an empirical basis with little or no theory. And, it has also been characterized as advancing by learning from failure [167]. Engineering shares with the legal profession the need to make immediate decisions and with the sciences the desire for making decisions on a firm basis. Regarding venturing into uncertain areas and risks, science uses the concept of "informed consent" of stakeholders, and engineering codes of conduct address the problem of working within one's abilities.

As mentioned, a variety of bases for argumentation and analysis might be used. Choosing the one (or few) to use can include several factors. While these will not all be listed here, one must consider that some arguments can be more complex or difficult to perform than others. However, choosing a tool because it is simple and easy, easy-to-use, or the engineers are most familiar with it will not always be the best choice. While engineering simplifications can be appropriate, ultimately bases and methods of reasoning and arguing need to yield results that adequately reflect and do not contradict reality.

Some arguments may be of appropriate form, but nevertheless inadequate in practice for a real product. For example, consider the following argument. If the system is in an acceptable (e.g. safe) state, each (and every one) of the individual actions within the system will, whether done concurrently with other actions or not, result in the system being in an acceptable state. Therefore, if started in an acceptable state, the system will always remain in an acceptable state.

The possible practical problems with such an argument include:

- Some actions when performed under certain circumstances actually do take the system from an acceptable state to an unacceptable state.

- The premise concerning concurrency turns out not to be true.

- Conditions or events happen that violate the assumptions of the argument (e.g. conditions outside of the conditions under which the system was designed to operate and behave properly).

- The system was not designed to handle certain situations (possibly the designers never thought of them).

- States thought at design time to be acceptable have unintended or unanticipated consequences that are unacceptable. These could be in the environment or, for example, interference among parts.

Underlying many of such practical problems is the problem of achieving universality – for example the need for everything to be correct, deal successfully with everything that might happen, or that all consequences will be tolerable. Generally, arguments must cover situations where one or more universalities is not previously being achieved or is not true. This, of course, requires that the product or its environment have provisions for these situations either individually or collectively. A simple example of a collective provision might be that if anything not within the acceptable set of events or conditions occurs, the product shuts down and humans are notified.

9.3.3.2 Probability *versus* possibility

The patterns of occurrences of "natural" events and common, non-malicious human behaviours are usually described probabilistically. The probability of a natural event contrasts with the need for concern for the possibilities open for intelligent, malicious actions whose probability is not determinable or not knowable particularly if the adversary deliberately violates any probability estimates one makes regarding its behaviour –

for example to achieve surprise. This distinction is central to the difference in reasoning between safety and security.

Combining probabilities can lead to an expected result. However, combining instances of possibilities is difficult to do in a way that does not simply result in the worse (or best) possible case. Knowing the worst case can lack usefulness to decision makers who must consider limited resources to overcome it – and its occurrence might never or quite rarely happen.

9.3.4 Structure of arguments

Overall assurance cases often make arguments falling into one of two patterns. Crudely, these two argument patterns are (1) nothing significant went and/or is and/or will be too wrong and (2) everything necessary went and/or is right, and/or will be or close enough (through the duration of applicability of top-level claim) – contrast [181] and [182] with [55].

Both have their difficulties. The first requires identifying everything significant that might go or be wrong. This is usually called risk identification and analysis. The second pattern, to be practical, must either argue that only the aspects it covers are significant for the assurance case or that this is true within the portions of the assurance case where the pattern is being used.

Each individual argument within the overall assurance case has the objective of showing its immediate super-ordinate claim or claims from its immediate subordinate supports. This can be achieved either directly from the evidence and assumptions or by breaking the claim into parts which are or are related to its immediate subordinate supports, e.g. sub-claims.

In the latter case, generally all of the things needing to be shown to be true in a claim are carried down and allocated to one or more of its sub-claims, and the argument shows how the combination of these sub-claims leads to achieves adequate support for the claim. This combining generally combines sub-claims by using specified arrangement(s) – often arrangements reflecting system structure. Specifying the arrangement(s) is usually necessary to effectively make the argument that composes the sub-claims into the claim. As a side effect, breaking down claims can lead to many of the requirements stated in one claim being repeated in its sub-claims, sometimes verbatim, sometimes slightly enhanced.

In addition, assurance cases needed sub-arguments covering the integrity of the assurance case (e.g. lack of tampering) and possibly the validity of the evidence.

To argue a claim directly, evidence must be adequate (or assumptions) and a number of things must be included in the argument. For example, SafSec divides the ways of arguing or showing into seven non-exclusive ways and calls these “frameworks.” Most important, compliance with each framework means meeting certain standards, such as the organizational roles being defined, including standard ones. Thereby, they provide one categorization of the rationales for use in direct arguments. For a single claim, many of these "frameworks" must usually be involved. [[182], pp 24-28] Evidence for a modest sub-claim may be as extensive. Sub-clause 9.4 covers evidence.

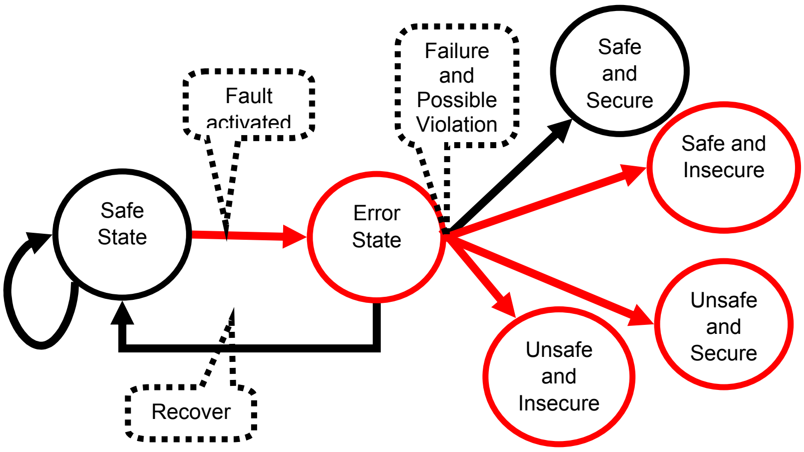


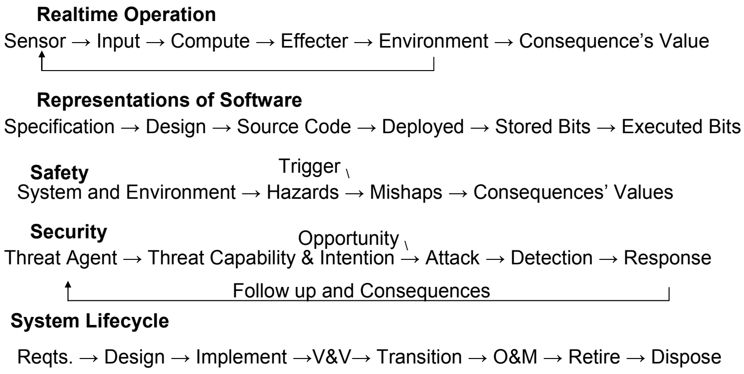
Figure 4 — Simple State Model

While possibly the bulk of products and situations are somewhat complex, consider first the other extreme is the simple notional state machine in Figure 4. While clearly idealized, in a limited situation it might possibly provide a basis for thinking about and the structuring part of arguments before adding in details. However, real products and particularly their environments can be complex. Moreover, as pointed out below most significant systems are for several reasons hard – not simple – to model.

To better understand the problem, first consider reasoning using one of the widely used bases, cause-and–effect relationships. Some simplified “cause and effect” chains are shown in **Error! Reference source not**

found.. More importantly, physical processes and operator-system-interaction are often use cause-and-effect models to underlie argumentation. However, cause and effect is not as simple a concept as it might sound.

While sometimes straightforward, cause and effect relationships are many times difficult to define, complex, subtle or indirect. Table 5 lists some of the ways in which relationships are categorized and characterized, and its length indicates that several subtleties can be involved in using them in arguments. Cause and effect arguments are quite important, but in practice often judgemental [172] and questionable [59] as well as problematic in how humans learn and perceive them [23].



Realtime Operation
Sensor → Input → Compute → Effecter → Environment → Consequence's Value
Representations of Software
Specification → Design → Source Code → Deployed → Stored Bits → Executed Bits
Safety
Trigger
System and Environment → Hazards → Mishaps → Consequences' Values
Security
Opportunity
Threat Agent → Threat Capability & Intention → Attack → Detection → Response
Follow up and Consequences
System Lifecycle
Reqts. → Design → Implement →V&V→ Transition → O&M → Retire → Dispose

Figure 5 – Simplified "cause and effect" chains

Cause-and effect relationships can relate many factors to an effect or a single cause to many effects, and chain to together in complex relationships such as cyclic networks. For example, they can be highly sensitive or non-linear, multi-way, cyclic, dynamic, involve feedback, involve humans as well as physical phenomena, and reflect coincidental as well as systematic timing and linkages as well as possibly being emergent.

Table 5: Relationship aspects that are possible bases for or relevant to arguments	
Existent or non-existent	Extensive or limited
Dependent or independent	Strong or weak
Established or postulated	Sensitive or insensitive
Known or unknown	Acyclic or cyclic
Credible or not credible	Positive or negative feedback loop
Plausible	Stable or unstable
Deniable (plausibly or convincingly)	Intentional or unintentional
Reputable or non-reputable (non-repudiation)	Purposeful
Long-time or new	Non-malicious or malicious
Permanent or temporary	Trustworthy or untrustworthy
Well-established or tentative	Trusting or untrusting
Common or uncommon	Private or confidential, or public or exposed
Frequently occurring or unique	Goal-directed
Ubiquitous or local	Multiple-objectives
Invariance	Cooperative or uncooperative
Correlation	Competitive or non-competitive
Cause and effect	Giving, taking, or sharing
One-to-one, one-to-many, many-to-many	Supplier-consumer
One-way or two-way	Request and receive
Static or dynamic	Centralized or decentralized
Direct or indirect	Peer-to-peer or controller-slave
Straight-line or roundabout	Use shared resource
Simple or complex	Support shared dependent
Positive or negative	Shared variables
Direct or inverse	Message passing
Re-enforcing or detractive	Decision-making
Affect increasing or decreasing (e.g. force multiplier)	Informational, physical, or social
Supportive or unsupportive	Phenomenological or -ological (e.g. geological, sociological, chemical, electromagnetic, quantum-mechanical)
Enabling or hindering	

Two link-related aspects of cause and effect that need to be considered are (1) common cause failure where the common links are not well understood, (2) multiple coincidental events (not all of which are linked) that can together cause an undesirable consequence.

Example Consider an accident caused by three events: (a) the operator was late to work because of a weather-related traffic jam, (b) the river near the plant overflowed, (c) there was an unnoticed crack in the plant foundation. Events (a) and (b) have a common cause - a storm. Event (c) is independent.

The concept of “resonances” has been introduced as one approach to thinking about complex systems, lack of predictability, and self-caused events [59]. While one aspect of the approach fits well phenomena such as rogue ocean waves, for many situations it is an analogy. A related term is "normal accident." Nevertheless, whether complexity is desirable or not, relationships are often complex.

In part, the difficulties sometimes experienced with reasoning about cause-and-effect and some other common modelling techniques result from the system's behaviour and combined effects from interacting with its environment usually being emergent phenomena – both normal performance and many failures. This means the relationships between individual components or aspects and the overall result can be subtle and complex and, therefore unsurprisingly, what will happen is hard to model or predict.

NOTE Describing the resulting situation in the context of the safety of modern complex systems one expert stated in a 2007 presentation, "Explanations of accidents cannot be limited to “component” failures and malfunctions — either alone or in combination. In complex socio-technical systems, accidents often arise from normal performance variability that interacts in unintended and unanticipated ways. The target for safety management should not be to reduce risks, but to increase the intrinsic ability on all levels of a system to adjust its functioning in the face of changes and disturbances (resilience)." [58]

Thus, the assurance case needs to cover all the conditions and events that could have a significant negative affect on the conclusion regarding the top-level claim including on its uncertainty. The potentially relevant universe of conditions and events can be hard to initially identify [2], and ascertaining which might have a significant effect can be difficult without at least initially including them in the assurance case. In some cases, the conditions and limitations associated with the top-level claim provide a limited universe, and it can readily be covered. However when this happens, it can be an indication that the assurance case is taking an unrealistic approach.

For any product interacting with humans and possibly if it is just near humans, human factors will normally raise conditions that must be covered. Among the dimensions of variation that need to be considered are time (absolute and duration), activities and tasks (use, administration, maintenance, transport, storage, installation, retirement, disposal), possible conditions in environment (e.g. weather, vibration, stacking, and surfaces), input and interactions, human characteristics and/or behaviour, transfers of control (e.g. ownership, custody, leaser, theft, capture, seizure), or modes of operation or product events or conditions. Others include opportunity or danger and their sources of, uncertainties and/or sources of uncertainties, phenomenological causes.

Product dependences are an important source of concerns, and possibly most importantly are kinds of consequences including affects on stakeholder interests. History, analyses, and product characteristics and qualities can give clues to what might happen. Many lists devoted to particular domains, industries, kinds of products, locations, environments, or qualities exist and can be consulted. Finally, to aid in addressing this identification, Annex D provides substantial but high-level lists and a number of references to online lists and other relevant material.

NOTE One set of conditions and events that should be avoided is composed of those requiring the existence of functionality or features in the product that are not part of its specification or not needed. However, if these extras are unavoidable then the resulting possible events and conditions need to be covered.

Historically, some kinds of conditions or events have received more attention than others. Perhaps, the problem aspect that has received the most attention is product (e.g. system) failure. A substantial volume of checklists, practice, and literature exists concerning product failure (e.g. [2], [76] Annexes A and B, [17] Chapter 18 page 475-524, and Annex D below particularly sub-clause D.3). While much of this work has been done in the communities addressing safety, security, or human error, product failure can result in less achievement of a positive property or consequence as well as negative properties or losses.

An assurance case will be more likely to approach completeness if it includes consideration for possibilities that are:

- Known items with relevant information about them known (obtainable) – ensuring none are overlooked.

- Known kinds of items with the relevant instance's existence, characteristics, or values unknown (known unknowns).

For completeness, one also considers the possibilities of:

- Known kinds of items whose existence, characteristics, or values are known but their relevance is unrecognized (unknown knowns).

- Items not known to be relevant or to exist, and nobody knows their characteristics or values (unknown unknowns).

Recognizing that the reality being argued about is often complex, the overall assurance argument needs to be broken down into levels of claims and sub-claim(s) where the objective is for the sub-claim(s) to be easier to show or closer to the evidence than the claim above them. A claim can be transformed into sub-claim that implies it, and/or it can be broken into parts that together imply it.

This connection of levels and the arguments that connect them are typically developed using both top-down and bottom-up approaches:

- The top-down approach breaks the claim down so smaller, more manageable arguments can be used in justifying the claim.

- The bottom-up approach identifies potential sub-claims, evidence, and assumptions and either:

    - Tries to show the desired claim from them

    - Simply asks what can be shown from them (almost always done but particularly done when assurance case is not being built to show a predefined top-level claim).

- A gap analysis (comparing the top-down and bottom-up results) may be used to identify what additional argumentation or argument support (sub-claims, evidence, or assumptions) are needed to justify the claim.

One basis for division into sub-claims is an argument "by cases". That is an argument that argues that (1) the claim is true for each of a set of conditions (e.g. night and day, temperature ranges, range of sizes) and (2) the set of conditions together subsume the complete condition under which the claim needs to be true.

The same does not follow from showing something is true of each (and every) instance. As will be elaborated below, the statement, "It is true of each part, so it is true of the whole," is often false.

An argument is only relevant for the condition (often including context) it presumes and/or applies to. The same is true for sub-claims, evidence and assumptions. They must apply to the relevant condition and this need can be used as a criterion to determine which evidence to create or use, what an assumption must cover, and which of the (sub-)claims already shown to be true might be used to support an argument. This does not mean that evidence from similar, but not identical conditions or situation is irrelevant. On the contrary evidence concerning prior versions of the product or the same product used elsewhere can have substantial relevance even though not normally as much as direct relevance as similar evidence from the condition or situation to which the claim applies.

Not only is evidence better if it is from the conditions that the claim applies to, but evidence is needed that together is relevant to the entire relevant condition. Showing that together this is true of the evidence is another argument that brings its supports together to show its claim.

The task of the breaking a claim into sub-claims that are connected to it by an argument can be difficult. However, several bases can be used for doing this. Many will be mentioned here.

Examples of ways of subdividing arguments include argumentation over subsystems, over life cycle or usage phases, over modes of operation, over kinds of use, over conditions of environment, over phenomenological aspects, over levels of abstraction, over terms used in a claim, based on combining multiple measurements, or multiple other properties, over development activities, over test results and other evaluative results, over history possibly over of a collection of histories of product instances, over risks, over causes or partial solutions, over kinds of consequences, or argumentation using some other existing analysis or structuring method (e.g. HAZOP or Ishikawa categories where appropriate).

Some are based on the effects of composition of components within the product or their combination with aspects of the environment. Examples range from the composition of metals in an alloy to the composition of multiple software subprograms into a program. In these cases and for most methods of sub-dividing, predicting the effect of the sub-claims or components when combined requires considering their interaction rather than simply "forming the union" of the components or sub-claims. As mentioned earlier, the statement, "It is true of each part, so it is true of the whole," is frequently false and invalid.

Therefore as mentioned earlier, prediction regarding combinations of elements generally requires specifying the arrangement of these elements – often these arrangements reflect the product or environment structure. As the arrangement of parts within a system is intended to produce certain results, the necessity to include this arrangement in the argument to predict some property of the result is unsurprising. Thus, often the design (or other) rationale can essentially contain the same argument needed in the assurance case as they have the same purpose – to show something the designer was trying to achieve is or will be true. Likewise, the same parallel between rationales can exist for implementation, manufacturing, and other activities because the rationale for why they are the way they are includes concern for the properties of interest to the assurance case. Together, they are trying to achieve them and show this achievement.

This points to one of the advantages of concurrent development of the product and its assurance case. The development process and the product can be aimed not only at achieving the claim but doing so in a way that can be shown to be adequate by the assurance case. The assurance case influences the product for example by causing it to be such that an argument is more practical to construct. This often results in a simpler product (at least internally), a product whose parts can be used in isolation to show certain sub-claims, and an arrangement of parts such that reasoning about the composition is both within the state of the art and practical. As far as the process doing them concurrently examples include requirements covering more conditions and events as well as adequate resilience, methods being used that produce few faults, and validation and/or verification being targeted to what is needed to be shown and showing that adequately.

NOTE To support the assurance case, one commonly needs the execution of a planned and systematic set of activities to provide grounds for confidence in product properties. These activities are designed ensure that both processes and products conform to their requirements, standards and guidance, and defined procedures [[152], I.A]. "Processes", in this context, include all of the activities involved in the design, development, and sustainment of product. For software, "software products" include the software itself, the data associated with it, its documentation, and supporting and reporting paperwork produced as part of the software process (e.g., test results, assurance arguments) as well as whatever else is needed to complete the assurance case. The "requirements" will include requirements for the properties that should be exhibited ultimately based on requirements to limit, reduce, or manage property-related costs and losses. The “standards and guidance” may be technical, defining the technologies that can be used in the software, or they may be non-technical, defining aspects of the process that are further delineated by the “procedures” that make satisfaction of the product’s requirements possible. [4]

As mentioned earlier, some lacks of universality mean that for an argument to be constructed requires the product or its environment to make provisions for this that can be used as bases for argumentation. Many approaches exist sometimes under such labels as fault tolerance, safeguards, safety or security controls, safety margins, risk mitigation, and risk sharing (e.g. by acquiring insurance).

NOTE 1 In addition to making such provisions, the problem can be reduced by such actions as increasing generality of product, designing conservatively or with safety margins, preventing or avoiding problems or their early detection and removal and exploiting the product environment to obtain help while avoiding over-reliance as well as attempting to achieve universality within the areas where universality is feasible (e.g. manufactured bolts are all within tolerances) – and perhaps also when coming close (or even closer) is feasible. Why have unnecessary problems?

NOTE 2 For software, [171] provides coverage of fault tolerance and [17] Chapter 18 addresses handling system failures.

Two objectives are common ones:

- Be resilient in response to events or conditions.

[4] “Do the right thing, do it right, and show it is right.” Sam Redwine

- Limit damage or decreases in benefits.

Among the principles sometimes mentioned in connection with resilience are redundancy, diversity, separation, generality, flexibility/adaptability, and restricting dependence. However, a system can have several objectives and activities related to resilience and limiting damage including:

- Forecast events and conditions.
- Maintain readiness.
- Detect events and conditions (desirable and undesirable particularly the latter including precursors, warnings, near misses, and suspicious events).
- Notify and warn.
- Record (e.g. via logs).
- Separation (e.g. by distance, time, barriers, or flow control).
- Continue service, although possibly degraded.
- Damage confinement (including by isolation and risk sharing).
- Diagnosis.
- Repair.
- Put product in a proper state:
    - When current state is detected or inferred to be illegitimate (recovery).
    - Preventively (e.g. regardless of knowing if needed or if indications exist that might otherwise later experience problem).
- Flexibility and the capability and tactics to successfully adapt and deal with events and conditions.
- Reserves and reserve capacity.
- Characterization, analysis, investigation of root cause or causer.
- Operational "safety" margins.
- Learn and improve
- Arrangements or agreements with entities in the environment to provide aid (possibly including alliances).

The issues also exist concerning the readiness for, response time, speed, capacity, efficiency/cost, and efficacy of these list entries along with doing them when not needed and not doing them when needed. Flexibility and adaptability are often provided, at least in part, by humans.

Thus, arguments can be based on a variety of kinds of evidence and a variety of means of reasoning. Some evidence (e.g. results from certain types of testing) can be relatively easy to ascertain the meaning of, have a known uncertainty, and even easily have their meaningfulness to the argument established. Others can have meanings that are hard to clearly identify, cannot be readily generalized, or leave huge amounts of uncertainty (e.g. deriving from inadequate or incorrect sampling).

A close connection exists between argument and evidence. The evidence needs to support the argument used, and the argument needs to be such as to effectively use the evidence – evidence either customarily obtained or especially identified or designed (e.g. especially collected or created). Constructing arguments to include the use of evidence that already exists or will be created or collected anyway is often efficient and/or necessary. However, custom evidence can be needed to fill gaps in this evidence, and it can be designed to provide especially effective support.

Two final concerns deserve mention. Firstly, arguments based on the usage of standards or practices can be strong or weak depending on the origin and track record of the standard or practice. Some have excellent records. However, their success in some organizations or a set of organizations with certain characteristics (e.g. having a strong quality or safety culture) does not ensure similar results will be seen if used elsewhere. Secondly, novelty can add to difficulties. The history of engineering has many examples of new artefacts substantially different from older ones that were not successfully created or did not work properly [167].

9.3.4.1 Seeking to contradict

Experience in several fields has shown human propensities to perceive things that support their existing opinion or desired outcome while not perceiving or misinterpreting things that do not and occurrences of groups reaching agreements among themselves with this very agreement causing a strong loss of objectivity which adds to the human tendency to conform. Several fields have institutionalized a process that systematically introduces contradictory evidence and argument normally by having participants whose explicit role is to do so. Examples include legal trials and system engineering red teams.

Efforts constructing assurance cases need to be concerned with these tendencies. In addition, an argument is to some degree strengthened if an effort to contradict it fails where this effort could reasonably be considered serious enough that contradictory argument or evidence would have been identified if this were possible. Even when such efforts are not successful in contradicting the assurance case argument, they can identify issues, weaknesses, conditions, events, or other possibilities that need to be considered by the assurance case but which were not previously being considered. They may also contradict only some portions of the assurance case causing them to need to be reworked.

A decision needs to be made concerning the kinds and amount resources to devote to an effort at contradiction. Leaving plausible areas for discovering weaknesses or errors unconsidered would appear to be unwise. The decision on amount might be made either initially or by observing the ongoing rate of return yielded during performance of the effort.

9.3.5 Conclusion

Arguments are the glue that holds the assurance case together. At each level from the top-level claim to underlying evidence or assumptions arguments relate the claims supported to support provided by its subordinates at the next lower level – sub-claims, evidence, or assumptions. Often an overall approach is used to structure the overall argument in the assurance case such as approach based on identifying all significant risks.

A variety of systems of reasoning can be used in argumentation. These vary in their applicability, power, resulting accuracy and uncertainty, and ease of use. The sub-claims, evidence, and/or assumptions supporting an argument have uncertainties associated with them, and the argument can increase or reduce uncertainty.

Arguments can need to deal with not just "normal" conditions but the others as well. They also need to deal with the possibilities that parts of the product will not behave as intended and that unforeseen events or conditions can occur including unintended or unforeseen consequences.

Assurance cases are frequently intended to and often do provide rationale for a product being appropriate for a use. Nevertheless, in the end despite everyone's best efforts, an assurance case can fail to show their top-level claim. One possible choice when this happens is to change the product; another is to strengthen the assurance case. However, particularly for products that exist or are well along in their development, often the questions asked are:

- What claim can be adequately argued?

- What use can be made of the product given this weaker or different claim?

Possibly the product can still be used under limited conditions, in a smaller market, or with added safeguards in its environment. If nevertheless used as originally intended, stakeholders must accept added risk or tolerate reduced benefits or other adverse consequences beyond their original intentions. Alternately, the conclusion can be made that the product should not be used or possibly not be developed.

NOTE Standards or approaches labelled has being for "evaluation" or "assessment" can sometimes be useful in identifying argumentation methods or methods of combining evidence as well as in identifying relevant information for use as evidence. This is true for "evaluation" or "assessment" of product, process, technology, organizational aspects, and particular qualities.

9.4 Evidence

9.4.1 Introduction

Evidence can be generated as a customary result or artefact, or because it is needed for the assurance case or for certification or licensure. All evidence might be useful in the assurance case, and no evidence should be unthinkingly ignored. Sub-clauses cover General issues, Meaning and meaningfulness, Kinds of evidence, and Assessments, certifications, and accreditations.

NOTE 1 Depending on how the creator of the assurance case conceptualizes it and the amount of inference that has already been done from the evidence beforehand (outside of the assurance case), evidence can be said to be supporting an argument or directly supporting a claim.

NOTE 2 A distinction is made between direct evidence that reflects relevant properties directly and backing evidence that is concerns the nature, quality, characteristics, and history of the evidence.

9.4.2 General

When judging the credence to be given to a piece of evidence, its relevance, visibility, traceability, and quality are crucial factors. Therefore, one must necessarily confirm that the evidence is generated or collected, managed, validated, stored, and used within the constraints of acceptable practices and controls. It must also achieve the objectives claimed in the assurance argument [[146] MoD DefStan 00-42 Part 3, section 9.1]. The body of evidence can become quite large, and for usability probably needs to be organized by some evidence framework and the assurance case aspects supported by tool(s).

Among the areas that evidence can be derived from are the:

- Entire lifecycle and across the supply chain.
- Environment.
- Intentions.
- Process.
- Means and resources (including people and tools).
- Products.
- Field experience.
- Support.
- Capabilities (possibly not yet exercised).

For any area or property, many means of obtaining evidence exist. Among these are human experience, history, observations, measurements, tests, evaluative and compliance results, analyses, defects, and, and inferences from evidence. Evidence can already exist or be newly created or collected – or at certain times still be planned for the future.

Evidence should be obtained both for the argument and against the argument (counter-evidence).

The evidence needs to be of adequate quality. This involves issues concerning both its origination and its preservation and handling. Origination-related issues include feasibility, conformance to standards and procedures, validity, subjectivity, accuracy, uncertainty (e.g. measurement uncertainty), affordability, and ultimately credibility and usefulness. While essential to assurance practice, details regarding exactly how to measure, demonstrate, or analyse particular properties are not covered in this international standard. These are the subjects of more specialised standards and literature of which a number are included in the Bibliography.

SafSec states, "Evidence shall be permanent, traceable and managed in such a way as to provide later readers confidence in its source, contents and validity.” [[182], page 9]. Another guidebook indicates [160]:

- Evidence must be uniquely identified so that arguments can uniquely reference the evidence.
- Evidence must be verifiable and auditable.
- Evidence should be protected and controlled by the configuration management (CM).
- …evidence needs to be accompanied by the metadata needed to properly use it within the assurance case.

Weaknesses in its validity or integrity can significantly affect – even destroy – evidence's usefulness.

9.4.3 Meaning and meaningfulness

To properly use evidence in support of an argument (or claim), both its meaning and meaningfulness need to be established. The meaning(s) of evidence can be easy or difficult to ascertain. Its meaning generally reflects the:

- Subject of the evidence.
- Properties about which the evidence is relevant.

For example, one might need to ask, "What are we really measuring here?"

The evidence's meaningfulness to each argument (or claim) that it supports is usually influenced by:

- Its accuracy and the uncertainty associated with its accuracy.
- Its generalizability beyond the instances from which it directly originated.
- The remaining uncertainty about inferences from the evidence (e.g. as reflected in its statistical significance in the testing a hypothesis).
- The relevance of its meaning to the argument's (or claim's) subject and its attributes (e.g. entity, property, conditions, and times of interest).

Usefulness of evidence can be reduced if it cannot be stated in terms that can be used by the system of reasoning being used, for example not stated probabilistically. The usefulness and meaningfulness of evidence is addressed in several standards or guides. For example, [160] states:

- Evidence must be sufficient for its use in the assurance case arguments, including both its quality and its provenance [history].
- The stated context and criteria apply to each piece of evidence. [e.g. relevance to version, and conditions and duration of applicability]
- Where inputs, states, or condition can vary, it must cover all possibilities or have a sufficient sample to justify the argument.

The last point addresses generalizability. Many analyses cover all possibilities but few tests do. However, some tests may ensure detection of all faults of a certain kind. In addition, some tests are more generalizable, for example if they use a statically sound sample.

Documents [196] and [198] include material relevant to evidence in general and from analyses, testing, and field service experience. To give a more detailed example, the following list covers points related to making testing evidence more meaningful (and provide backing evidence) is loosely adapted from these two sources and enlarged:

a) Test guidance, procedures, standards and tools defined.

b) Tools used validated and verified.

c) Test procedures are validated and verified.

d) Test equipment calibrated and resulting certificates available.

e) Testing covers where possible and practical:

   1) Complete top-level claim.

   2) Needs of arguments (or sub-claims) directly supported by testing.

   3) Relevant properties.

   4) Top-level claims conditions and what is possible during its duration of applicability.

   5) Adequate coverage of the input domain.

   6) All possibilities or is such as to be generalizable to them.

   7) (Any aspect not adequately covered by tests is covered by another method such that altogether adequate coverage results.).

f) Tests reflect needed (low) uncertainty:

   1) Needed uncertainty reflected in documents governing testing.

   2) Needed uncertainty reflected in test specifications.

   3) Complexity of claims and related input analysed and used in selection of test data.

   4) Consequences of failing analysed and used in selection of test data.

   5) Tests adequately thorough.

   6) Needed uncertainty reflected in test criteria and criteria for ending testing.

g) The test methods and techniques used are appropriate for the properties under consideration.

h) Rationale for item and item recorded, reviewed, and subject to audit for the following:

   1) Test specifications – created independently.

   2) Objective for each test.

   3) Test procedure.

   4) Quality evidence for test procedures.

   5) Test criteria (e.g. for acceptable test results) complete and correctly reflects support needed.

   6) Test results.

   7) An analysis of test results.

   8) Detected and implied faults analysed.

i) Testing versus operation:

   1) Tests' configurations identical to operational.

   2) Differences between the operational and test environments identified and affects assessed.

j) Conduct of testing:

  1) Testing performed independently.

  2) Test guidance, procedures, standards and tools followed:

    i) Procedures or tools used to ensure:

      I) Testing follows test procedure

      II) Results satisfy the test criteria

  3) Test observed independently and reports produced.

  4) The test environment and activities recorded accurately.

k) Tests meet test criteria.

l) Testing results provide required support for arguments (or claims) directly supported.

This last point is simply a restatement of what testing is supposed to achieve related to the assurance case. This list does not include more general related evidence regarding testing such as personnel competence, and adequate time, resources, and facilities. Reflecting reality, the list is long and varies from concerns that apply to all testing to those some might use only when unusually low uncertainty is required. The sources give limited indication of where some points lie along this dimension.

9.4.4 Kinds of evidence

The introduction (9.4.1) provided a broad list of areas from which evidence might be derived. To elaborate further, somewhat obvious steps and evidence that can contribute to the assurance case include:

a) Ensure for the areas listed below:

  1) Their adequacy (as exist or performed).

  2) Adequate observations and measurements are created and/or collected, and recorded.

b) Areas:

  1) The quality and history of the people who produced it.

  2) The quality and history of the tools used in producing it.

  3) The quality of the environment in which it was produced.

  4) The characteristics and history of the process, activities, tasks, methods, techniques, and technology used to produce it.

  5) The quality of the definitions, policy, and procedures governing the process, activities, tasks, methods, and techniques used in production and the fidelity with which they were followed.

  6) Characteristics of the designs including the extent to which they can be practically reasoned about and provide resilience.

  7) Quality and results of analysis and simulations.

  8) Results of reviews, audits, tests, and other evaluations of product.

  9) Proofs.

  10) The execution and operations history of the product

11) Related experience with and consequences.

NOTE IEC 60300-3-2:1993, Dependability management – Part 3: Application guide – Section 2: Collection of dependability data from the field [69] can be relevant – even in some situations not involving dependability properties.

12) Indications of the realism of the assumptions made.

This list is not likely to cover all the evidence that is needed. Consider the list in [181] and [182] that lists a number of areas of evidence on which risk-oriented arguments might be built:

- Organizational: the goal is achieved by some organization.
- Procedural: certain actions have been carried out.
- Risk Directed Design: "document a justification for achievement, by the system, of each residual risk; and document a justification that the evidence of achievement of risk reduction is appropriate for the level and importance of the risk reduction."
- Modular Certification and Technical Modularity: organizational or system interfaces, particularly with external systems, need the "other" side of the interface to justifiably have the assured qualities claimed. In addition, "Module boundaries shall match the organizational boundaries."
- Evidence: requirements have been established for the recording, handling, and characteristics of evidence to be used.
- Evaluation/Assessment: the project documented a means of demonstrating the achievement, by the system, of each residual risk to a level of [uncertainty] appropriate for that risk, obtain agreements on evaluations and assessments among the parties involved, and carry them out successfully (as determined by evaluation/assessment results).

The need to support arguments should drive the evidence selected, and verification and validation activities. However, if the product is not as claimed, supporting evidence should be harder to obtain and contradictory evidence easier. Thus, similar to the need to achieve claims, the need for evidence drives development and maintenance decisions. These include evidence selection, generation, and maintenance as well as making this evidence easier to obtain.

9.4.5 Assessments, certifications, and accreditations

A substantial body of relevant experience and practices exists in the assessment, certifications, and accreditation communities, and this standard benefits from this prior work both outside and inside ISO and IEC. Certifications and their related techniques can add to the evidence available for the assurance case and an assurance case can supply evidence needed in certification. Many regulatory situations and certification processes do not offer the freedom to provide what product producers consider would be the best assurance case and use it for approval, certification, or proof of compliance.

For example, the aviation and nuclear power industries have long histories of standards and certification and the security community in ISO/IEC JTC 1/SC 27 has been working on the topic of assurance for many years. Security examples include the Common Criteria, FIPS 140 for cryptology, and *ISO/IEC 27002 Information technology. Code of Practice for Information Security Management* combined with ISO/IEC 27001 (formerly with UK standard BS7799-2:2002) form a basis for an Information Security Management System (ISMS) certification of an operational system.

The field of medical devices is gaining experience. As a final example, the UK Ministry of Defence and Civil Aviation Authority have also produced standards of interest including assurance-case-based standards for reliability, maintainability, and safety – e.g. [146], [149], [150], [196], and [197]. Many ISO-related standards are listed in the Bibliography.

Standards exist addressing the assessment of software and systems processes. Three examples are ISO/IEC 15504 *Information technology -- Process assessment*, the Capability Maturity Model Integrated (CMMI) from the Software Engineering Institute in the US, and ISO 21827 - *Systems Security Engineering Capability Maturity Model*.

Professional certification is also used in some areas. The safety community (e.g. commercial aviation) has utilized certification (designated agent or licensure) of key personnel as part of its approaches. A number of safety and computer security certifications exist from management-oriented ones to technical ones about specific products – for example, certifications from the International Information Systems Security Certification Consortium (ISC)^2 and the SANS Institute.

9.5 Management and life cycle of assurance case

Management and life cycle issues include activities directly involving the assurance case and the affect that the assurance case on other activities. For example, results would likely be best if the assurance case is considered from the beginning of concept development forward, it is used to influence all activities and products, and the assurance case becomes an integral part of the overall engineering process. These could all be done only if the product and the assurance case are being developed concurrently. However, the last two always apply during the duration or the top-level claim. The scope of the set of activities covered by this International Standard Part 2 is different if the assurance case is being developed concurrently with a product or not – e.g. for an already existing product.

Thus, the life cycle of the assurance case is not always the same as that of the product. It is covered normatively in ISO/IEC 15026 Part 2 only insomuch as is possible to adequately ensure the quality of the assurance case and its usefulness would not be clearly endangered by actions within processes, activities, and tasks. The full possibilities for relationships and integration between product and assurance case life cycles are covered in ISO/IEC 15026 Part 4.

Activities involving the assurance case can extend beyond its duration of applicability to cover such areas as archiving and obligations and liabilities remaining from its period of applicability. Process or activity issues include not only the “process” in a limited way but assurance case methods, practices, techniques, and tools as well as the responsibilities, competence, motivations, ethics, and independence (e.g. organizational affiliation) of all involved as well as the environment. An example is concern for the effort being well orchestrated possibly including having a single individual being responsible for the entire assurance case.

Concurrent maintenance is covered during the duration of applicability as well as during any concurrent development with product. Considering just the assurance case itself several specific activities must be done such as configuration management and approvals.

One of the principal uses of the assurance case is in risk management. Following ISO/IEC 16085 [97], all risks should be considered concurrently. In practice, this includes risks related to:

- The product and its life cycle.
- The assurance and its life cycle.
- The project.
- Organizations.
- Individuals.
- Assets inside and outside the product.
- The environment.
- Governments and regulators.
- Society and nations and their interests.

Clause 11 below more fully covers the assurance case within the life cycle processes.

9.6 Decision making using the assurance case

9.6.1 Introduction

The assurance case provides information and reduced uncertainty to decision makers and provide a basis for the confidence end-users need in the product before they feel comfortable using it, and it may provide the basis for confidence that the producer needs before releasing the product. These decisions – to release the product or to use it – are just two of many decisions that could benefit from less uncertainty.

While activities such as independent evaluation add to grounds for confidence, the bulk of the wherewithal for the assurance case might reasonably be expected to be satisfied as part of the processes that produce the product because without this how would the producer rationally have and maintain the confidence they need. Moreover, some might also reasonably say that the absence of such wherewithal would be grounds tending to support a determination of inadequacy.

Using the full assurance case is unsuitable for almost all decision makers. They need presentations with the relevant content in a form they can understand and use. Of course, care is needed to ensure the consistency of such presentations and the full assurance case and that needed information is not missing from them.

A user of the assurance case might need to make three differing decisions:

- How confident am I in the accuracy of the assurance case?
- Under what circumstances is the product trustworthy?
- Shall I actually place trust (reliance) in the product?

Related to these are the questions of:

a) What does the claim make as a claim about the product's (future) behaviour?

b) How good is the agreement of the product’s (future) behaviour with its claim?

c) And how uncertain should I be about my answer concerning the product’s agreement with its claim?

d) What will happen in the environment?

e) And how uncertain should I be about my predictions concerning the products environment?

Next asking:

f) What does the combination of answers a)-d) mean?

g) How uncertain should I be about my answer to what the combination means?

h) What will happen?

i) How uncertain should I be about my answer to what will happen?

And lastly:

j) What should I decide to do?

For many kinds of products, answering these questions has always been difficult, for example for large systems or pieces of software. Some assurance cases might only answer a) and a non-malicious version of c). This would appear to leave a number of questions unanswered. In theory, an assurance case should answer a) through i) and thereby possibly making the answering of what to do, j), easy.

9.6.2 Levels of assurance and confidence

The degree of confidence that can be or is justifiably engendered based on a specific assurance case may vary by individual or organization and the situation. The less uncertainty about assurance case’s claim presumably the higher the degree, or level, of justified confidence. Arguably, “high-confidence” is not a synonym for “low-uncertainty” or “high-assurance”. It is possible to have a high-level of unjustified confidence. This conversion of an amount of uncertainty (related to for example about the correctness of the as built product) into a level of justified confidence in suitability for certain applications is not straightforward or well understood. This can be exacerbated when maliciousness is involved.

For this and other reasons, consequences are sometimes directly included within the assurance case. While this closes a logical gap, it does not remove the decision maker's act of judgement regarding the merited degree of confidence.

Historically, the use of assurance has often included the assignment of risk-based levels that were used to match the functionality and uncertainty of the product or product element with its use as well as giving guidance to producers aimed at achieving the appropriate functionality and uncertainty. First, a structure of levels or categories might include terms or methods for answering a) through c). Examples for a) might include automobile kilometres per litre and crash ratings, fire retardant rating, safe or vault rating, and credit rating. For b) an example might be a reliability rating, and for c) the standard deviation yielded by reliability testing. Presumably, the degrees of inadequacy and uncertainty in meeting a claim are related to how the product was produced and to what evidence was collected and how. These (production and collection) could be guided by the assigned level – as with integrity levels (clause 10}. The nature of this guidance and the fidelity with which it was followed could give input to b) and possibly c). Possible consequences might be indicated by asset values or sensitivity levels (e.g. secret, top secret).

Levels have sometimes been used to give the answer to d) (e.g. level of economic activity or threat level) and the desired answer to f). Levels are also sometimes used to invert the process used to answer f) and solve for a) (required behaviour) as well as give corresponding guidance to ensure this answer to a) will be achieved (or possibly bettered) and that yields an acceptable (or tolerable) answer to b). This is done, for example, in doing analysis to go from level of risk to required integrity level.

When for lack of input or an invertible process this cannot be done, in the worst case the safest answer can be to give guidance aimed at achieving the best possible answers to a) and b) although this is likely to be overkill. Having done so, the question becomes, “Is the best possible good enough?” In practice, the theoretically best usually exceeds the best feasible and this in turn exceeds what is practicable and affordable.

Despite difficulties, these are fundamental questions that need to be addressed in a practical way. The assurance case with its claims, arguments, and evidence provides one such way.

10 ISO/IEC 15026 and integrity levels

10.1 Introduction

Integrity levels are suitable for use for certain levels of risk or to support an assurance case and impose criteria especially on the project, evidence collected, and product. The use of integrity levels has been useful in the past to users of ISO 15026:1998, and a revised and improved version integrated with the remainder of the revised ISO/IEC 15026 should be even more so. This clause outlines some of the issues and concepts underlying integrity levels and their use. It directly addresses some concerns of the users and potential users of ISO/IEC 15026 Part 3 on integrity levels. This clause is potentially useful to users of ISO 15026:1998 as well as the newer ISO/IEC 15026 Part 3.

Figure 6 — Product and Environment shows an overview of the mental model underlying much of Part 2. Consequences take their values from their affect on the interests (e.g. on their funds, health, equipment, or natural environment) of stakeholders. Such a consequence occurs when its precondition exists, and such preconditions occur as a result of conditions in the product's environment and the product's behaviour. A behaviour of the product occurs when its precondition exists within the product, and conditions in the product occur because of prior conditions, internal behaviours within the product some of which are the result of input from the environment. Likewise conditions in the environment can result from initiating events in the environment.

Tree of events and conditions can lead to the preconditions for consequences at their roots. Sequences of conditions and initiating or transition events are often not inevitable; rather the transitions can need to be treated as chance events and consideration given to their timing and possible variations in size or value.

A basic strategy is the prevention of preconditions for adverse consequences and therefore the initiating or transition events leading to them –and the preconditions for those. Another strategy is to limit the possible adverse consequences, e.g. their size, duration, and propagation.

Sub-clauses cover Defining integrity levels, Establishing integrity levels, Planning and performing using integrity levels, the key issues of Conditions and their initiating or transitioning events, Issues and limitations in using this international standard Part 3, Outcomes of use, and a brief Conclusion.

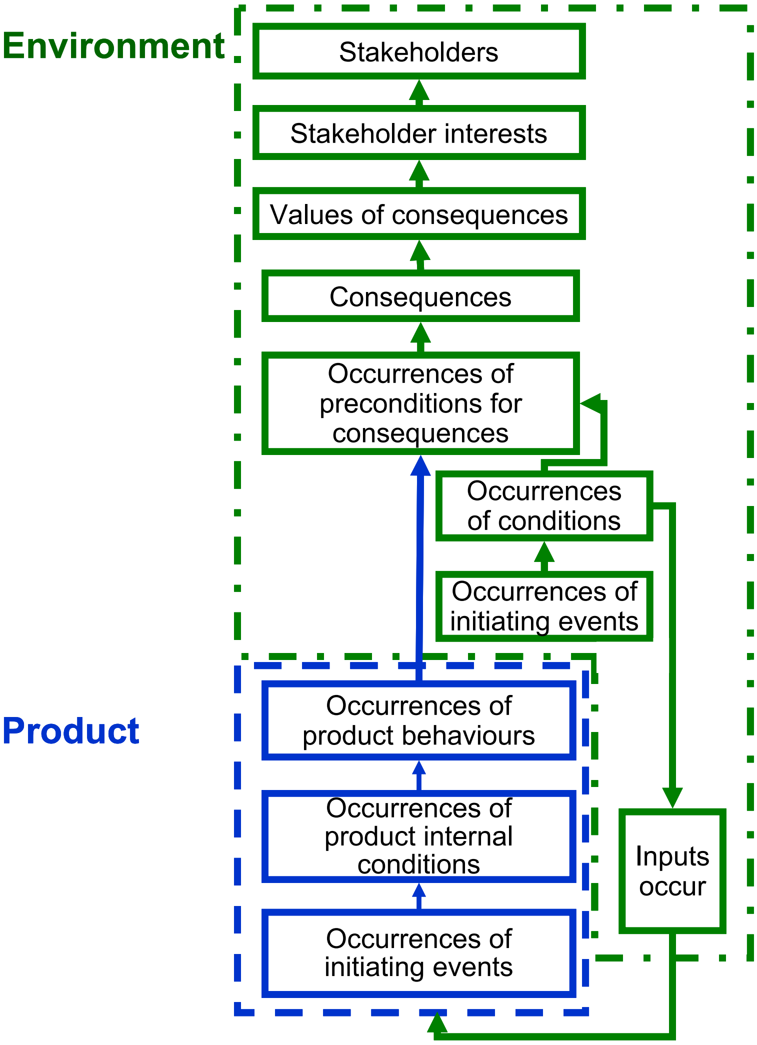

Environment
Stakeholders
Stakeholder interests
Values of consequences
Consequences
Occurrences of preconditions for consequences
Occurrences of conditions
Occurrences of initiating events
Product
Occurrences of product behaviours
Occurrences of product internal conditions
Occurrences of initiating events
Inputs occur

Figure 6 — Product and Environment

10.2 Defining integrity levels

The specifications associated with an integrity level shall document two kinds of related requirements called respectively, the "integrity level" and "integrity level requirements":

k) **"Integrity level" –** What integrity level **fulfils or claims:** namely that the product or element meets:

1) A certain level of a property such as risk, reliability, or occurrences of dangerous failures.

2) Within specified uncertainty limitations.

3) Under specified conditions.

l) **"Integrity level requirements" –** What it **imposes** on:

1) What is done and how, when, etc. – including on organization, processes, activities, tasks, methods, means and resources including personnel and tools, work environment, communication, management or coordination, record keeping, and other aspects of performance.

2) The product or element – including on associated material, services, and artefacts.

3) The evidence to be obtained possibly including limitations on its associated uncertainty.

The first, the requirements, those to fulfil, are called the "integrity level." For clarity and to better specify what is meant this Part 3 sometimes uses the term "integrity level's claim" for the requirements the integrity level fulfils.

Ultimately, evidence is central to designing and evaluating the design of the integrity levels. To be acceptably established, the designed levels must be shown to be such that:

4) If the required evidence exists and meets the criteria regarding it, it will be adequate to meet the required limitations on property values and uncertainties implied by the requirements it must fulfil from k).

5) Meeting the integrity level requirements imposed on evidence by an integrity level will show the meeting of all the integrity level's requirements.

The requirements related to integrity levels can differ for differing circumstances such as differing materials (e.g. software versus concrete) or construction or testing techniques (e.g. destructive versus non-destructive).

Whether thinking in terms of individual claims or assurance cases, the needed product integrity levels derive from the limitations on the values and uncertainties regarding the property values of the product itself and/or its elements (e.g. behaviours and contents). These properties of the product itself are under specified conditions that often include aspects of its environment. At least conceptually, these derive in turn from the limitations regarding consequences or from the top-level claim of assurance case. More particularly, for the assurance case, the evidence generated in conforming to an integrity level must adequately support the sub-claims supported by it.

10.3 Establishing integrity levels

10.3.1 Introduction

Risk analysis is used to establish the needed integrity level for the entire product. This risk analysis may or may not involve an assurance case. Once the integrity level is established for the entire product, integrity levels need to be established for what it depends upon including its internal elements.

10.3.2 Risk analysis

The approach to establishing the required integrity level for the entire product is risk analysis. The activities in risk analysis cannot generally succeed by only an outside in approach or an inside out approach (likewise top down or bottom up). The analysis needs to be approached from several directions. Nor can it succeed in a single attempt although a serious relatively early effort can be useful. Risk analysis is an ongoing and iterative process that must balance what is not yet knowable with what needs to be known and be prepared for learning and change.

Therefore all of the activities will normally benefit from the following steps:

a) Involvement of relevant expertise.

b) Examination of the environment of the product with resulting identifications.

c) Review of history relevant to the situation and similar situations.

d) Review of relevant standards and publications.

e) Serious concern for completeness and efforts to evaluate the degree of completeness of results.

f) Build an improving representation and understanding of the situation and keep records of information relevant to this even if it is not immediately needed.

To establish what the required product integrity levels are, one establishes the following:

a) The real-world requirements on consequences.

b) The limitations on values and associated uncertainties of claims regarding consequences.

c) What these consequence-related limitations imply are the required limitations on values and their associated uncertainties regarding claimed properties of the product itself and its elements.

d) The combination of design and properties of the implementation is required within the product to ensure meeting these limitations on values and uncertainties.

These need to be followed by establishing:

a) The integrity level requirements that if met will adequately assure the limitations on property values and their uncertainties required of the implementation of a product and its elements for it to be adequate in combination with the design.

b) What must be done and shown to establish within the limitations on uncertainty that the implementation of a product and its elements meets these integrity level requirements.

Consequently, integrity levels most directly derive from the severities of the property values and associated limitations on uncertainties regarding whether the product's implementation adequately meets its verification-related claims – those about the properties of the product itself. The integrity levels resulting from risk analysis are a translation of the values of consequences into the occurrences and timings of conditions or behaviours of the product. This translation is propagated to the integrity levels internal to the product and of its dependences as they are also in terms of occurrences and timings. Thus, integrity levels are a codification of what is needed to be done and shown for various ranges and severities of limitations on property values and their associated uncertainties.

NOTE Property values and their uncertainty values can vary in meaning and similarity. The uncertainty of the correctness of a given response might be reasonably thought of as the related reliability of the system. On the other hand, the uncertainty regarding whether the system's reliability is within a range, e.g. greater than a certain value or between two values, is distinctly different than the reliability of the system.

This international standard including its Part 3 does not cover risk analysis in detail. Many standards and guidelines exist that offer guidelines for risk analysis or can aid in identification of potential adverse consequences. IEC 61508 "Functional safety of electrical/electronic/programmable electronic safety-related systems" provides an approach to risk analysis. Another guide for risk analysis is IEC 300-3-9 "Risk Analysis of Technological Systems". As safety-specific terminology is used in IEC 300-3-9, the terms "hazard" and "harm" must be interpreted as "dangerous condition" and "adverse consequence" respectively. IEC 60300 Dependability management also provides guidance. Companion parts of the International Standard can aid particularly ISO/IEC 15026-1's annex on phenomena can help with identification of dangers and ISO/IEC 15026-2 Assurance case with analysis and reasoning as well as specification of claims.

Other specialized standards include ISO 13849 on machinery, ISO 14620 on space systems, ISO 19706 on fire, ISO/TS 25238 on health informatics, ISO/IEC FDIS 27005 on information security, and UK CAP 760 on air traffic and airports. Also of possible interest are the more general risk management standards ISO/IEC 16085 and ISO/IEC 15939. Some are more comprehensive than others, but subsets of these can be helpful in each of the steps in risk analysis and provide useful resources related to risk analysis and evaluation.

10.3.3 Element integrity levels

Once, with or without an assurance case, risk analysis has established an integrity level – possibly varying among interfaces – for the entire product, integrity levels need to be established for what it depends on. This can be complicated by not being free to assign integrity levels to be achieved for dependences on existing external elements or existing internal elements being reused. Thus, given such constraints and the integrity levels required of the product's behaviour at its various external interfaces, required integrity levels need to be assigned elements whose behaviours are depended upon including internal elements.

An element needs to be assigned an integrity level at least as high as that required by any of its uses. The integrity level required by a use depends on the integrity level required of the element using it and its role within the design of this using element.

Redundancy, diversity, and separation or isolation can affect this level. If failure of an element can only result in a dangerous condition in combination with other elements being in a particular state, then possibly one can assign a less stringent limitation to the element's failure occurrences than otherwise assigned. If an element offers one among several opportunities offered by different elements to result in a dangerous condition, then its integrity level might possibly need to be higher than that required of their combined behaviour.

While often the same as that of a using element, the property (possibly including multiple primitive properties) required can be affected by its role within its using element's design.

Thus, a product element is assigned the highest integrity level derived from:

a) Required integrity levels of product interfaces it provides.

b) Integrity levels of using elements and its place in the design of each using element.

10.4 Planning and performing

After establishing and assigning integrity levels with their integrity level requirements, one needs to plan and perform consistent with them – doing the following:

a) Plan to perform what is required to meet and show product has (or had or will have) integrity levels.

b) Perform what is required to meet and show product has (or had or will have) integrity levels including obtaining evidence to show this.

And finally, one uses what has been shown:

c) Review, gain approval, and communicate to create needed corrective action, confidence and/or quality of decisions.

Plans derive from real-world realities and are driven in part by integrity-level-requirements-related evidence concerns including obtaining it and ensuring its required values and quality.

10.5 Conditions and their initiating or transitioning events

Outside the product, much of the reasoning is or can be based on concern for conditions that could lead to adverse consequences and their initiating events or preconditions. Likewise, inside the product reasoning is based on conditions that can lead to dangerous product behaviours and the initiating events or preconditions for these conditions.

Figure 7 shows how these concerns interact. The dangerousness of product behaviours can differ by the conditions of its environment. As shown by the arrows above the line, these behaviours and conditions often need combining during analysis to establish whether adverse consequences will result or not. The actual conditions of its environment might or might not be known within the product depending on its sensors or inputs and their processing.

Likewise the product or its designers might or might not be cognizant of all the initiating events for a condition within the environment. Thus, dangerous conditions can need to be dealt with even though not all of their possible initiating events are known or recognizable.

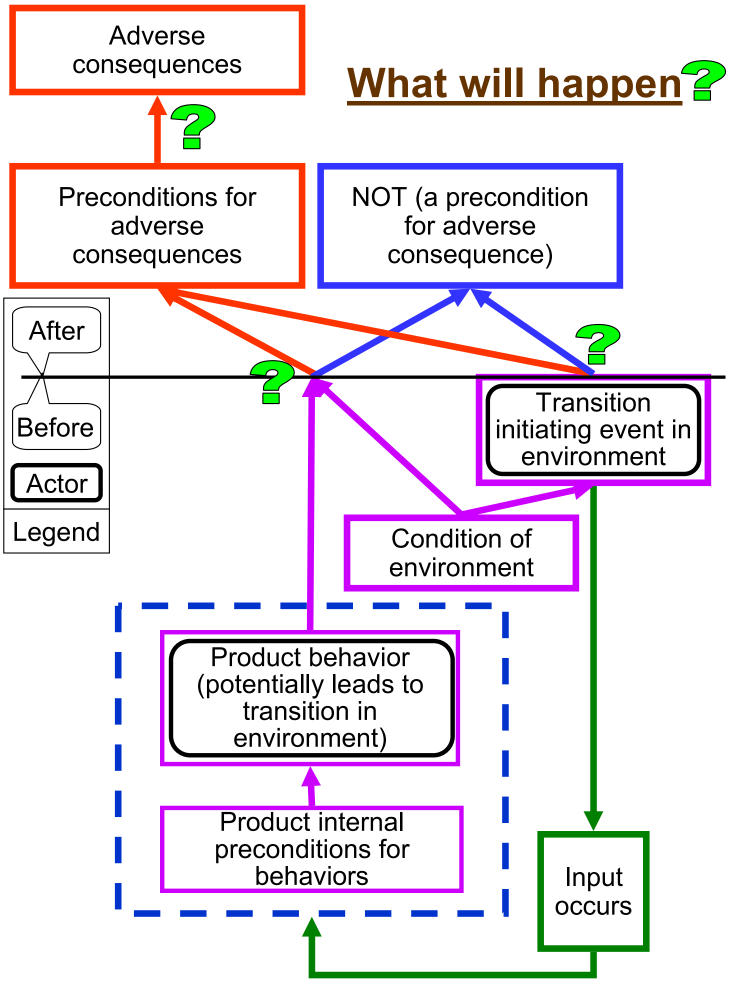

Adverse consequences
What will happen?
?
Preconditions for adverse consequences
NOT (a precondition for adverse consequence)
After
Before
Actor
Legend
?
?
Transition initiating event in environment
Condition of environment
Product behavior (potentially leads to transition in environment)
Product internal preconditions for behaviors
Input occurs

Figure 7 — Two actors cause transitions

10.6 Issues

For the approach to integrity levels presented herein to be most effective several issues regarding the products specification and analysability need to be addressed first. For example, one question that needs to be resolved is, "Do product behaviours exist that can lead to adverse consequences but are not forbidden by the specification or can dangerous conditions exist within the product that are not categorized as errors by the design specification?'.

Some portions of Part 3 presume the analysability of the product and complete knowledge of its relevant relationships with its environment as well as all product behaviours leading to dangerous conditions being failures to meet specifications – unless they are deliberately intended to be allowed. In addition, Part 3 often presumes for purposes of analyses that dangerous conditions have identifiable initiating events or causes that can be used during analyses. Users of this international standard Part 3 need to maintain awareness of the limitations resulting from these presumptions.

The portion of this international standard Part 3 covering the establishment of integrity levels generally presumes no such behaviours exist but warns they might. Particularly for products other than analysable ones and situations where the relationship between product behaviours and adverse consequences is not firmly established, uncertainty can exist concerning the existence of such behaviours, possibly substantial uncertainty.

In complex socio-technical systems, explanations of mishaps or claim violations cannot be limited to "component" failures. Adverse consequences can result from normal behaviour variability and unintended or unanticipated interactions [59] [60].

Thus, regardless of how they arise, dangerous conditions and adverse consequences are subjects for mitigation.

Complexity and lack of understanding or predictability of the product and/or relevant environment can create a situation where the approach provided by this standard Part 3 promises less than in – possibly simpler – situations with predictability and analysability. Nevertheless, this international standard Part 3 is useful and includes improvement over ISO/IEC 15026:1998 as well as explicitly covering issues at best tacit in many integrity-level-related documents. In addition to explicitly recognizing its shortcomings and limitations on applicability, it covers such central areas as defining individual and sets of integrity levels and their integrity level requirements; customized risk criterions; and dealing with dependencies in general not just internal elements, variations among risks at different product interfaces, and the lack of certain kinds of hard to know knowledge. Finally, it provides a generic set of requirements, guidance, and recommendations useful to developers of more specialized standards or other governing documents.

10.7 Outcomes

In using integrity levels, the following kinds of outcomes occur:

a) Requirements established for each integrity level because of its role in assurance – particularly in any assurance cases.

b) Specification established for the requirements (criteria) on a product or product element whenever it is assigned a particular integrity level.

c) Decision factors and process established for deciding assignments of integrity levels.

d) Assignment of integrity level to each product and/or product element (or portion).

e) Showing achievement of assigned integrity levels (that is, meeting the assigned level's criteria).

These outcomes, of course, imply activities exist to:

f) Plan for achieving them.

g) Achieve them.

h) Ensure related activities can be and are done adequately.

i) Document related plans, performance, inputs, and outputs including results, evaluations, and related approvals.

j) Obtain related agreements or approvals.

Part 3 emphasizes obtaining agreements among authorities (e.g. design authority and integrity assurance authority) and possibly approvals by authorities (e.g. integrity assurance authority).

10.8 Conclusion

Integrity level requirements reflect what is required to achieve and show the product or product element has (or had or will have) the properties claimed by its integrity level. A product's integrity level states what would be an adequate in terms of properties of the entire product - possibly differing at different external interfaces. Thus, showing it has a basic part in showing the meeting of larger claims involving the product and its environment including consequences – desirable or undesirable. If such larger claims are not made, then achieving and showing element integrity levels supplies a basic part of showing the top-level claim regarding the product itself.

In practice, integrity levels are often discussed in terms emphasizing the evidence needed to meet the integrity level requirements and thereby provide evidence for the arguments supporting claims regarding properties of the product itself. However, the quality of the arguments justifying meeting integrity level requirements as showing the achievement of its related integrity level is also important including their affects on uncertainties. Argument and evidence (as well as assumption) related uncertainties are a central concern and part of establishing integrity levels requirements.

In practice, obtaining agreements or approvals is important parts of ensuring integrity-level-related requirements are met. The International Standard 15026 Part 3 uses such terms as independent approval authority, design authority, and integrity assurance authority for particular roles.

11 ISO/IEC 15026 and the life cycle

11.1 Introduction

The affects of using parts of ISO/IEC 15026 can range from limited changes in an organization and its life cycle processes to sweeping affects. Widespread but not necessarily large changes are likely when introducing the full integration an assurance case throughout the life cycle.

An assurance case is seldom created without affecting the product’s life cycle. Figure 8 — Life cycle processes — provides an overview of the life cycle processes of both ISO/IEC 15288 and ISO/IEC 12207. In overall terms, it shows that an organization conducts projects in order to satisfy its goals. Its projects, at least the projects that are relevant to the cited standards, deal with systems. Each box depicts a process and those processes are classified as relevant to an organization, its projects, or its systems. The project-enabling processes are executed by an organization to support its projects. The organization also executes agreement processes in order to do business with other organizations. Each project is managed by the project management processes, with appropriate support from the project support processes.

While ISO/IEC 15288 and ISO/IEC 12207 are used as a baseline for discussion in this clause and receive special treatment in this international standard Part 4, their use is not required for conformance for conformance to Part 4. Any life cycle process and its definition that meet the requirements of Part 4 for life cycler processes may be used.

The activities of a project that deal with systems are depicted as the technical processes. So far, all of these processes are described in both ISO/IEC 15288 and ISO/IEC 12207 although the processes in ISO/IEC 12207 are specialized to and, in some cases, have different names reflecting that specialization. ISO/IEC 12207 contains additional processes that are unique to it. They are depicted as implementation processes, support processes, and reuse processes. ISO/IEC DTR 24748-1 Systems and software engineering — Guide for life cycle management [124] can aid in better understanding and performing the life cycle, and it provides a good supplement to this clause.

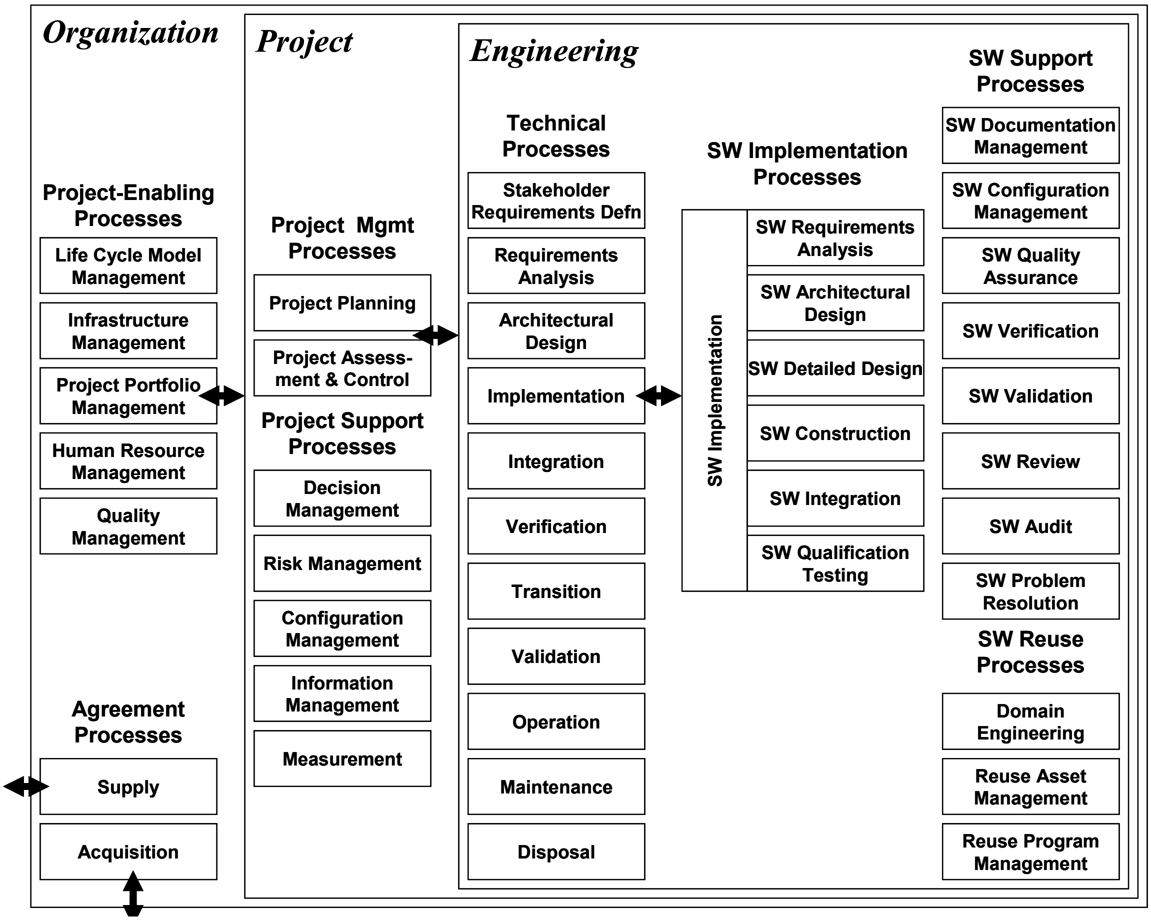


Figure 8 — Life cycle processes

This International Standard provides additional requirements and/or guidance beyond ISO/IEC 15288 and ISO/IEC 12207 potentially relevant to all the processes shown in Figure 8 — Life cycle processes.

ISO/IEC 15026 is suitable for use as part of an acquisition or supply agreement. The project-enabling processes depicted in the organization part of the diagram can have substantial affects on assurance- and assurance-case-related activities and artefacts as well as the product, but generally these are less direct than project-related ones. Thus, the Product Technical Processes are the ones of most central concern followed by the project support and implementation processes. Major sub-clauses cover Product technical processes, Post-Development, Organization processes including Acquisition and supply, and a brief Conclusion.

11.2 Product technical processes

11.2.1 Introduction

Generally, engineering disciplines have knowledge of (1) the “right ways” to engineer and (2) pitfalls or weaknesses – this is true for both the engineers' activities and their products. Therefore, ISO/IEC 15026 is concerned with both of these. Despite its apparent orientation towards undesirable events, conditions, and affects, it reflects that both accomplishing an adequate product and being sure that the product is adequate involve these two aspects – engineering from the perspectives of positive methods and avoiding negative pitfalls. Life cycle processes including activities, tasks, and the methods need to reflect this as well as showing adequate achievement via an assurance case and possibly though meeting integrity level requirements.

This sub-clause uses the structure of the Technical Processes in ISO/IEC 15288 to briefly mention a number of aspects and factors that tend to be particularly relevant to users of ISO/IEC 15026. Unless taken into account from the beginning, many of the requirements and claims of interest are difficult to achieve – that is, retrofitting them is difficult or impractical. Likewise, an assurance case is best if used from the beginning and it influences every activity – including planning, conducting, managing, and evaluating – as well as the product. This includes activities that directly deal with the assurance case.

11.2.2 Stakeholder requirements definition

Existing and newly gathered information and interactions with stakeholders result in a list of needs and preferences related to the product, its property of interest, and associated limitations on uncertainty as well as providing an understanding of the situation including identifying constraints – e.g. relevant laws, regulations, standards, policies, guidelines, interfaces, and compliance, contractual, evaluation, certification, or approval requirements. Organizations may have existing enterprise architectures, including security elements and frameworks that unify the means of complying to governing directives.

Stakeholders have needs and requirements not only for benefits but also for limiting adverse consequences and uncertainties. These are to stakeholders' interests and can relate, at least in part, directly to the particular assets and entities. The limitations on uncertainties may be needed to support the adequacy of a variety of stakeholder decisions including ones regarding consequences and risks.

Unfortunately, the normal situation for systems and software includes maliciousness and often an insecure environment. This adds to the non-malicious problems and makes concern for dangers and their sources and affects particularly important and relevant in circumstances where ISO/IEC15026 is most likely to be used. As mentioned earlier, ultimately these concerns are driven by potential real-world consequences associated with the product throughout its life cycle. Usually, this is accompanied by concern for related events, conditions, benefits, losses, and expenses.

Products are produced and utilized to meet needs, not just to avoid problems and dangers that can arise in relationship to them. Typically, many kinds of stakeholder needs are relevant to the product and to activities involving the product although these can vary over the life cycle. Table 6 lists some of the common kinds of subjects that can involve stakeholder needs particularly when safety and/or security are concerns.

The environment plays a key role in shaping needs and solutions as well as in their assurance. This includes systems that may incorporate the product, the dangerousness of the environment, ways in which the product may be accessed, entities that will have or seek to have access (or deny access) and many other aspects.

Table 6 — Information about sources of danger

• Causes, control, and motivations	• Duration or persistence, frequency, and timing
• Links and relationships	• Capacity to, and causes and chance of change
• Propensities and intentions	• Limitations and dependencies
• Capabilities and resources	• Possible uses, conditions, and roles
• Violations, damage, and losses	• Methods and approaches
• Gains	• Warnings

NOTE Natural events can be the source of many dangers. [[35], p. 11] lists kinds of natural disasters occurring in the years 1900-2002.

Of course, initial stakeholder requests and stated needs might or might not remain to become part of the agreed-upon set of needs and solutions, and therefore might or might not later be met by the specifications. Some of what is involved in the evolutionary progression that arrives at specification is mentioned in the next sub-clause.

11.2.3 Requirements analysis

Some of the more relevant kinds of analyses are analysis and resolution of conflicts among stakeholders and their needs, risk analysis, feasibility analysis, and tradeoff analysis. Feasibility analysis may consider technical, economic, human wellbeing, legal and regulatory, marketing, organizational, social, political, environmentalism-oriented, and mission-oriented feasibility as well as other aspects of feasibility particularly those of concern to key stakeholders or decision makers. All of these can involve costs and benefits, and reflect possible sources of opportunities, difficulties, dangers and uncertainties as well as positive or adverse consequences.

Table 7 — Some kinds of needs

Decision making	Limitations on uncertainty
Consequences	Limitations on expenses and use of resources and

	time
Stakeholder interests and asset protection	Interface and environment requirements
Compliance	Trustworthiness and trust management
Usability	Reliability
Availability, tolerance, and survivability	Maintainability, sustainability, and evolvability
Deception	Assurability
Validatability, verifiability, and evaluatability	Certification and accreditation
Market success	

Particularly when adverse consequences or risks will be crucial in decisions needed for product success (e.g. for its purchase and use), the existence of a capability to ensure and adequately show results in this area that decision makers find acceptable (or at least tolerable) can be decisive for establishing the feasibility of the product and its success. This requirement for assurability is equivalent to the question of can an adequate assurance case be produced. Of course, if an adequate product is not produced, this would be difficult or impossible so both issues – producing and showing – are important.

Feasibility involves looking forward thru the life cycle. Key efforts often include creating a concept for production, a concept of operations, and a concept for assurance. The concept for assurance needs to be shown to be feasible in the context of corresponding concepts for the life cycle. The requirements for the assurance case need to be established – at least high-level ones – identifying properties of interest.

Tradeoffs exist among properties including between safety or security, and efficiency, speed, and usability. Requirements analysis and design decisions may exacerbate or ease these tradeoffs. For example, innovative user interface design may ease security's impact on usability [34]. Business tradeoffs also exist (or are believed to exist) for producers between effort and time-to-market, and safety or security. Finally, producers want users and other developers to use features, especially new features, of a product but the likelihood of this is reduced if these are shipped "turned off" because of safety, security, or other concerns with adverse consequences.

The decisions regarding the extent of the assurance case and resulting effort have essentially the same bases as other similar decisions about risk management. ISO TR 15443-3 *Information technology — Security techniques — A framework for IT security assurance – Part 3: Analysis of assurance methods* addresses the issue of how to decide on the extent and nature of the appropriate assurance case at substantial length. It covers the goals, dimensions, elements, and structure of such decisions.

Projects with quality, systematic, and in-depth product-related risk management might see little change required to use an assurance case and possibly net benefits. Using an assurance case with limited additional artefact(s) to record relationships among parts of mainly existing artefacts is a possibility. Furthermore, the question might not be how much is risk management is warranted, but doing what is warranted in a more explicit, systematic, reviewable, auditable, and manageable way. Any change – even to being more systematic and thoughtful – usually requires added initial effort. Excepting this initial learning and organizational change – using an assurance case appropriately might very well not cause unwarranted expense and aid in better decisions on where to put effort for greatest risk reduction.

In practice, the processes for requirements and the process for architectural design overlap (as well as sometimes overlapping with research and implementation). Thus, the actual work and decisions regarding uncertainty, consequences, and assurance case(s) may involve widespread input from and influence on all three. Among these are decisions regarding the specification of top-level assurance case claim(s) and an approach to assurance.

11.2.4 Architectural design

The requirements for and of the assurance case are a driver of process- and product-related decisions. Thus, decisions about the assurance case can drive decisions regarding the product. The rationale for the product's design needs to justify that it will meet assurance case claim(s) and that this can be adequately shown – preferably, that it is adequately analyzable regarding the property(ies) involved. For example, for availability of a computer-based service, this might include its concurrency interactions being simple enough to allow (automated) analysis for livelock and deadlock. Thus, one can think of the operational product and the assurance case as being co-designed. Regardless of the area be it structure, flows, materials, electromagnetism, computing, manufacturing, or other, the design needs to be adequately analyzable, reviewable, and its implementation dynamically testable. In addition, simulation methods, test facilities for models such as wind tunnels and water tanks, and the use of CAD-CAM make dynamic testing of designs increasing powerful. These are all sources of evidence for the assurance case.

The design may be created in a way to “guarantee” the claim’s relevant aspects using justified assumptions – say ones regarding sub-claims related to its implementation that presumably are later verified. Nevertheless, reviews of every human-readable artefact normally need to be performed. In addition, review and analysis would be expected at the very least to verify that the process was correctly followed and no known pitfalls or known kinds of weaknesses resulted.

Design tradeoffs usually exist, for example, when considering including multiple safety controls or defence in depth a tradeoff can exist between using the same amount of resource to construct one strong control or defence versus many that are each weaker. In addition, undue inflexibility regarding a claim (or claims) can result in design problems.

Usually, the assurance case is easier to create and understand if portions of it are based on the design’s structure and rationale as well as its implementation’s agreement with it.

Four sorts of assurance issues that confront the designers of systems with properties to be assured are:

- The system might do something that it should not or when it should not thereby allowing, facilitating, contributing to, or causing undesired events, conditions, or consequences to manifest (i.e. error of commission).

- The system might fail to do something that it should do at the time it should, failing to prevent undesired events, conditions, or consequences from manifesting themselves (i.e. error of omission).

- Whether the system is intended to prevent them or not, the system might include capabilities intended to mitigate or minimize undesired consequences (these are subject to points 1 and 2). Such capabilities can affect predictions concerning consequences.

- The system might be such that adequately low uncertainties and/or consequences cannot be achieved or cannot be shown.

Positive measures to eliminate, prevent, avoid, limit exposure, or tolerate potentially adverse events, conditions, or consequences have all the advantages of prevention over cure.

Designers need to simultaneously consider the design of the product for (1) what the product will do and (2) adequately showing that the design will result in it doing what it should and have acceptable (or at least tolerable) consequences. In addition, consideration must be given to the feasibility of creating and transitioning the product so that the product as built and its consequences (1) will be and (2) can be shown to be acceptable (or at least tolerable) over its lifespan. Both of the pairs of 1 and 2, can have uncertainty regarding them, but this must be kept acceptable (or at least tolerable).

NOTE When safety and security are concerns, a few of the possible measures/countermeasures include limiting the paths that sources of danger might use to affect the system (e.g. reduce attack surface to reduce what information and services an attacker has access to), limiting the portion of the system that must be trusted and related dependencies on its environment, limiting the opportunities for and incidence of vulnerabilities and weakness in relevant elements (e.g., reducing the size and complexity of elements, static analysis, peer reviews, and/or random testing), avoiding states or preconditions allowing, facilitating, causing, or contributing to violations of claims or adverse consequences (e.g. keeping temperatures within allowable limits), and limiting the potential consequences of a violation (e.g., through isolation, limited privilege, mutually suspicious components, damage confinement, quick recovery, or an active countermeasure that detects and counters an attack's effects).

At the completion of product design, the design of the assurance case and its planned arguments and evidence should be equally complete encompassing all that is known about the product including the plans for related activities (e.g. implementation, integration, verification and validation, transition, and maintenance) including known and planned details as well as establishing constraints on future decisions needed for the assurance case's feasibility.

11.2.5 Implementation

Implementation activities and artefacts often span multiple levels of abstraction. Each level of abstraction needs to be shown to be consistent with the assurance case claim possibly by showing its agreement with the next higher level artefact. Such arguments across levels of abstraction are common and generally necessary. Regardless of the area – computer hardware or software, communications, structures, flows, materials, electromagnetism, medicine, transportation, so on – the implementation of such arguments means the implementation needs to be adequately analyzable, reviewable, and appropriately dynamically testable including intermediate descriptions, rationales, and products as well as any manufacturing.

Automatic generation from specifications or designs of software, instructions for manufacturing (e.g. machine tools), or other implementation artefacts including the product causes the means of generation to become a source of uncertainty. The same is true of most tools or aids. This means they become concerns of the assurance case. Indeed, any property or aspect affecting or reflecting their suitability and trustworthiness (e.g. ease of integration and use, throughput, correctness and reliability, accuracy, security, support, and availability) can become an issue.

11.2.6 Integration

The use of effort and care in integration and the placement of efforts to verify proper integration of product elements to become the product are usually risk based and should reflect the needs of the assurance case. From the perspective of an assurance case, the more serious uncertainties or risks are the ones that could most affect the achievement of the top-level claim and the showing of that achievement. This can affect allocation of effort.

11.2.7 Verification and validation

A key driver of the verification and validation efforts is the assurance case's specific needs for evidence. These depend on what properties and aspects the assurance case's top-level claim encompasses and the needs to verify or validate sub-claims reflecting their importance or criticality to the argument. The assurance case can be the major or just a minor driver of the overall verification and validation effort. In any case it calls for purposeful tasks with the objectives of meeting specific needs for evidence. Some examples of tasks aimed at verifying or validating sub-claims might be:

- Provide evidence showing that top-level claim agrees with needs, requirements, or property-oriented policy.
- Develop verification and test plans for the property of interest.
- Develop or acquire the product ensuring the top-level claim and its related properties such as to:
    - Facilitate the creation and structuring of arguments and sub-claims within the assurance case and aiding in identifying appropriate and sensible assumptions.
    - Provide needed evidence (adequately covering behaviours, conditions, sources of uncertainties and risks) for the assurance case's arguments and sub-claims require.
- Assure design provides for meeting top-level claim and supports creation of arguments and successful analysis, conduct of argumentation, and creation and/or collection of evidence.
- Assure product contains only items called for in design.
- Assure implementation is consistent with design and supports creation of arguments and successful analysis, conduct of argumentation, and creation and/or collection of evidence.
- Assure that product is free of critical weaknesses or vulnerabilities, corresponds to design, and claims as well as contains only items called for in design.
- Perform property-oriented testing including testing the property and property-related functionality.
- Provide an analysis of insidious possibilities (e.g. rare but catastrophic events, covert channels).
- Perform ongoing monitoring of opportunities and dangers, needs, and environment to verify the continued validity of and improve evidence and assumptions.
- Verify changes performed such as to maintain conformance to – possibly revised – top-level claim while continuing to agree with and show support for complete assurance case as revised (if needed).

- Verify transition conforms to requirements (e.g. conditionality and assumptions) of assurance case – e.g., correct deployment and installation.

- Verify that operation, use, disposal, etc. are being performed in a manner that conforms to the requirements of the assurance case.

Clearly verification and validation have a strong interconnection with the assurance case and the evidence it requires. As has been covered, this relationship exists throughout the relevant portions of the life cycle including during the duration of applicability of the top-level claim and in regard to all relevant ISO/IEC 15288 processes including organizational and technical processes.

11.3 Post-Development

Post development processes or concerns include training, deployment, monitoring, maintenance, transfers of control (legitimate and illegitimate), operation and use, retirement, disposal and/or other activities involving the product or its environment that are necessary to ensure the product's assurance case claims after its development including during the duration claimed for its top-level claim.

ISO/IEC 15288 includes a transition process, and an ISO/IEC 15026 Annex covers many possible post-development concerns in the context of planning for them.

The process and period of operation and maintenance are often the main topic of the assurance case's top-level claim. However, transition can have different relevant problems and concerns. For transition emphasis is often placed on adequate initial training. However, the following list mentions a few less obvious transition related activities where ensuring the meeting of the requirements imposed by the assurance case can require tailored arrangements or procedures while they are occur:

a) Obtaining needed changes in infrastructure or interfaces in environment particularly if product operation of begins before some of them are ready.

b) Beginning operation with a beta version, initial or partial operational capabilities, or with only a portion of the product and switching over to full product operation.

c) Start-up of backup and initial parallel operation with these backups.

d) Parallel operation with replaced product and cross verification.

e) Possible transition to and use of fallback modes possibly including fallback to replaced product and including concern for differing dependencies on product's environment or arrangements for backup.

During the duration of applicability of the top-level claim, the assurance case needs to be maintained along with the product. This need can start earlier than the start of applicability if the assurance case is completed or approved earlier or if the completed or approved assurance case or corresponding product is nevertheless inadequate or out of date. This includes the assurance case or its evidence being for older versions of the product or it making assumptions about a future version or about the environment that do not become true.

The monitoring and collecting of relevant field data that could strengthen or weaken the assurance case is essential. As many techniques exist for collecting and recording data (e.g. [70]), the crucial decision is what data to collect, record, and analyse. Clearly, any testing or field data used as important evidence in the assurance case needs to be monitored. This includes, for example, Data that could show if claims in the assurance case were violated or relevant incidents (e.g. near misses). As in field collection of data for any purpose, the usual concerns exist including expense, resources, automation of collection, and data validity.

Transfers of control and the activities that follow related to the product can cause problems such as negating required assurance case conditions or assumptions. Such transfers of control can be legitimate such as sale or lease, transfer to a maintenance facility or storage, a change in governance, or seizure by law enforcement or confiscation by a government. Transfers can be illegitimate such as theft or capture of the product, or takeover of a facility by strikers or activists. In different situations, coverage by an assurance case might be appropriate related to any of these or to other kinds of transfers of control.

Retirement and disposal are not processes that are necessarily beyond the needed duration of applicability. Examples of well-known concerns are safe disposal of hazardous materials or ordinance and the need to remove or destroy sensitive data before the retiring or disposal of the media on which copies of it reside.

11.4 Organization processes

11.4.1 Introduction

Organizational processes provide governance over projects, supply their environment and enable and support them often including human resource activities, and aid in a project's relationships with outside entities such as in marketing, acquisition, and supply. ISO/IEC 15026 is suitable for use as part of an acquisition or supply agreement process.

11.4.2 Project-enabling processes

The project-enabling processes depicted in the organization part of the diagram in Figure 8 — Life cycle processes can have substantial affects on assurance including the assurance case and related activities and artefacts as well as the product. Generally, these are less direct than project processes, but nevertheless often important.

One of the larger points of leverage (for good or bad) the larger organization can have on the project is through its effects on personnel; personnel competence, motivation, trustworthiness, communication skills, and ability to relate inside and outside the organization. Effects on these are major and obvious including on personnel available by location, recruitment, assignment and retention, for example by compensation, training, human resources function; and the governance, structure and culture of the organization.

Personnel affect the assurance case, and the assurance can place requirements for acceptable personnel and their characteristics, As an example of the latter, sub-claims can be made about the behaviour of operational personnel with these generating requirements for training.

To the extent the organization decides or controls the life cycle model and the available infrastructure, it can have other major effects on the project. Developmental and operational practices and infrastructure can result in differences in what the assurance case can show and on meeting the requirements imposed by the assurance case. Likewise, similar kinds of effects can occur related to maintenance, retirement, and disposal.

Thus, project enabling processes can have a number of effects within the assurance case, what it can show, and meeting the requirements imposed by it.

11.4.3 Acquisition and supply

The most straightforward and comprehensive treatment of the assurance case in the acquisition-supply process might appear to be the requiring by the acquirer and supplying by the supplier of a full and fully relevant assurance case. However, this sometimes is not feasible or acceptable to one or both parties. Nevertheless, acceptable agreements can be reached, for example to provide the acquirer with the information it needs for decision making while not disclosing business secrets or proprietary material.

ISO/IEC 15026 Part 2 covers assurance cases, but Part 3 covering integrity levels and Part 4 covering life cycle processes can also be utilized in acquisition. The Parts are consistent and can be used in pairs or all together.

In preparation for acquisition, an approach to assurance or an assurance strategy would appear in the feasibility study and be further elaborated to accompany any operational concept document. Requests for proposals can contain requirements for and on an assurance case including requiring conformance to parts of ISO/IEC 15026 and possibly other domain or location dependent standards. These exist for several sectors or purposes and for particular acquirers particularly government agencies. Some examples including related guidance documents are [146], [149], [150], [154], [155], [156], [164], [165], [181], [182], [196], [197], and [198].

The request for proposals (RFP) could provide information, requirements, and guidance regarding what top-level claim or claims are required including on properties and limitations on their values, durations, conditions, and the limitations on uncertainties and consequences that are required by the acquirer. Establishing an agreement on the (e.g. acceptable, tolerable, or allowable) limitations on uncertainties in the top-level claim is a vital step.

Note Legal proceedings have distinctions among burdens of proof – that is, the required degree of uncertainty. For example, in the U.S. levels exist such as preponderance of evidence, clear and convincing evidence, and beyond a reasonable doubt. Common language has these and many others terms regarding uncertainty. While terms such as inconceivable or wildly imaginative are not useful; and an example natural language set ranging from impossible to certain is: impossible, possibly possible, just possible, forlorn hope, surprising, unlikely, doubtful (on one point, a few points, many points), plausible, credible, even-odds, preponderance of evidence, probable, convincing evidence, highly likely, beyond a reasonable doubt, almost certainly certain, certain.

To form an agreement concerning uncertainty the parties need a shared conceptual understanding of its meaning in this situation ([203] p. 29-30). They must address the possibilities of having uncertainties about the issues of meaning and meaningfulness and the factors that underlie them. Evidence related issues include accuracy and the associated uncertainty, generalizability, inferences, and relevance to a particular property or argument. As [203] lists more specifically this includes the need to recognize that they can have uncertainties (and therefore possibly disagreements) concerning measurement, sampling, mathematical modelling, cause and effect, inferences, and categorization. Agreeing beforehand on standards of quality for the assurance case including evidence and incrementally agreeing or approving the assurance case designs and plans and its development and maintenance can increase the ease of dealing with this. However, affordable provisions may need to be made for a resolution process when disagreements persist either on substance or possibly on which party provides funding.

EXAMPLE To address possible uncertainty-related disagreements or doubts, the parties might agree upon third parties for mandatory expert consultation and if still needed, third parties to arbitrate deciding such issues such as what will be treated as correct, which party is correct, and is an objection from one party reasonable or in some cases not clearly unreasonable. Arbitration involves relying on established legal standards for burdens of proof with possibly differing standards for different kinds of issues.

A description of the proposed assurance case should appear in a proposal document during acquisition. This would allow the acquirer to evaluate the supplier on this point. Conformance to non-assurance-case standards can be required and used as a whole or by its parts as evidence in the assurance case.

An interesting related implication is that, in the situation where a user is relying only on claims made by the assurance case that supported the producer's decision to release the product, this requires that this supplier assurance case be at least as strong as what the user needs for a decision to use the product.

Acquirers also need to monitor and evaluate the assurance case and progress and risks regarding it during the period of supplier performance and when taking possession of or accepting the product. This evaluation can be more informative and reassuring than acceptance testing – which should always be done in any case and which might supply additional evidence for the next version of the assurance case. The assurance case can also be useful in obtaining required certifications and accreditations.

Relied upon off-the-shelf items must be considered in the assurance case. An obligation exists to adequately establish their relevant properties and to do so within required limitations on uncertainty. The amount of information concerning off-the-shelf items varies greatly. Usually, an off-the-shelf item is not accompanied by a relevant assurance case.

In the absence of adequate pre-existing evidence, it must be generated or collected. Trusted third parties can sometimes be used to avoid propriety or other access issues. Many techniques for obtaining evidence exist, and only a few are mentioned here. Among these are visual or other electromagnetic inspection techniques and evidence on prior field experience. Testing is a possibility if access can be gained to the item. Reports might exist in the literature, and analyses can sometimes be useful.

If one needs to generate evidence for software, one place to start is by using ISO/IEC 25051:2006 on testing off-the-shelf items. In high-consequence situations or others demanding low uncertainty, one generally needs to go beyond testing. If for software the source code is available, then static analyses for weakness, common vulnerabilities, and certain types of bugs can be done for many programming languages particularly commonly used languages, and code reviews and inspections are possibilities. Static analysis to explore structure or produce pictorial representations as well as some other static analysis techniques can sometime be done for binary code as well as source code.

The obligations and responsibilities regarding maintenance of the assurance case and the remainder of the product should be clear (and enforceable) in the agreement between the parties as well as obligations and responsibilities related to other relevant life cycle processes.

12 Conclusion

To be successful, a usually essential factor is actually having an adequate product. Evidence regarding the product and related information artefacts is crucial, but additional evidence can be quite useful. How probable or what is the chance that the approach taken will accomplish an adequate product? Showing the use of an approach likely to achieve the needed result and performing it well and equally importantly showing this can yields part of the grounds for confidence and better decision making.

Two basic areas essential to the best possible use of ISO/IEC 15026 have been addressed in some detail above. First, users need to have an adequate understanding of the concepts used in ISO/IEC15026 that are typically shared across the communities served that often have not previously been using the same terminology – or that in ISO/IEC 15026 . ISI/IEC 15026 uses these concepts many of which have been covered in this Technical Report. It also uses terminology designed to be reasonably readily understood by all its users.

Second, this Technical Report covers several concerns that arise when using ISO/IEC 15026. This Technical Report provides a basis for easier understanding and use of ISO/IEC 15026 as well as for understanding of the rationales behind it. This Technical Report can aid in learning, instruction and training, discussion, finding relevant references, and gaining intellectual mastery of the issues. Importantly, it is potentially useful in the development or revision of related standards and guides that intend to be consistent with ISO/IEC 15026 or elaborate on it.

NOTE The appreciation of the contents of this Technical Report may undergo change as work proceeds on the other parts of ISO/IEC 15026. A revision of this Technical Report ISO/IEC 15026-1 is expected to be later revised reflecting any such changes and published as International Standard ISO/IEC 15026-1.

Annex A
(Informative)

Frequently asked questions

A.1 A dozen frequently asked questions and their answers

These are questions that have been frequently asked. They are repeated here with answers and internal references to related clauses, sub-clauses, and annexes:

1) Parts 2-4 of the standard were not easy to read the first time. Why is this? The documents are designed to ease serious, repeated use. See sub-clause 8.3.

2) Some requirements that "should" be done do not fit my situation. Why is this? One needs a clear understanding of the usage of the terms "should" and "should preferably" to understand how these two terms respectively require justification for not following the requirement and do not impose this. Your situation is actually acceptable. See clause 3, Terms and definitions, and sub-clause 8.3.

3) What is an assurance case? An assurance case makes a claim regarding some property and provides arguments, evidence, and where appropriate assumptions to support it establishing a conclusion regarding it and the conclusions associated uncertainty. It can be used for any property. See clause 8.5.

4) Why uncertainty is not always expressed as a probability? Answer: Probability values may be extremely difficult or impossible to establish where an adversary deliberately goes against the probability estimates one makes. See sub-clause 9.3.3.2.

5) Parts 2-4 mention maliciousness in several places. As I am not concerned with security properties, why do I need to concern myself with maliciousness? Answer: Malicious actions can affect almost any property. See sub-clause 8.3.

6) I need to create an assurance case related to an existing product. Clearly, I cannot integrate assurance case development with product development. What do Parts 2-4 say about this? Answer: Part 2 supports assurance case separately from developing the product. Part 2 and 4 can help with the issue of integrating the two during the duration of applicability of the top-level claim. See sub-clauses 9.5 and 11.3.

7) How do the Parts of ISO/IEC 15026 related to each other? See sub-clause 8.4.

8) I am about to do an acquisition; what does ISO/IEC 15026 have to help me with this? Parts 2, 3, and 4 can be used in acquisitions. See sub-clause 11.4.3 for a limited discussion of this.

9) I am in the early planning stages; does ISO/IEC 15026 have anything to help me.

10) How does ISO/IEC 15026 related to the other standards I must or already use? Can I use them together? Answer: ISO/IEC 15026 is consistent with several standards and often still usable with others. See Annex C and sub-clause 8.4.

11) How can I approach the construction of a supporting argument? Answer: Many approaches are possible. A key issue is the meaningfulness of the argument. See sub-clause 9.3.

12) Why does Part 1 have a substantial annex on security and cover no other property in depth? Answer: For many initial users of ISO/IEC 15026, security has previously not been important in their area of concern until recently and they have only limited familiarity with it and its many aspects. See Annex E.

Annex B
(Informative)

Difficulties with terms and concepts

B.1 Introduction

Terms, concepts, and principles that relate to ISO/IEC 15026 span multiple disciplines, activities, roles and technologies. These terms and concepts have a long and varied history. Over time, many terms have come to be used within and across specialist communities such as systems, software, reliability, safety, maintainability, information security, software security, human factors and others well as in different application domains – but often in differing ways.

Debates occur about which meaning is the "right" one and competing dictionaries and ontologies. This is true both in general and among ISO publications. This is compounded by some terms having further variations within popular usage or in yet other professions. For example, psychology, law, and mathematics address issues that are also important here.

Without proper awareness and care, these differences among specialities or communities can cause confusion that hinders productive communication including, the users of this international standard. To avoid this confusion, this Technical Report recommends clear, unambiguous terminology even though sometimes this requires using two or three words, or even a phrase, where a particular sub-community may currently use one word. It also tries to help the reader obtain a clearer understanding of the underlying concepts, and to recognize subtleties such as references to what is true in a certain situation, what is known, and how well it is known. Also, readers possibly will be aided in discerning the assumptions implicit in various concepts and statements.

This Part 1 covers many concepts useful to users of ISO/IEC 15026 and preparers of conformant or compatible standards or other governing documents. These are concepts and principles that, while not all are always relevant everywhere, span multiple disciplines, activities, roles, properties, application domains, and technologies. It emphasizes concepts needed for understanding the area of software and systems assurance and, in particular, those central to the preparation and use of Parts 2-4 of the ISO/IEC 15026 International Standard. As part of this, several particularly relevant issues and rationales are discussed. In addition, it aids users of the other parts of ISO/IEC 15026 in related use and conformance.

This Part 1's scope reflects the scopes of the other initial parts of ISO/IEC 15026:

- 15026-2: Assurance case.
- 15026-3: System integrity levels.
- 15026-4: Assurance in the life cycle.

This technical report provides:

- Terms and concepts that span these three parts and their areas of concern.
- Coverage of issues that can aid in their use and conformance.

B.2 Variations in the usage of terms

Currently, the terms used for the concepts and properties covered in ISO/IEC 15026 vary across the communities of interest. To aid users of ISO/IEC 15026 in understanding the situation, this sub-clause covers some kinds of these variations. First, in particular, terms effective meanings tend to change depending on which of the following they are used in regard to:

- What is needed.

- What is specified.
- What happens or is done.
- What is actually true (e.g. about the software, system, or environment).
- What has been measured, observed, inferred, etc.
- The uncertainty in these measurements, estimates, conclusions, etc.
- The degree of confidence one has.
- The related decisions one makes.

The fourth, “What is actually true,” may be referred to but is seldom known exactly. Rather one has, “What has been measured, observed, inferred, etc.” with their uncertainties.

Other kinds of variations also exist including (1) what is conceivable, possible, or feasible; (2) entities, events, behaviours, or conditions; and (3) allow, facilitate, contribute to, cause, or affect or effects – as well as contextual issues.

NOTE 1 For example, outside this and consistent standards, the usage of "assurance" varies as different speakers (or writers) use it to refer to an entity, capability, condition, event, consequence (computing or real-world), physical or mental state, action, activity, or process.

NOTE 2 In this international standard "uncertainty" is used in a general way to mean lack of certainty. This usage allows the term “uncertainty” to be applied to anything. Different communities restrict the application of this term to limited usage, for example to predictions of future events, to physical measurements already made, or to unknowns. While this may be convenient within these communities, this International Standard crosses many communities.

Finally, the effective meaning of a statement (e.g. a specification) can also depends on the situational assumptions underlying it. Thus, the meaning of a statement often derives, at least in part, from the scope to which it is being applied.

B.3 Conclusion

Because of the lack of a common treatment of shared concerns, existing standards addressing different application areas and different topics or properties apply differing terminology and seemingly differing concepts when addressing common concepts. This phenomenon increases the costs to users who must concern themselves with more than a single important property. For example, to conform to International Standards on both safety and security, users may find that they need somewhat redundant risk assessment and management processes. On the other hand, this International Standard reflects and benefits from these more specialized standards and has as a purpose to provide a common underlying framework, terminology, concepts, and set of requirements related to assuring properties and to provide a basis for future standards or revisions of standards treating specific properties to benefit from, apply, specialize, and extend.

Annex C
(Informative)

ISO/IEC 15026 relationships to standards

C.1 Relationships

Because this International Standard is intended for application in a variety of contexts, the user may confront differences in terminology and concepts. It is not appropriate for this standard to introduce a broad variety of terms because they may conflict with terminology specific to the context of application. The essential concept introduced by this International Standard is the statement of *claims* in an *assurance case* and the support of those claims. Other terms are less important; nevertheless, it is important that the readers be provided with an understanding of those terms. Therefore, this International Standard uses the terminology and concepts consistent with ISO/IEC 12207:2007, ISO/IEC 15288:2007, and ISO/IEC 15289:2006.

This International Standard does not presume that it is applied in conjunction with ISO/IEC 12207:2007, ISO/IEC 15288:2007, or ISO/IEC 15289:2006. Those who have an alternative basis for their life cycle processes may also use this standard.

Part 1 of this International Standard provides background and information that could be useful in understanding and using this Part 2.

Part 4 of this International Standard provides tasks related to the assurance case that must be performed integrated within life cycle processes particularly for concurrent development and maintenance of the product and its assurance case. It provides general requirements and these requirements instantiated in the contexts of ISO/IEC 12207:2007 and ISO/IEC 15288:2007, the International Standards for software and system life cycle processes. Conformance to Parts 2 and 4 is separate.

This International Standard provides a *process view* for systems and software assurance. It provides a statement of purpose and a set of outcomes and requirements suitable for systems and software assurance.

NOTE 1 ISO/IEC 15288 and ISO/IEC 12207 characterize life cycle processes for systems and for software products, respectively. However, some situations exist, as in this International Standard, where a unified focus is needed for activities and tasks that are selected from disparate processes to provide visibility to a significant concept or thread that cuts across processes across the life cycle. ISO/IEC 25961 (or 42010:2006 if renumbered) provides a solution to this problem. That standard defines the term "view" as a "representation of a whole system from the perspective of a related set of concerns." If we view the set of processes such as those provided by ISO/IEC 15288 and ISO/IEC 12207 as a system, then we can use the term “process view” to denote a representation of the whole set of processes from the perspective of a related set of concerns.

NOTE 2 The concept of a process view was formulated and described in an annex of ISO/IEC 15288:2007. Like a process, the description of a process view includes a statement of purpose and outcomes. Unlike a process, the description of a process view does not include activities and tasks. Instead, the description includes guidance explaining how the outcomes can be achieved by employing the activities and tasks of the various processes in ISO/IEC 12207 and ISO/IEC 15288. This International Standard was developed upon the basis of the Specialty Engineering Process View provided as an example in ISO/IEC 15288. It, however, does not presume and does not imply any engineering speciality in assurance and is complete and is self-contained, not necessitating compliance with ISO/IEC 12207 or ISO/IEC 15288. The provisions regarding the assurance case and assurance planning are intended to be compatible with the provisions of ISO/IEC 15289:2006 for information items resulting from life cycle processes. However, the relevant provisions of this International Standard are self-contained; it is not necessary to also show conformance to ISO/IEC 15289.

NOTE 3 This standard benefits from much prior work both outside and inside ISO and IEC and within many fields. Much pioneering work has been done in the safety community. In one example elsewhere, the security community in ISO/IEC JTC 1/SC 27 has been working on the topic of assurance for many years concentrating on systems and software, while security is only one of many areas covered by ISO/IEC 15026, it benefits from SC 27's work. While security is only one of many areas where this International Standard Part 2 can be applied and where mention might be made of standards, two security-related examples to mention are ISO/IEC TR 15443, Information technology—Security techniques—A framework for IT security assurance that discusses the need for arguments and evidence in the context of information technology security [multiple parts], provides a security focus that is not limited to information technology and ISO/IEC 15408:2005, Information technology—Security techniques—Evaluation criteria for IT security [multiple parts], provides a particular form of an assurance case specialized to a specific form of claim. However, no dependency of this International Standard exists upon the use of any of them.

Users of this International Standard may require risk assessment and risk management and measurement processes that are more fully elaborated than the treatments provided in ISO/IEC 12207 and ISO/IEC 15288. Two International Standards, ISO/IEC 16085 and ISO/IEC 15939, are useful in this regard. However, users interested in assurance of some specific properties may decide to apply risk assessment and management and measurement standards that are specifically applicable to the relevant properties and products.

Some material regarding assurance-related planning and its supporting analyses has been adapted from IEEE Std 1228:1994. Because the relevant material is self-contained and fully explained in this International Standard, there is no dependency established upon the use of IEEE Std 1228.

The provisions of this International Standard are generally consistent with those of the ISO/IEC 25000 series of standards related to product quality, and aim to be generally consistent with the ISO/IEC 27000 series of standards related to information security management systems, the IEC 61508 multi-part standard on functional safety, and various standards of IEC TC 56 related to dependability. However, except as specifically cited, there is no dependency of this International Standard upon the use of those standards.

Many international – as well as industry and national – standards exist addressing the concerns of safety, security, reliability, maintainability, dependability, human factors, and other important topics, but, to date, no common treatment exists of the shared aspects of these concerns. ISO/IEC 15026 including this Part 2 addresses assurance in a common manner. The motivation is to provide a unified view spanning these many areas across the life cycle. This top-level standard may be applied in conjunction with other standards that address the specific concerns of the properties that are of interest. These other standards, depending upon their source and other factors, may not be completely harmonized with this International Standard, so their application may require that the user resolve perceived inconsistencies.

NOTE A discussion of the assurance case and three existing standards not explicitly calling for an assurance case appears in [10].

Several potentially relevant standards are listed in the Bibliography.

NOTE Industry and agency standards and guides explicitly about assurance cases include [146], [149], [150], [154], [155], [156], [164], [165], [181], [182], [196], [197], and [198].

C.2 More on relationships to life cycle process standards

Figure 9 depicts the relationship of the several standards related to the life cycle processes. At the bottom of

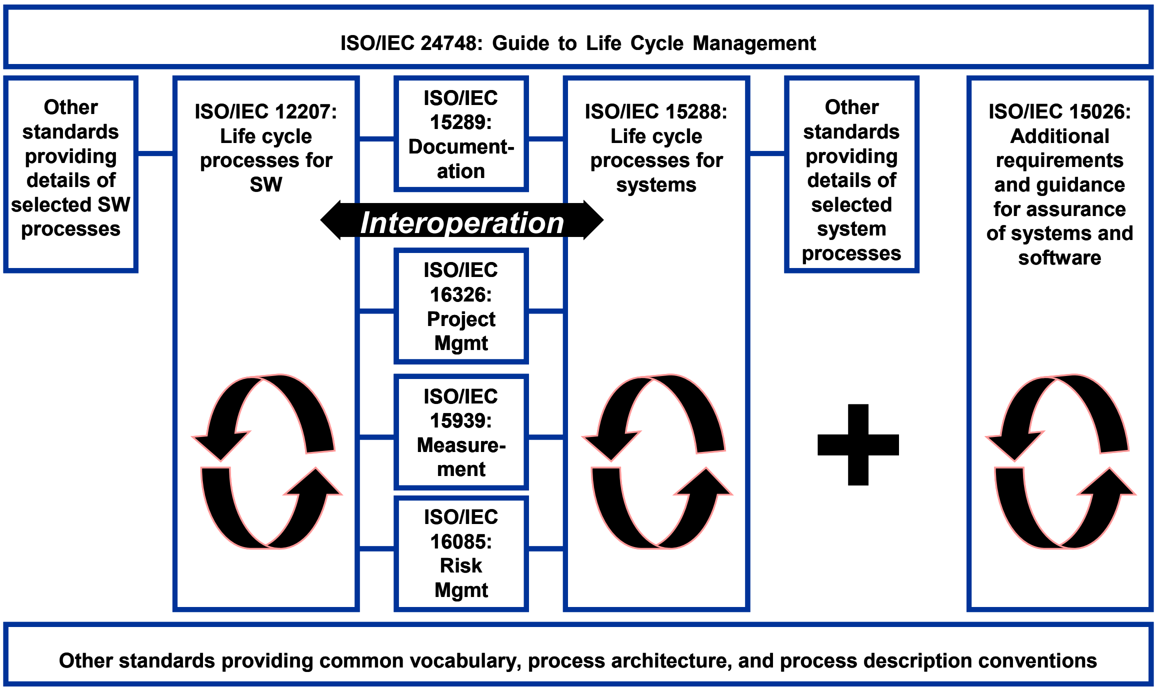

ISO/IEC 24748: Guide to Life Cycle Management
Other standards providing details of selected SW processes
ISO/IEC 12207: Life cycle processes for SW
ISO/IEC 15289: Document-ation
Interoperation
ISO/IEC 16326: Project Mgmt
ISO/IEC 15939: Measure-ment
ISO/IEC 16085: Risk Mgmt
ISO/IEC 15288: Life cycle processes for systems
Other standards providing details of selected system processes
ISO/IEC 15026: Additional requirements and guidance for assurance of systems and software
Other standards providing common vocabulary, process architecture, and process description conventions

Figure 9 — Some relationships among stadards

the diagram is a foundation of a number of standards that provide common vocabulary, architecture for processes, and a convention for describing those processes. The other depicted standards are built upon this foundation. ISO/IEC 15288 and ISO/IEC 12207 provide life cycle processes for systems and software respectively. They are intended to be interoperable, hence useful for systems with varying content. The two life cycle process standards are supported by four standards that provide additional requirements and guidance on shared issues: ISO/IEC 15289 for documentation resulting from the execution of life cycle processes; ISO/IEC 15939 for the measurement process; ISO/IEC 16085 for the risk management process; and ISO/IEC 16326 for the project management process. In addition there are other standards that provide additional requirements and guidance for selected processes. ISO/IEC 24748 is a guide describing how life cycle processes are organized to manage the entire life cycle of a system or software. This International Standard, ISO/IEC 15026, is intended to be compatible with these other standards.

The goals of assurance, the selection of claims to be assured, assurance-related planning, and the construction and maintenance of the assurance case have influences within all life cycle processes.

Annex D
(Informative)

Phenomena

D.1 Introduction

ISO/IEC 15026 can be used for positive issues and consequences. For example, some users are interested in readiness, performance or gain, but others are interested dangers, risks, or losses. To aid them, this annex contains several tables and lists with related sources and kinds of these as well as a list of online sources for information about dangers and lastly lists of the basic types of force and matter.

D.2 Sources and kinds

Phenomena can be thought of as associated with fundamental phenomena such as fundamental force and states of matter, or thought of in as related to pragmatic lists of phenomena often compiled from prior experience. This sub-clause offers tables with lists of both kinds.

D.2.1 Kinds and sources of phenomena and the locations of them and their consequences

Phenomena can have many causes, points, of origin, methods of occurrence, kinds and locations, and consequences. This table is not exhaustive but are useful. This list and the sources may be used in combination with standards and guidelines for domain areas.

Table 1: Some kinds and sources of phenomena	
Altitude (e.g. aircraft, mountain)	Kinetic (e.g. movement, (de)acceleration, explosion, vibration, sound)
Backup and recovery	Logistics, provisioning, and sustainment
Biological (e.g. agricultural (e.g. famine), medical (e.g. health care, epidemic))	Maintenance (e.g. preventive, repair, lack or incorrect)
Boundary crossing (osmosis, intrusion)	Materials (e.g. structural, semiconducting, hazardous)
Capacity limitations (e.g. bandwidth, mental overload)	Nuclear and radiological
Chemical	Observation and perception (range, acuity, surveillance)
Cyber (e.g. computer phenomena or cyber attack)	Physical (physics)
Earthquake	Physical degradation (e.g. ware)
Electrical (e.g. power supply, lighting)	Precipitation (e.g. hail, ice storm, snow, deluge, draught)
Electromagnetic (e.g. EMP)	Readiness (e.g. high state of, inadequate)
Emergency	Sensitivity and tolerance
Equipment (e.g. size, weight, function, safety)	Slide (e.g. landslide, avalanche)
Environment (e.g. systems, work, natural)	
Extraterrestrial sources (e.g. solar flares, meteors)	Strength (e.g. weakness)
Fire (e.g. equipment, structure, wild fire)	Stress
Flexibility (e.g. inflexibility)	Submerge (e.g. submarine, dip, bathe)

Flight (e.g. aircraft, parachute)	Surprise (e.g. lack of anticipation or warning, shock)
Float (e.g. watercraft, ship, person)	Temperature (e.g. operating range, extreme)
Flood including water damage	Testing (e.g. reliability, load, security, unrealistic testing)
Global warming or cooling	Training and practice
Health (e.g. fitness, injury, epidemic)	Use (e.g. amount, ease, wear)
Human behaviour (performance, error, (in)competence, speed, efficiency/productivity)	Volcano
Information (e.g. amount, accuracy, timeliness, lacking, wrong, corrupt)(See ISO/IEC 25012)	Wave/surge
Insect or rodent (e.g. pollination, infestation)	Wind

Table 2: Social causes and locations

Social Causes	
Activism (e.g. political, social, religious)	Recreation (e.g. prank)
Civil unrest	Revolution
Crime	Subversion
Diplomacy (e.g. trade agreements)	Terrorism
Espionage	Vandalism
Industrial competitiveness	War
Legal constraint, law enforcement and judiciary	

Possible locations for phenomena occurrence and gain or damage	
Business and trade	Raw material supply
Cyber	Space (e.g. disasters)
Environmental	Persons (physical, psychological, and relationships)
Industrial (e.g. accidents)	Information
	Internal to product
Infrastructure	
Agriculture and food	Search and rescue
Communications	Energy (e.g. electricity, petroleum)
Emergency and disaster response and recovery	External affairs (e.g. national)
Decontamination and cleanup services	Financial services
Emergency health care	Fire fighting
Emergency hazardous materials response	Information technology and services
Emergency management	Internal governance
Emergency and evacuation transportation	Housing and shelter
Emergency water and food supply	Natural resources
Insurance payouts	Hazardous materials services and response
Long-term community recovery and mitigation	Public health and medical services
Mass care, clothing, housing, and human services	Public safety and security
Mortuary services	Public works and engineering
Public health	Search and rescue
Public safety and security	Transportation
Restoration of electric power	Water supply

D.3 Information about dangers and weaknesses

Current information on vulnerabilities, weaknesses (which may be vulnerabilities), and the exploits that target them can be found in a number of sources, including books (in which the information may be better organized as an introduction to the subject, but will be less current), articles, vendors' and independent "alert" services, and databases. For examples, see the following:

- ACM Committee on Computers and Public Policy The Risks Digest: Forum on risks to the public in computers and related systems, - http://catless.ncl.ac.uk/risks.
- Canada: Transportation Safety Board - www.tsb.gc.ca.
- Common Attack Pattern Enumeration and Classification (CAPEC) http://capec.mitre.org.
- France : Bureau d'Enquêtes et d'Analyses pour la sécurité de l'Aviation Civile - http://www.bea-fr.org/.
- Germany: Bundesstelle für Flugunfalluntersuchung - www.bfu-web.de/.
- Internet Security Systems (ISS) X-Force Database - http://xforce.iss.net/xforce/search.php.
- MITRE Corporation Common Weaknesses Enumeration - http://cwe.mitre.org/.
- MITRE Corporation dictionary of Common Vulnerabilities and Exposures - http://www.cve.mitre.org/.
- NIST National Vulnerability Database - http://nvd.nist.gov/.
- Open Source Vulnerability Database - http://www.osvdb.org/.
- OWASP Top Ten - http://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project.
- Purdue University Center for Education and Research in Information Assurance and Security (CERIAS) Cooperative Vulnerability Database - https://cirdb.cerias.purdue.edu/coopvdb/public/.
- SANS Top Twenty - http://www.sans.org/top20/.
- Secunia Vulnerability and Virus Information - http://secunia.com/.
- SecurityFocus Vulnerabilities - http://www.securityfocus.com/vulnerabilities.
- UK: Air Accidents Investigation Branch, - www.aaib.dft.gov.uk/.
- US DoD's Joint Task Force-Global Network Operations (JTF-GNO) Information Assurance Vulnerability Alert (IAVA) program - http://www.cert.mil/.
- US: National Transportation Safety Board - www.ntsb.gov/.
- US-CERT Vulnerability Notes Database - http://www.kb.cert.org/vuls/.

The ACM's Risks Digest has over twenty years of information on individual incidents and offers a search capability. For software developers the most useful may be the Common Weaknesses Enumeration (CWE) [31] although the Common Attack Pattern Enumeration and Classification (CAPEC) is emerging its useful companion as well, and books such as [43], [57], [207] are available.

What is publicly known, however, may be less than is known to producers or researchers. Potential attackers may know exploits no one else knows. In addition, even after shipment some software vendors make significant efforts to discover vulnerabilities through internal efforts whose results are often not publicized.

D.4 Fundamental forces and states of matter

Below is a table listing fundamental forces and another states of matter, While starting to find and explore phenomena of interest from such fundamental phenomena is offend to far from the concrete phenomena of interest, they are occasionally useful. The next sub-clause has more pragmatic tables.

Table 3: Fundamental forces

Fundamental forces	**Range of force**
Strong interaction	10^{-15} meters
Electromagnetic	Infinite
Weak force	10^{-18} meters
Gravitational	Infinite

Table 4: States of matter

States of matter	**Changes**	
Solid	**Solid to liquid** Melting or fusion	**Liquid to solid** Freezing
Liquid	**Liquid to gas** Vaporization, boiling, evaporation	**Gas to liquid** Condensation
Gas	**Gas to solid** Solidification	**Solid to gas** Sublimation
Ionized Plasma		
Quark-gluon plasma		
Bose-Einstein condensate		
Fermionic condensate		
Other		

Annex E
(Informative)

Security

E.1 Introduction

NOTE Material in this Annex is excerpted or adapted from [174] with permission of the editor.

ISO/IEC 25010 and ISO/IEC 15026:1998 define security as, "The protection of system items from accidental or malicious access, use, modification, destruction, or disclosure."

ISO/IEC 25010 associates the following qualities with security: confidentiality, integrity, non-repudiation, accountability, authenticity, security compliance, and immunity (the degree to which the software product is resistant to attack), plus related survivability and safety. Another definition of security is, “All aspects related to defining, achieving, and maintaining confidentiality, integrity, availability, accountability, authenticity, and reliability.” (ISO/IEC13335-1). In addition, security-related usability can also be important to facilitate ease of system operation and use and to avoid unacceptable user inconvenience with this possibly resulting in users deliberately avoiding or bypassing security features.

“Security [of information] often requires the simultaneous existence of 1) availability for authorized actions only, 2) confidentiality, and 3) integrity with no improper meaning ‘unauthorized’ [changes].”[5] [[14], p. 13] Security is not as simple as this last seems. Neither confidentiality nor integrity can generally be achieved without entities being adequately identified (identity known or established to some level of uncertainty usually desired to be firmly established including being verified – authenticated). In addition, for the identified entity, only actions permitted to it are allowed or possible – usually through use of access control mechanisms, separation, or encryption.

As with many other properties, security properties are not concerns confined to the scope of computing and information resources. For example, real-world lawbreakers and consequences are also certainly relevant as are physical, personnel, operational, and communications security.

When attacks occur, the product may also be required to detect those attacks and alert users, continue service, confine damage, rapidly recover, and aid in diagnosis and repair to prevent future problems.

Sometimes properties including security properties are properties of the whole system. This means that these properties are determinable or observable only[6] in the context of how the multiple elements of the system interact among themselves, and how the system responds to stimuli from external entities (e.g., input from users). Thus, these properties are said to emerge or be “emergent” with system composition.

While necessary for safety or security in the majority of systems, the mere presence of safety or security functionality does not make a product safe or secure. To provide security, individual pieces of security functionality must be impossible to bypass, corrupt, or cause to fail, and the same is generally true for safety. Given accurate facts about its environment, these inabilities to corrupt, cause to fail, and bypass can emerge from the inherent properties of the product possibly by use of separation and isolation within the design. Software may attempt to achieve this, but dependencies including those on hardware and other system elements meaning these inabilities must ultimately (also) be achieved at the system (or higher) level.

Since for some properties such as security ones, the product is preserving what might be considered systems level properties and may protect many kinds of stakeholder interests and computing resources such as

[5] Another definition of security is, “All aspects related to defining, achieving, and maintaining confidentiality, integrity, availability, accountability, authenticity, and reliability.” (ISO/IEC13335-1).

[6] This does not mean that (1) similar or analogous properties may not exist at lower levels or that (2) one might not design a system so a guarantee can be derived from the behaviours of only a portion of a system and a lack of opportunities (to violate or help violate security) for the remainder.

hardware data, software, and running processes; and does so in a systems context; the distinction between a property being software- or system-level matters little. Thus, this annex generally avoids trying to label topics system versus software; rather the all-encompassing term product is used spanning systems, software, services, and possibly other products as well as their elements or constituents. On the other hand, system-subsystem relationships and differences in levels of abstraction are important and noted where required.

NOTE While exhibiting reliability, safety, and maintainability may not directly result in a security property, the last can contribute to keeping security "up to date." All may make it easier to show that product is secure.[7]

E.2 Kinds of security

Lists of kinds of security or security-related areas can like direct attention to potential areas of concern or motivation for security. On a wide scope, some speck of kinds of security using a variety of terms. These include:

- Operational security.
- Transportation security.
- Financial security.
- Personal security.
- Infrastructure security.
- Environment security.

Emphasizing computing and communications, ISO 7498-2 lists several kinds of security-related areas:

- *Administrative security*, e.g. controlling the importation of software, procedures for investigating security breaches, audit trail analysis.
- *Communications security safeguards*, e.g. authentication, access control, data confidentiality, data integrity, non-repudiation.
- *Computer security safeguards*, e.g. operating system and database system security facilities.
- *Emanations security*, e.g. radio frequency emanation controls (TEMPEST protection).
- *Physical security*, e.g. locks or other physical controls, equipment tamper-proofing.
- *Personnel security*, e.g. employee screening for sensitive posts, security training and awareness.
- *Media security*, e.g. protecting stored data, secure destruction of computer storage media, media scanning for viruses.
- *Life cycle controls*, e.g. trusted system design, implementation, evaluation and certification, programming standards and controls, documentation controls.

Other terms are also used related to computing and communications such as:

- Information and Communication Technology (ICT) security.
- Communications security.
- Data security.

[7] For further information on this topic, see *Security in the Software Lifecycle*, Section 3.1.3, "Other Desirable Properties of Software and Their Impact on Security".

- Application security.

- Information security.

- Network security.

Examples of abbreviations in use include COMPSEC, COMSEC, EMSEC, PHYSEC, ICTSEC, INFOSEC, and TRANSEC.

E.3 Security-related properties

E.3.1 Introduction

Beyond what has been covered so far, this section covers additional conceptual material that should be part of the knowledge of everyone involved or interested in security. ISO/IEC 25010 provides definitions for security-related qualities, More in-depth treatments of security properties are available in [20], [174], and [135]; and [14] contains information on characterization and categorization.

NOTE For example, the history of rigorous professional attention to the theory of systems reliability and availability goes back to the 1930's and before, and serious attention to computer and software security goes back at least into the 1960's with a substantial amount of important work occurred in the 1970's and early 1980's. Luckily, a project at the University of California Davis collected many of these seminal computer security works and placed them on the Internet [184]. While their contents may be reflected in later publications, these works are still relevant today – for example, [183], [203] and [7].

E.3.2 Confidentiality

Computing-related confidentiality topics include access control, encryption, hiding, intellectual property rights protection, traffic analysis, covert channels, inference, and anonymity. The last four are discussed here.

E.3.2.1 Traffic analysis

The levels, sources, and destinations of communications traffic can sometimes be revealing even if the content is encrypted. For example, traffic increases in organizations tend to foreshadow major events. The main issues in traffic analysis are ease of detection and analyzability. Factors include concealment of origin and destination of communications and the levelling or randomization of traffic volumes and message sizes.

E.3.3 Covert channels

Covert channels are “abnormal” means of communication using such means as timing of overt messages, locations in messages not normally used (e.g. unused bits in packet headers), or (un)availability of resources to convey messages. These may be ignored in low or moderate security situations. While covert channels based on resources can potentially be eliminated, the objectives in high-security systems are usually to identify and minimize covert channels of all kinds. Covert communication channels are measured by the bit rate that they can carry. See [[20], p. 462-469], [153], and [[159], Chapter 8].

E.3.4 Data aggregation inference

Potential can exist to violate confidentiality or privacy by aggregating data whose individual disclosure would not result in harm. Identity theft is often facilitated by the attacker aggregating data.

E.3.5 Inference

Confidential data may be inferable from other data that is available. One example is inferring individual data by comparing data for different groups – an individual's grade in a course can be calculated from the average grade in the course and the average grade of everyone but the individual.

E.3.6 Anonymity

Anonymity can involve concealing one's identity, activities, attributes, relationships, and possibly existence. Issues include concealing the identity associated with particular data and who is communicating with whom including determining that the same (but unidentified) entity is involved in two communications – linkage. Desired or required privacy[8] is one motivation for anonymity. [23]

E.3.7 Formal security models for confidentiality

A formal security model is a mathematically precise statement of a security policy. Such a model must define a secure state, an initial state, and how the model represents changes in state. The model must be shown to be secure by proving the initial state is secure and all possible subsequent states remain secure. David Bell and Leonard LaPadula of the MITRE Corporation defined the first formal model of confidentiality[9], which stated that if multiple hierarchical levels of confidentiality exist, then one cannot write higher confidentiality data into lower confidential areas and one cannot from a lower confidentiality area read something at a higher level. See [[20], Chapter 5] for an extended exposition also including definitions of "basic" and "simple" security.

A more modern (1980's) model is non-interference. The two concepts are that no one at a lower level of confidentiality should see behaviour that (1) results in any way from any behaviour at a higher level – non-interference [[20], p. 448-50] – or alternately (2) from which any information can be derived about behaviour at a higher level – probabilistic non-interference [52].

E.3.8 Integrity

To maintain system integrity one needs to keep the system in legitimate states or conditions. "Legitimate" must be specified – an integrity security policy could be conditional. For example, it might be allowable for the system to enter otherwise illegitimate states during a transaction, as long as it returns to a legitimate state at the end of the transaction. Early on Biba establish a fundamental integrity property [20] and Clark and Wilson [26] provided in 1987 a discussion of commercially relevant integrity.

Two key sub-problems within integrity are:

- Has something changed?
- Were all of the implemented changes authorized?

Checking that data is unchanged can only have meaning in terms of the question, "Since when?" In practice, this usually means that one must query, "Since in whose possession?" (This possession may or may not be at a specified time.)

Kinds of items where proper privileges and authorization can be of concern include:

- Creating.
- Viewing.
- Changing.
- Executing.
- Communicating.
- Sharing.

[8] Including protection from cyberstalking

[9] David Elliott Bell and Leonard J. LaPadula, "Secure computer systems: mathematical foundations". MITRE Corporation, 1973 - and - "Secure computer systems: unified exposition and MULTICS interpretation". MITRE Corporation, 1976.

- Encrypting/decrypting.
- Deleting/destroying.

In discussing integrity-related change authorizations, changes commonly concern:

- Credentials (evidence of identity and possibly other attributes).
- Privileges,
- Data.
- Software (possibly considered data).
- The point(s) or paths of execution.
- Time (e.g. resetting the system clock)..

Sequence and structure can also be the concern of “integrity” properties. For example, transactional integrity ensures that all parts of a transaction succeed, or none do—it is atomic. Relational integrity (in relational databases) enforces that master-detail relationships are correctly maintained (e.g., if you delete a purchase order, you delete related “detail” records such as purchase order lines enumerating items and quantities ordered.). As mentioned, in 1977, K.J. Biba of the MITRE Corporation defined a mandatory integrity policy model that provided a corollary to the Bell-LaPadula mandatory security model.[10]

E.3.9 Availability

Along with reliability, engineering for availability has a long history in computing. Many traditional approaches and means of prediction exist, but all presume lack of maliciousness. (This is no longer so common in the related area of disaster recovery.) As with all security properties, achieving a specified level of availability is a more difficult problem because one must consider maliciousness. Some of the old approaches and almost all the means of calculation no longer work.

Denial of service attacks from outside – particularly distributed ones originating from many computers simultaneously – can be difficult to successfully overcome. Non-distributed attacks that attempt to take over, exhaust, or destroy resources (e.g. exhaust primary storage) also are a threat. Interestingly, any mechanism designed to deny illegitimate access can tempt attackers to discover a way to use it to deny legitimate access (e.g. locking accounts after a certain number of incorrect passwords tries would allow a malicious person to lock one out of one’s account by multiple tries to log in as one with random passwords). Speed of repair or recovery can affect availability.

From a security viewpoint, systems need not only to remain available but preserve their other required security properties, e.g. confidentiality, whether available are not.

E.3.10Accountability

For entities that interact with the system to be held accountable for their actions, those entities must be identified. “Each access to information must be mediated based on who is accessing the information and what classes of information they are authorized to deal with. This identification and authorization information must be securely maintained by the computer system and be associated with every active element that performs some security-relevant action in the system.”[11]

Audit information enables actions affecting security to be traced to the responsible party. The system should be able to record the occurrences of security-relevant events in an audit log or other protected event log. The ability to select the audit events to be recorded is necessary to minimize the expense of auditing and to allow efficient analysis.

[10] K. J. Biba. “Integrity Considerations for Secure Computer Systems” (in MITRE Technical Report TR-3153). The MITRE Corporation, April 1977.

[11] Source: DOD 5200.28-STD, Department of Defense Trusted Computer Evaluation Criteria, December 1985.

Audit data must be protected from modification and unauthorized destruction and, in some environments, their confidentiality must be protected. Because they permit detection and after-the-fact forensic investigations of security violations[12], audit logs can become the targets of attacks that attempt to modify or delete records that could indicate an attacker's or malicious insider's actions. In systems that process sensitive data, the audit logs may contain portions of that data, and thus would need to be protected as appropriate for the sensitivity level of that data. In addition, the design of intrusion detection and auditing mechanisms must avoid allowing the exhaustion of log storage space to become a form of attack.

E.3.11Non-repudiation

Non-repudiation provides proof that any entity that uses a system or acts upon data cannot later deny those actions. Non-repudiation forces users to assume responsibility for their actions so that they cannot disclaim those actions "after the fact" nor deny any event related to themselves—for example, they cannot deny (or repudiate) having been the sender, authorizer, or recipient of a message. Several means of achieving non-repudiation involve cryptographic signatures (more frequently called digital signatures).

ISO/IEC 13888 Information technology – Security techniques – Non-repudiation addresses both symmetric and asymmetric techniques. In symmetric non-repudiation, both the sender and recipient of information are provided with proofs: the sender receives proof that the information was received by the recipient; the recipient receives proof of the identity of the sender. In asymmetric non-repudiation, proof is provided to only one of the parties in a two-party transaction regarding an action of the other party (e.g., sender receives proof of delivery, or recipient receives proof of sender identity, but not both).

E.3.12Protecting privacy

Privacy needs are one of the key reasons for security. Privacy is a motivation for confidentiality, anonymity, and not retaining data. Avoiding falsehoods that could damage reputations requires data accuracy and integrity. Concern for privcy is widespread, and several relevant laws and regulations exist industry-oriented, sub-national, national, and international.

E.3.13Safety and security

Not everyone defines the same agreed to boundary between safety and security. However, despite these disagreements many tend toward the position that traditionally they share concerns for adverse consequences and non-malicious but dangerous actions, and security is additionally concerned with malicious and illegal or illegitimate actions as well as confidentiality.[13] Safety concerns often centre on lives, health, and environmental damage as well as property damage.

In recent years, the safety community has more examples of successful experience with producing very-low-defect and high-confidence systems and software than does the software security community. The safety community's experience provides valuable lessons for software security practitioners in both producing and assuring software in high-consequence systems (for an introduction and example see [192], [56], and [55]). However, the traditional safety engineering approach differs from the security one in a critical way – it presumes non-existence of maliciousness. Today, security is a concern for most systems as many are directly or indirectly exposed to the Internet or to insider attack as well as to subversion during development, deployment, and updating. While safety-oriented systems so exposed now must also face the security problem, this sub-clause speaks of traditional safety engineering that does not address maliciousness.

Safety and security are often mentioned together, and some advantage might derive from treating them together. For example, of an approach to an assurance case including security and safety is proposed in [181][182][14].

[12] Other forensic support includes support for identifying suspects and investigating insiders and outsiders. For insiders where the identity of the user may be known, automated recognition of use in an unusual fashion could help support identification of suspects.

[13] Also, security concerns often have more interest in confidentiality than safety concerns do.

[14] The objective of SafSec, developed by Praxis High Integrity Systems is to provide a systems certification and accreditation (C&A) methodology that addresses both safety and security acceptance requirements. SafSec was originally developed for C&A of the United Kingdom Ministry of Defense (UK MOD) Integrated Modular Avionics, and other

When both are required, a number of areas are candidates for partially combining safety and security engineering concerns including:

- Understanding of the situation.
- Goals.
- Solutions.
- Activities.
- Assurance case[15].
    - Claims and particularly subclaims.
    - Arguments.
    - Evidence.
    - Assumptions.
- Evaluations.

In addition, both safety and security have practical concerns for correctness.

12.1.1 Other security-related concerns

In addition to a product's preservation of required security properties within its digital domain, it can contribute to other systems, organizational or societal security goals including:

- Establishing the authenticity of users and data.
- Establishing accountability of users.
- Providing usability including transparency to users to gain acceptance and resulting security.
- Providing the abilities to:
- Deter and mislead attackers,
- Force attackers to overcome multiple layers of defence,
- Support investigations to identify and convict attackers.
- Limiting real-world damage.
- Aiding physical security, such as in monitoring and entrance control.

Thus, digital systems can help address security concerns at a number of levels.

advanced avionics architectures. SafSec is an example of how approaches from assurance of safety as a required property of software have been applied to the assurance of security properties of software.

[15] The SafSec effort provides guidance on one way to combine assurance cases. [SafSec Standard [182]] [SafSec Guidance [181]]

E.4 Traditional system and software security principles and guidelines

E.4.1 Introduction

Saltzer and Schroeder published their list of principles in 1975, and they remain important [183]. Everyone involved in any way with secure software systems needs to be aware of them. Most of the list below follows the principles proposed by Saltzer and Schroeder and liberally quotes edited selections from that text. These principles have relevance throughout secure software system development and sustainment including requirements; design; construction; and verification, validation, and evaluation.

They cover a number of topics. Several principles help in reducing the number of opportunities for violations. As opportunities or possibilities for violations cannot always be eliminated, steps need to be taken to ensure users properly utilize security and efforts toward security are expended in the best places. To reduce the uncertainties related to the adequacy or correctness of the software system, the portion of the system and mechanisms ensuring security should be as small and simple as practicable and be thoroughly reviewed and analysed.

Defences and protection may not be perfect, and violations will occur. For follow-up, learning, and improvement records of what occurred are needed. In addition, requiring multiple successes by an attacker before substantial damage results can increase time or effort attacker needs to expend and provide some tolerance for vulnerabilities or weaknesses.

E.4.2 Least privilege

Least privilege is a principle whereby each entity (user, process, or device) is granted the most restrictive set of privileges needed for the performance of that entity's authorized tasks. Application of this principle limits the damage that can result from accident, error, or unauthorized use of a system. Least privilege also reduces the number of potential interactions among privileged processes or programs, so that unintentional, unwanted, or improper uses of privilege are less likely to occur.

E.4.3 Complete mediation

Every access to every (security-sensitive) object must be checked for proper authorization; and access denied if it violates authorizations. This principle, when systematically applied, is the primary underpinning of the protection system, and it implies the existence and integrity of methods to (1) identify the source of every request, (2) ensure the request is unchanged since its origination, and (3) check the relevant authorizations as well as ensure request denied if unauthorized and not otherwise (unless by some other mechanism). It also requires that design proposals to allow access by remembering the result of a prior authority check be examined sceptically.

E.4.4 Fail-Safe defaults

This principle calls for basing access decisions on permission rather than exclusion. Thus, the default situation is lack of access, and the protection scheme identifies conditions under which access is permitted. To be conservative, a design must be based on arguments stating why objects should be accessible, rather than why they should not.

E.4.5 Least common mechanism

Minimize the security mechanisms common to more than one user or depended on by multiple users or levels of sensitivity. Whenever the same executing process services multiple users or handles data from multiple security levels or compartments, this potentially creates an opportunity for illegitimate information flow. Every shared mechanism (as opposed, for example, to non-communicating multiple instances) represents a potential information path between users or across security boundaries and must be designed with great care to ensure against unintentionally compromising security. Virtual machines each with their own copies of the operating system are an example of not sharing the usage of the operating system mechanisms – a single instance is not common across the users of the virtual machines. Thus, one desires the least possible sharing of common mechanisms (instances).

E.4.6 Separation of privilege

A protection mechanism that requires two keys to unlock it is more robust and flexible than one that allows access to the presenter of a single key. By requiring two keys, no single accident, deception, or breach of trust is sufficient to compromise the protected information.

Redundancy is also used in the traditional "separation of duties" in human financial processes, e.g. the persons who fill out the check and sign it are two different people.

E.4.7 Psychological acceptability

It is essential that the human interface be designed for ease of use, so that users routinely and automatically apply the protection mechanisms correctly.[16]

E.4.8 Work factor

The cost of performing to a higher level of quality or of a countermeasure to eliminate or mitigate vulnerability should be commensurate with the cost of a loss if a successful attack were to otherwise occur. Generally, the more valuable the asset targeted by an attacker, the more effort and resources that attacker is willing expend, and therefore the most effort and resources the defender should expend to prevent or thwart the attack.

E.4.9 Economy of mechanism

"Keep the design as simple and small as possible" applies to any aspect of a system, but it deserves emphasis for protection mechanisms, since design and implementation errors that result in unwanted access paths may not be noticed during normal use.

E.4.10Open design

Security mechanisms should not depend on the ignorance of potential attackers, but rather on assurance of correctness and/or the possession of specific, more easily protected, keys or passwords. This permits the mechanisms to be examined by a number of reviewers without concern that the review may itself compromise the safeguards.

The practice of openly exposing one's design to scrutiny is not universally accepted. The notion that the mechanism should not depend on ignorance is generally accepted, but some would argue that its design should remain secret since a secret design may have the additional advantage of significantly raising the price of penetration. The principle still can be applied, however, restricted to within an organization or a "trusted" group.

E.4.11Analyzability

Systems whose behaviour is analyzable from their engineering descriptions such as design specifications and code have a higher chance of performing correctly because relevant aspects of their behaviour can be predicted in advance. In any field, analysis techniques are never available for all structures and substances. To ensure analyzability one must restrict structural arrangements and other aspects to those that are analyzable.[17]

E.4.12Recording of compromises

If a system's defence was not fully successful, trails of evidence should exist to aid understanding, recovery, diagnosis and repair, forensics, and accountability. Likewise, records of suspicious behaviour and "near misses", and records of legitimate behaviours can also have value.

[16] Usable security is a significant issue and is addressed in both the Requirements and Design sections.

[17] Certainly one would never want a critical structure such as a bridge to be built using a structural arrangement whose behaviour could not be analysed and predicted. Why should this be different for critical software?

E.4.13Defence in depth

Defence in depth is a strategy in which human, technological, and operational capabilities are integrated to establish variable protective barriers across multiple layers and dimensions of a system. This principle ensures that an attacker must compromise more than one protection mechanism to successfully exploit a system. Diversity of mechanisms can make the attacker's problem even harder. The increased cost of an attack may dissuade an attacker from continuing the attack. Note that multiple less expensive but weak mechanisms do not necessarily make a stronger barrier than fewer more expensive and stronger ones.

E.4.14Treat as conflict

E.4.14.1 Introduction

Because many security-related issues and activities concern a conflict pitting the system and those aiding in its defence against those who are attacking or may attack in the future, one needs to bring to the situation all that one can of what is known about conflicts. This includes recognizing when attacks in a particular area (e.g. the computer security arena) are part of a wider conflict. Below are some of the key concepts.

E.4.14.2 Intelligent and malicious adversary

The dangers (e.g. threats or hazards) faced may come from a number of sources including intelligent, skilled adversaries. When possession, damage, or denial of assets or interests is highly valuable to someone else, then considerable skill and resources could be brought to bear. When – as is often the case – poor security makes these actions relatively easy and risk free, even things of low value may become attractive.

One cannot simply use a probabilistic approach to one's analyses because, for example, serious, intelligent opponents tend to attack where and when you least expect them – where your estimate of the probability of such an attack is relatively low.[18]

Where judging dangers and consequences is difficult, one possible approach is to make them at least not unacceptable (tolerable) and "as low as reasonably practicable" (ALARP).[19] In employing the ALARP approach, judgments about what to do are based on the cost-benefit of measures or techniques one might apply – not on total budget. However, additional techniques or efforts are not employed once risks have achieved acceptable (not just tolerable) levels. This cost-benefit at the margin (that is, net benefit from a best possible additional step) approach is attractive from an engineering viewpoint, but the amount of benefit cannot always be accurately established.

E.4.14.3 Security is a system, organizational, and societal problem

Security is not merely a specialized concern, and mistaken decisions can result from confining attention to only the product or elements of it. Attempts to break security are often part of a larger conflict such as business competition, crime and law enforcement, social protest, political rivalry, or conflicts among nation states, e.g. espionage. Specifically in secure product development, the social norms and ethics of members of the development team and organization as well as suppliers deserve serious attention. Insiders constitute one of the most dangerous populations of potential attackers, and communicating successfully with them concerning their responsibilities and obligations is important in avoiding subversion or other damage.

E.4.14.4 Measures encounter countermeasures

Despite designers' positive approach to assuring protection, in practice one also engages in a measure-countermeasure cycle between offence and defence. Currently, reducing the attackers' relative capabilities and increasing systems' resilience dominates many approaches. Anonymity of attackers has led to

[18] In theory, game theory techniques that do not require the probabilities to be known could be applicable, but little progress has been made in practice.

[19] ALARP is a significant concept in UK law, and an excellent engineering-oriented discussion of it appears in Annex B of DEF STAND 00-56 Part 2 [Ministry of Defence 2004b]

asymmetric situations where defenders must defend everywhere and always,[20] and attackers can chose the time and place of their attacks. Means for reducing anonymity – thereby presumably making deterrence and removal of offenders more effective – could somewhat calm the current riot of illicit behaviour.

E.4.14.5 Learn and adapt

The attack and defence of systems is a normal but not a mundane situation; it is a serious conflict situation with serious adversaries such as criminal organizations, terrorist groups, and nation states as well as competitors committing industrial espionage. Serious analyses of past and current experience can improve tactics and strategy.

One should not forget how to be successful in conflicts. While it is difficult to state the principles of conflict in a brief manner, some principles exist such as exploiting the arenas in which the conflict occurs; using initiative, speed, movement, timing, and surprise; using and trading-off quality and quantity including technology and preparation; carefully defining success and pursuing it universally and persistently but flexibly; and hindering adversaries.

[20] Defending everything may not be possible or may waste resources. “He who defends everything defends nothing.” – Frederick II

Annex F
(Informative)

Selected Related Standards

F.1 Introduction

This informative annex lists some of the International Standards related to systems and software and their dependability, safety, security, and some other properties. A brief description of many standards is provided. The discursive part of this material derives in part from work by the US National Defense Industry Association and is used with their permission.

F.2 ISO/IEC 2382-14:1997 Information technology -- Vocabulary -- Part 14: Reliability, maintainability and availability

F.3 ISO 2394:1998 General principles on reliability for structures

F.4 ISO 9000-3, 12207 and 15288 - SDLC QA standards

Extending the general-purpose Quality Assurance standards in the ISO 9000 series, the following standards cover the application of QA processes to the System or Software Development Life Cycle specifically and have applicability to the requirements of system assurance:

1.1. **ISO 9000 Part 3** *Guidelines for the application of ISO 9001 to the development, supply and maintenance of software* covers software engineering, guiding the application of ISO 9000, the quality assurance standards, to the systems development process:

- **ISO 12207** covers software life cycle processes, providing a conceptual framework and terminology.
- **ISO 15288**:2002 *System Life Cycle Processes* covers systems engineering by defining a set of processes and terminology.

F.5 ISO 13335 - IT security ISO 9241-400:2007 Ergonomics of human--system interaction -- Part 400: Principles and requirements for physical input devices

F.6 ISO 12100-1:2003 Safety of machinery -- Basic concepts, general principles for design -- Part 1: Basic terminology, methodology

F.7 ISO 12100-2:2003 Safety of machinery -- Basic concepts, general principles for design -- Part 2: Technical principles

F.8 ISO 13335 Information technology – Security techniques – Management of information and communications technology security – series

ISO 13335 (which started life as a Technical Report TR before becoming a full ISO standard) comprises a set of guidelines for the management of IT security, focusing primarily on technical security control measures:

- **ISO 13335-1**:2004 “Information technology – Security techniques – Management of information and communications technology security – Part 1: Concepts and models for information and communications technology security management”. Explains the concepts and models for information and communications technology security management. (ISO/IEC TR 13335 parts 1 and 2 were combined into the revised ISO/IEC 13335-1: 2004. The original TR13335-2:1997 “Guidelines for the management of IT security - Part 2: Managing and planning IT security” was cancelled.)
- **ISO 13335-2**, when published, is expected to cancel and replace ISO/IEC TR 13335-3:1998 and ISO/IEC TR 13335-4:2000.
- **ISO TR 13335-3**:1998 “Information technology – Guidelines for the Management of IT Security – Part 3: Techniques for the management of IT Security”. Covers techniques for the management of IT security. This standard’s revision inserts into ISO 27005
- **ISO TR 13335-4**:2000 covers the **selection of safeguards** (meaning technical security controls). This standard is also currently under revision and will be inserted into ISO 27005
- **ISO TR 13335-5**:2001 provides management guidance on network security. This standard is currently under revision, being merged into ISO/IEC 18028-1. ISO/IEC 18028-1 will eventually cancel and replace ISO/IEC TR 13335-5:2001.ISO 15408 - Common Criteria

F.9ISO 13849 Safety of machinery -- Safety-related parts of control systems – series

- ISO 13849-1:2006 Safety of machinery -- Safety-related parts of control systems -- Part 1: General principles for design
- ISO 13849-2:2003 Safety of machinery -- Safety-related parts of control systems -- Part 2: Validation

F.10 ISO 14620 Space systems -- Safety requirements -- series

- ISO 14620-1:2002 Space systems -- Safety requirements -- Part 1: System safety
- ISO 14620-2:2000 Space systems -- Safety requirements -- Part 2: Launch site operations
- ISO 14620-3:2005 Space systems -- Safety requirements -- Part 3: Flight safety systems

F.11 ISO 14625:2007 Space systems -- Ground support equipment for use at launch, landing or retrieval sites -- General requirements

F.12 ISO/IEC TR 15446:2004, Information technology -- Security techniques -- Guide for the production of Protection Profiles (PPs) and Security Targets (STs)

ISO/IEC TR 15446 provides guidance relating to the construction of PPs and STs that are intended to be compliant with ISO/IEC 15408 (the "Common Criteria") and gives suggestions on how to develop each section of a PP or ST. This supported by an annex that contains generic examples of each type of PP and ST component, and by other annexes that contain detailed worked examples.

F.13 ISO 15408:1999

ISO 15408:1999 describes the Common Criteria for Information Technology Security Evaluation. Products that are evaluated against the Common Criteria have a defined level of assurance as to their information security capabilities that is recognized in most of the world. Unfortunately, the evaluation process is quite costly and slow, and is therefore not very widely used apart from the government and defence markets.

F.14 ISO/IEC TR 15543, Information technology—Security techniques—A framework for IT security assurance

- **General**
  - Guides selection of an appropriate assurance method when specifying, selecting, or deploying a security service, product, or environmental factor such as an organization or personnel (known as a deliverable).
  - Facilitates the understanding of the assurance type and effort required to achieve confidence that the deliverable satisfies stated IT security assurance requirements and security policy.
  - Describes fundamentals of security assurance and relation to other security concepts:
    - Clarifies why security assurance is required and dispels misconceptions that increased assurance is gained by increasing the strength of security mechanisms.
    - Includes a categorization of assurance types and a generic life cycle model to identify the appropriate assurance types required for the deliverable:
      - Demonstrates how security assurance must be managed throughout the deliverable's life cycle requiring assurance decisions to be made by several assurance authorities for the life cycle stage relevant to their organization (i.e. developer, standards, and consumer).
      - Accommodates different assurance types and maps into any life cycle approach so as not to dictate any particular design.
  - Includes advanced security assurance concepts, such as combining security assurance methods.
- **The three parts of ISO/IEC TR 15543**
  - Part 1, Overview and Framework provides fundamental concepts and general description of assurance methods:
    - Targets IT security in developing a security assurance program, determining the security assurance of deliverables, entering assurance assessment audits (e.g. ISO 9000, ISO/IEC 21827, ISO/IEC 15408-3), or other assurance activities.
  - Part 2, Assurance Methods describes a variety of assurance methods and approaches and relates them to Part 1 security assurance framework model:
    - Identifies qualitative properties of assurance methods.
    - Aids in understanding how to obtain assurance in a given life cycle stage of deliverable.
  - Part 3, Analysis of Assurance Methods analyses the various methods with respect to their assurance properties and aids Assurance Authorities:
    - In deciding relative value of Assurance Approaches and determining that they will provide the assurance results most appropriate to their needs.
    - To use assurance results to achieve desired confidence of the deliverable.

F.15 ISO/TR 17944:2002 Banking -- Security and other financial services -- Framework for security in financial systems

F.16 ISO/IEC 18014-2:2002 Information technology -- Security techniques -- Time-stamping services – [Three Parts]

F.17 ISO/IEC 18028 Information technology -- Security techniques -- IT network security

F.18 ISO 19706:2007 Guidelines for assessing the fire threat to people

F.19 ISO 19770 - Software asset management

ISO/IEC 19770-1:2006 promotes the implementation of an integrated set of software asset management processes, using good practices for efficient software management. Contents include Scope, terms and definitions, Field of application, Conformance, Intended usage, Agreement compliance, General Software Asset Management processes, Control environment for Software Asset Management, Planning and implementation, Inventory processes, Verification and compliance processes, Operations management processes and interfaces, Life cycle process interfaces.

F.20 ISO/IEC TR19791 Information technology -- Security techniques -- Security assessment of operational systems

ISO/IEC TR 19791:2006 provides guidance and criteria for the security evaluation of operational systems. It provides an extension to the scope of ISO/IEC 15408. The main additions address evaluation of the operational environment surrounding the target of evaluation, and the decomposition of complex operational systems into security domains that can be separately evaluated. It does not define techniques for the identification, assessment and acceptance of operational risk.

F.21 ISO 21827 - Systems Security Engineering Capability Maturity Model

Like other Capability Maturity Models (CMMs), the Systems Security Engineering (SSE) CMM defines the essential characteristics of SSE processes, emphasizing those which indicate process maturity. The model covers the entire systems development life cycle from concept definition to decommissioning. It applies to those developing or integrating secure products/systems, and those providing specialist security services such as security engineering. It was published as ISO 21827 in 2002.

F.22 ISO/IEC 25000 series

This series covers software quality.

F.23 ISO/IEC 25020:2007 Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Measurement reference model and guide

F.24 ISO/IEC 25051:2006 Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing

F.25 ISO/TS 25238:2007 Health informatics -- Classification of safety risks from health software

F.26 ISO/IEC 26702:2007 Systems engineering -- Application and management of the systems engineering process

ISO/IEC 26702:2007 defines the interdisciplinary tasks which are required throughout a system's life cycle to transform customer needs, requirements and constraints into a system solution. In addition, it specifies the requirements for the systems engineering process and its application throughout the product life cycle. ISO/IEC 26702:2007 focuses on engineering activities necessary to guide product development, while ensuring that the product is properly designed to make it affordable to produce, own, operate, maintain and eventually dispose of without undue risk to health or the environment.

F.27 ISO/IEC 27000-series

ISO has reserved the ISO/IEC 27000-series numbering for a range of information security management standards. The following ISO 27000-series standards are either published or planned.

F.27.1 ISO 27000

Contains the vocabulary and definitions i.e. terminology for all of these information security management standards

F.27.2 ISO 27001

Is the Information Security Management System requirements standard (specification) against which organizations are formally certified compliant In October 2005, British Standard BS 7799 part 2 was adopted by ISO, re-badged and released as the new international information security standard ISO/IEC 27001:2005. ISO 27001 is the formal standard against which organizations may seek independent certification of their Information Security Management Systems (meaning their frameworks to design, implement, manage, maintain and enforce information security processes and controls systematically and consistently throughout the organizations).

F.27.3 ISO 27002

The standard currently known as ISO 17799 and formerly known as BS 7799 part 1. This is the code of practice for information security management describing a comprehensive set of information security control objectives and a menu of best-practice security controls

F.27.4 ISO 27003

An implementation guide

F.27.5 ISO 27004

Information security management measurement standard to help measure the effectiveness of information security management system implementations.

F.27.6 ISO 27005

Will be an information security risk management standard (will replace the recently issued BS 7799 Part 3).

F.27.7 ISO 27006 - ISO/IEC 27006:2007 Information technology -- Security techniques -- Requirements for bodies providing audit and certification of information security management systems

A guide to the certification/registration process for accredited ISMS certification/registration bodies.

F.27.8ISO/TR 27809:2007 Health informatics -- Measures for ensuring patient safety of health software

F.28 ISO 28003:2007 Security management systems for the supply chain -- Requirements for bodies providing audit and certification of supply chain security management systems

Bibliography

[1] Abran, Alain, James W. Moore (Executive editors); Pierre Bourque, Robert Dupuis, Leonard Tripp (Editors). Guide to the Software Engineering Body of Knowledge. 2004 Edition. Los Alamitos, California: IEEE Computer Society, Feb. 16, 2004. Available at http://www.swebok.org.

[2] Adamski, A, and R. Westrum. "Requisite imagination: The fine art of anticipating what might go wrong." In [58], p. 193-220, 2003.

[3] Adelard. The Adelard Safety Case Development Manual. Available at http://www.adelard.com/web/hnav/resources/ascad.

[4] Alexander, Ian. Systems Engineering Isn't Just Software. 2001. Available at http://easyweb.easynet.co.uk/~iany/consultancy/systems_engineering/se_isnt_just_sw.htm.

[5] All;en, Julua H., Sean Barum, Robert J. Ellison, Gary McGraw, and Nancy R. Mead. Software Security Engineering: A Guide for Project Managers. Addison-Wesley, 2008.

[6] Altman, W., Ankrum, T., Brach, W. “Improving Quality and the Assurance of Quality in the Design and Construction of Nuclear Power Plants: A Report to Congress.” U.S. Nuclear Regulatory Commission: Office of Inspection and Enforcement. 1987.

[7] Anderson, James P. “Computer Security Technology Planning Study Volume I”, ESDTR-73-51, Vol. I, Electronic Systems Division, Air Force Systems Command, Hanscom Field, Bedford, MA 01730 (Oct. 1972).

[8] Anderson, Ross J. Security Engineering: A Guide to Building Dependable Distributed Systems, Second Edition. John Wiley and Sons, 2008.

[9] Armstrong, J.M. and Paynter S.P. “The Deconstruction of Safety Arguments through Adversarial Counter-argument.” School of Computing Science, Newcastle University CS-TR-832, 2004.

[10] Ankrum, T. Scott, and Alfred H. Kromholz, "Structured Assurance Cases: Three Common Standards," Ninth IEEE International Symposium on High-Assurance Systems Engineering (HASE'05), pp. 99-108, 2005.

[11] Atchison, B., Lindsay, P., and Tombs, D. “A Case Study in Software Safety Assurance Using Formal Methods.” Technical Report No. 99-31. Sept. 1999.

[12] ATSIN Number 17 Issued 9. Lapses and Mistakes. Air Traffic Services Information Notice, Safety Regulation Group, ATS Standards Department. UK Civil Aviation Authority, August 2002.

[13] Australian Government: Department of Industry, Tourism and Resources. “National Offshore Petroleum Safety Case Guidelines.” [Online Document] 20 Sept 2006 [cited on: 13 Feb 2007] Available HTTP: http://www.industry.gov.au/content/itrinternet/cmscontent.cfm?objectID=F5BA1B26-65BF-4956-B95174C5AF384515

[14] Avizienis, Algirdas, Jean-Claude Laprie, Brian Randell, and Carl Landwehr, “Basic Concepts and Taxonomy of Dependable and Secure Computing,” IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 1, pp. 11-33, Jan.-Mar. 2004. Available at http://csdl.computer.org/dl/trans/tq/2004/01/q0011.pdf

[15] AVSAF: Aviation Safety. [Website] 2004 [cited on: 09 Feb 2007] Available HTTP: http://www.avsaf.org/case/

[16] Bahill, A.T. and B. Gissing, "Re-evaluating Systems Engineering Concepts Using Systems Thinking". IEEE Transaction on Systems, Man and Cybernetics, Part C: Applications and Reviews, Vol. 28 No. 4 pp. 516-527, November 1998.

[17] Berg, Clifford J. High-Assurance Design: Architecting Secure and Reliable Enterprise Applications, Addison Wesley, 2006.

[18] Bernstein, Lawrence and C. M. Yuhas. Trustworthy Systems through Quantitative Software Engineering. Wiley-IEEE Computer Society Press, 2005. About reliability not security.

[19] Bishop, Matt, and Sophie Engle. “The Software Assurance CBK and University Curricula.” Proceedings of the 10th Colloquium for Information Systems Security Education, 2006.

[20] Bishop, Matt. Computer Security: Art and Practice, Addison-Wesley, 2003.

[21] Bishop, P. and Bloomfield, R. “A Methodology for Safety Case Development.” Industrial Perspectives of Safety-critical Systems: Proceedings of the Sixth Safety-critical Systems Symposium, Birmingham. 1998.

[22] Bishop, P. and Bloomfield, R. “The SHIP Safety Case Approach.” SafeComp95, Belgirate, Italy. Oct 1995.

[23] Buehner, M. J. & Cheng, P. W. (2005). Causal Learning. In R. Morrison & K. J. Holyoak (Eds.) Handbook of Thinking and Reasoning [62]. Cambridge University Press, pp143-168.

[24] Cannon, J. C. Privacy, Addison Wesley, 2005]

[25] Chung, Lawrence, et al. Non-Functional Requirements in Software Engineering, Kluwer, 1999.

[26] Clark, David D. and David R. Wilson, “A Comparison of Commercial and Military Computer Security Policies,” Proc. of the 1987 IEEE Symposium on Security and Privacy, IEEE, pp. 184-196, 1987.

[27] CNSS, National Information Assurance Glossary, May 2003. Available at http://www.cnss.gov/full-index.html

[28] Committee on Information Systems Trustworthiness, Trust in Cyberspace, Computer Science and Telecommunications Board, National Research Council, 1999.

[29] Committee on National Security Systems (CNSS) Instruction 4009: National Information Assurance (IA) Glossary. Revised May 2003. Available at http://www.cnss.gov/Assets/pdf/cnssi_4009.pdf.

[30] Common Criteria Recognition Arrangement (CCRA). Common Criteria v3.1 Revision 2. NIAP September 2007. Available at http://www.commoncriteriaportal.org.

[31] Common Weaknesses Enumeration. MITRE, 2007. Avail;able at http://cwe.mitre.org (200707)

[32] Cooke, NancyJ., Jamie C. Gorman, and Jennifer L. Winner. “Team Cogitation.” p. 239-268 in [45].

[33] Courtois, Pierre-Jacques. Justifying the Dependability of Computer-based Systems: With Applications in Nuclear Engineering. Springer, 2008.

[34] Cranor, Lorrie, and Simson Garfinkel. Security and Usability: Designing Secure Systems that People Can Use. O’Reilly, 2005.

[35] Dayton-Johnson, Jeff. Natural disasters and adaptive capacity. OECD Development Centre Research programme on: Market Access, Capacity Building and Competitiveness. Working Paper No. 237 DEV/DOC(2004)06, August 2004

[36] Defence Materiel Organisation, Australian Department of Defence. +SAFE, V1.2: A Safety Extension to CMMI-DEV, V1.2. Software Engineering Institute, CMU/SEI-2007-TN-006, March 2007

[37] Defence Materiel Organisation, Australian Department of Defence. +SAFE, V1.2: A Safety Extension to CMMI-DEV, V1.2. Software Engineering Institute, CMU/SEI-2007-TN-006, March 2007

[38] Department of Defense Instruction 8500.2 (6 February 2003). Information Assurance (IA) Implementation. Washington, DC: US Department of Defense, 2003. Available at http://www.dtic.mil/whs/directives/corres/pdf2/i85002p.pdf.

[39] Department of Defense Strategic Defense Initiative Organization. Trusted Software Development Methodology, SDI-S-SD-91-000007, vol. 1, 17 June 1992.

[40] Department of Homeland Security National Cyber Security Division's “Build Security In” (BSI) web site, (http://buildsecurityin.us-cert.gov).

[41] Dependability Research Group. “Safety Cases.” [Online Document] [cited on: 13 Feb 2007] Available HTTP: http://dependability.cs.virginia.edu/info/Safety_Cases

[42] Despotou, Georgios, and Tim Kelly. “Extending the Safety Case Concept to Address Dependability,” Proceedings of the 22nd International System Safety Conference, 2004.

[43] Dowd, Mark John McDonald, Justin Schuh. The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities. Addison-Wesley, 2006.

[44] Dunbar, Kevin, and Jonathan Fugelsang. “Scientific Thinking and Reasoning.” p. 705-727 in [62]

[45] Durso, Francis T. (Editor), Raymond S. Nickerson (Editor), Susan T. Dumais (Editor), Stephan Lewandowsky (Editor), Timothy J. Perfect (Editor). Handbook of Applied Cognition 2 edition. Wiley, 2007.

[46] Ellsworth, Phoebe C. “Legal Reasoning.” P. 685-704 In [62]

[47] Ericsson, K. Anders (Editor), Neil Charness (Editor), Paul J. Feltovich (Editor), Robert R. Hoffman (Editor).The Cambridge Handbook of Expertise and Expert Performance. Cambridge University Press, 2006.

[48] Eurocontrol. “Safety Case.” [Online Document] 08 Oct 2006 [cited on: 13 Feb 2007] Available HTTP: http://www.eurocontrol.int/cascade/public/standard_page/safety_case.html

[49] Fenton, N., Littlewood, B., Neil, M., Strigini, L., Sutcliffe, A., and Wright, D. “Assessing dependability of safety critical systems using diverse evidence.” IEE Proceedings – Software. 1998.

[50] Gasser, M. Building a Secure Computer System. Van Nostrand Reinhold, 1988. Available at http://nucia.ist.unomaha.edu/library/gasser.php

[51] Government of Western Australia: Department of Consumer and Employment Protection. “The Safety Case.” [Online Document] 21 April 2006 [cited on: 13 Feb 2007] Available HTTP: http://www.docep.wa.gov.au/resourcessafety/Sections/Petroleum_Hazard_Facilities/Guidance_material_and_publications/The_safety_case.html

[52] Gray, J. W. “Probabilistic Interference.” Proceedings of the IEEE Symposium on Research in Security and Privacy. IEEE, p. 170-179, 1990

[53] Greenwell, W., Strunk, E., and Knight J. “Failure Analysis and the Safety-Case Lifecycle.” IFIP Working Conference on Human Error, Safety and System Development (HESSD) Toulouse, France. Aug 2004.

[54] Greenwell, William S., John C. Knight, and Jacob J. Pease. "A Taxonomy of Fallacies in System Safety Arguments” 24th International System Safety Conference, Albuquerque, NM, August 2006.

[55] Hall, Anthony and Rodrick Chapman. “Correctness by Construction: Developing a Commercial Secure System,” IEEE Software, vol. 19, no. 1, pp.18-25, Jan/Feb 2002.

[56] Herrmann, Debra S. Software Safety and Reliability, IEEE Computer Society Press, 1999

[57] Hoglund, Greg, and Gary McGraw. Exploiting Software: How to break code. Addison-Wesley, 2004.

[58] Hollnagel, Erik. (Ed.). Handbook of cognitive task design. Lawrence Erlbaum Associates, 2003.

[59] Hollnagel, Erik. Barriers and Accident Prevention, Ashgate, 2004.

[60] Hollnagel, Erik (Editor), David D. Woods (Editor), Nancy Leveson (Editor). Resilience Engineering: Concepts and Precepts. Ashgate Pub Co, 2006.

[61] Hollnagel, Erik. “Human Error:Trick or Treat?” in [45], p. 219-238, 2007.

[62] Hollnagel, Erik. Human Factors: From Liability to Asset. Presentation, 2007. Available at www.vtt.fi/liitetiedostot/muut/Hollnagel.pdf

[63] Holyoak, Keith J. and Robert G. Morrison. The Cambridge Handbook of Thinking and Reasoning. Cambridge University Press, 2005.

[64] Howard, Michael, and David C. LeBlanc. Writing Secure Code, 2nd ed., Microsoft Press, 2002.

[65] Howard, Michael, and Steve Lipner. The Security Development Lifecycle. Microsoft Press, 2006.

[66] Howell, C. Assurance Cases for Security Workshop (follow-on workshop of the 2004 Symposium on Dependable Systems and Networks), June, 2005.

[67] Ibrahim, Linda, et al, Safety and Security Extensions for Integrated Capability Maturity Models. Washington D.C.: United States Federal Aviation Administration, Sept. 2004. Available at http://www.faa.gov/ipg/pif/evol/index.] International Council on Systems Engineering INCOSE. Guide to Systems Engineering Body of Knowledge (G2SEBoK). Available at http://g2sebok.incose.org/.

[68] IEC 50 (191) International Electrotechnical Vocabulary, Chapter 191: Dependability and Quality of Service

[69] IEC 60300 Dependability management [several parts]

[70] IEC 60300-3-2:1993, Dependability management – Part 3: Application guide – Section 2: Collection of dependability data from the field

[71] IEC 60300-3-15 Dependability management - Part 3-15-Guidance to engineering of system dependability.

[72] IEC 60812:2006, Analysis techniques for system reliability - Procedure for failure mode and effects analysis (FMEA)

[73] IEC 61025:2006, Fault tree analysis (FTA)

[74] IEC 61078:2006, Analysis techniques for dependability - Reliability block diagram and boolean methods

[75] IEC 61508:2005, Functional safety of electrical/electronic/programmable electronic safety-related systems [several parts]

[76] IEC61508-7:2000 Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 7: Overview of techniques and measures

[77] IEC 61511:200x, Functional safety - Safety instrumented systems for the process industry sector [several parts]

[78] IEC 61882:2001, Hazard and operability studies (HAZOP studies) - Application guide

[79] IEEE Std 1228-1994, IEEE Standard for Safety Plans

[80] ISO/IEC 2382-14:1997 Information technology -- Vocabulary -- Part 14: Reliability, maintainability and availability

[81] ISO 2394:1998 General principles on reliability for structures

[82] ISO 9000 Part 3 Guidelines for the application of ISO 9001 to the development, supply and maintenance of software

[83] ISO 9241-400:2007 Ergonomics of human--system interaction -- Part 400: Principles and requirements for physical input devices.

[84] ISO 12100-1:2003 Safety of machinery -- Basic concepts, general principles for design -- Part 1: Basic terminology, methodology.

[85] ISO 12100-2:2003 Safety of machinery -- Basic concepts, general principles for design -- Part 2: Technical principles.

[86] ISO 12100-2:2003 Safety of machinery -- Basic concepts, general principles for design -- Part 2: Technical principles

[87] ISO/IEC 12207:1995 Information technology -- Software life cycle processes

[88] ISO 13335 Information technology – Security techniques – Management of information and communications technology security – series

[89] ISO 13849 Safety of machinery -- Safety-related parts of control systems – series

[90] ISO 14620 Space systems -- Safety requirements -- series

[91] ISO 14625:2007 Space systems -- Ground support equipment for use at launch, landing or retrieval sites -- General requirements

[92] ISO/IEC 15288:2002 Systems engineering -- System life cycle processes

[93] ISO/IEC 15408:2005, Information technology -- Security techniques -- Evaluation criteria for IT security [three parts]

[94] ISO/IEC TR 15446:2004, Information technology -- Security techniques -- Guide for the production of Protection Profiles (PPs) and Security Targets (STs)

[95] ISO/IEC 15443:2005, Information technology--Security techniques--A framework for IT security assurance [three parts]

[96] ISO/IEC 15939:2007 Systems and software engineering -- Measurement process

[97] ISO/IEC 16085:2006, Systems and software engineering -- Life cycle processes -- Risk management

[98] ISO/IEC TR 16326:1999 Software engineering -- Guide for the application of ISO/IEC 12207 to project management

[99] ISO/TR 16982:2002 Ergonomics of human-system interaction -- Usability methods supporting human-centred design

[100] ISO/TR 17944:2002 Banking -- Security and other financial services -- Framework for security in financial systems.

[101] ISO/IEC 18014-2:2002 Information technology -- Security techniques -- Time-stamping services – [Three Parts]

[102] ISO/IEC 18028 Information technology -- Security techniques -- IT network security

[103] ISO/TR 18529:2000 Ergonomics -- Ergonomics of human-system interaction -- Human-centred lifecycle process descriptions

[104] ISO 19706:2007 Guidelines for assessing the fire threat to people

[105] ISO 19770 - Software Asset Management

[106] ISO/IEC TR19791 Information technology -- Security techniques -- Security assessment of operational systems

[107] ISO/TS 20282-2:2006 Ease of operation of everyday products (series)

[108] ISO/IEC 21827:2002, Information technology -- Systems Security Engineering -- Capability Maturity Model (SSE-CMM®)

[109] ISO/IEC 25000 Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE

[110] ISO/IEC 25010 Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) – Quality model

[111] ISO/IEC 25020:2007 Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Measurement reference model and guide

[112] ISO/IEC 25030:2007 Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Quality requirements

[113] ISO/IEC 25040 Software engineering - Software product Quality model and guide – Requirements and Evaluation (SQuaRE) – Evaluation reference

[114] ISO/IEC 25051:2006 Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing

[115] ISO/TS 25238:2007 Health informatics -- Classification of safety risks from health software

[116] ISO/IEC 26702:2007 Systems engineering -- Application and management of the systems engineering process

[117] ISO/IEC CD 27000 Information technology -- Security techniques -- Information security management systems -- Overview and vocabulary

[118] ISO/IEC 27001:2005 Information technology -- Security techniques -- Information security management systems -- Requirements

[119] ISO/IEC 27002:2005 Information technology -- Security techniques -- Code of practice for information security management

[120] ISO/IEC CD 27004 Information technology -- Security techniques -- Information security management measurements

[121] ISO/IEC FDIS 27005 Information technology -- Security techniques -- Information security risk management

[122] ISO/IEC 27006:2007 Information technology -- Security techniques -- Requirements for bodies providing audit and certification of information security management systems

[123] ISO/IEC FCD 27011 Information technology -- Security techniques -- Information security management guidelines for telecommunications

[124] ISO/IEC TR 24748 Systems and software engineering — Guide for life cycle management

[125] ISO/TR 27809:2007 Health informatics -- Measures for ensuring patient safety of health software

[126] ISO 28003:2007 Security management systems for the supply chain -- Requirements for bodies providing audit and certification of supply chain security management systems

[127] ISO/IEC 42010:2007 Systems and software engineering -- Recommended practice for architectural description of software-intensive systems

[128] ISO/IEC Directives — Part 2: Rules for the structure and drafting of International Standards Fifth edition, 2004.

[129] Kazman R., J. Asundi, and M. Klein, Making Architecture Design Decisions: An Economic Approach, SEI-2002-TR-035. Software Engineering Institute, Carnegie Mellon University, 2002.

[130] Kazman R., M. Klein, and P. Clements, ATAM: Method for Architecture Evaluating the Quality Attributes of a Software Architecture. Technical Report CMU/SEI-200-TR004. Software Engineering Institute, Carnegie Mellon University, 2000.

[131] Kelly, T. "Arguing Safety – A Systematic Approach to Managing Safety Cases." Doctorial Thesis – University of York: Department of Computer Science. Sept 1998.

[132] Kelly, T. "Reviewing Assurance Arguments - A Step-by-Step Approach." Workshop on Assurance Cases for Security: The Metrics Challenge, International Conference on Dependable Systems and Networks, 2007.

[133] Kelly, T. and Weaver, R. "The Goal Structuring Notation – A Safety Argument Notation." Workshop on Assurance Cases: Best Practices, Possible Obstacles, and Future Opportunities, Florence, Italy. July 2004.

[134] Ladkin, P. "The Pre-Implementation Safety Case for RVSM in European Airspace is Flawed." [Online Document] 29 Aug 2002 [cited on: 09 Feb 2007] Available HTTP: http://www.rvs.uni-bielefeld.de/publications/Reports/SCflawed-paper.html

[135] Landwehr, Carl, "Computer Security," IJIS vol. 1, pp. 3-13, 2001.

[136] Lautieri, S., Cooper, D., and Jackson, D. "SafSec: Commonalities Between Safety and Security Assurance." Proceedings of the Thirteenth Safety Critical Systems Symposium - Southampton, 2005.

[137] LeBoeuf, Robyn A., and Eldar B. Shafir. "Decision Making." p. 243-266 In [62]

[138] Leveson, Nancy. "A Systems-Theoretic Approach to Safety in Software-Intensive Systems," IEEE Transactions on Dependable and Secure Computing 1, 1 (January-March 2004): 66-86, 2004.

[139] Lipner, Steve and Michael Howard, The Trustworthy Computing Security Development Lifecycle, Microsoft, 2005. Available: http://msdn.microsoft.com/security/ dufault.aspx ?pull=/library/en-us/dnsecure/html/sdl.asp#sdl2_topic8?_r=1

[140] Maguire, Richard. Safety Cases abd Safety Reports: Meaning, Motivation and Management. Ashgate, 2006

[141] McDermid, J. "Software Safety: Where's the Evidence?" 6th Australian Workshop on Industrial Experience with Safety Critical Systems and Software (SCS '01), Brisbane. 2001.

[142] McGraw, Gary. Software Security: Building Security In. Addison Wesley, 2006.

[143] McLean, J. "Security Models." Encyclopedia of Software Engineering (J. Marciniak editor). Wiley 1994.

[144] Meier, J.D., Alex Mackman, Srinath Vasireddy, Michael Dunner, Ray Escamilla, and Anandha Murukan, Improving Web Application Security: Threats and Countermeasures, Microsoft, 2004. Available at: http://download.microsoft.com/download/d/8/c/d8c02f31-64af-438c-a9f4-e31acb8e3333/Threats_Countermeasures.pdf.

[145] Merkow, Mark S. and Jim Breithaupt, Computer Security Assurance Using the Common Criteria, Thompson Delamr Learning, 2005.

[146] Ministry of Defence Standard 00-42 Issue 2, Reliability and Maintainability (R&M) Assurance Guidance. Part 3, R&M Case, 6 June 2003.

[147] Ministry Of Defence. Defence Standard 00-55 (PART 1)/Issue 2, Requirements for Safety Related Software in Defence Equipment Part 1: Requirements, 2 1 August 1997.

[148] Ministry of Defence. Defence Standard 00-55 (PART 2)/Issue 2, Requirements for Safety Related Software in Defence Equipment Part 2: Guidance, 2 1 August 1997

[149] Ministry of Defence. Interim Defence Standard 00-56, Safety Management Requirements for Defence Systems Part 1: Requirements, 17 December 2004.

[150] Ministry of Defence. Interim Defence Standard 00-56, Safety Management Requirements for Defence Systems Part 2: Guidance on Establishing a Means of Complying with Part 1, 17 December 2004.

[151] Moore, A., Klinker, E., and Mihelcic, D. "How to Construct Formal Arguments that Persuade Certifiers." Industrial Strength Formal Methods. Academic Press. 1999.

[152] National Aeronautics and Space Administration (NASA) Software Assurance Guidebook (NASA-GB-A201). Available at http://satc.gsfc.nasa.gov/assure/agb.txt.

[153] National Computer Security Center. A Guide to Understanding Covert Channel Analysis of Trusted Systems, NCSC-TG-030, NCSC, November 1993. Available at: http://www.radium.ncsc.mil/tpep/process/overview.html

[154] National Offshore Petroleum Safety Authority. "Safety Case Guidelines." September 2004.

[155] National Offshore Petroleum Safety Authority. Guideline: Construction Safety Cases. 30 Aug 2006.

[156] National Offshore Petroleum Safety Authority. Safety case approach. [Online Document] [cited on: 13 Feb 2007] Available HTTP: http://www.nopsa.gov.au/safety.asp

[157] National Research Council (NRC) Computer Science and Telecommunications Board (CSTB). Cybersecurity Today and Tomorrow: Pay Now or Pay Later. National Academies Press, 2002. Available at http://darwin.nap.edu/books/0309083125/html.

[158] National Security Agency, The Information Systems Security Engineering Process (IATF) v3.1. 2002.

[159] Naval Research Laboratory, Handbook for the Computer Security Certification of Trusted Systems, US Naval Research Laboratory, 1995

[160] NDIA System Assurance Committee. Engineering for System Assurance version .90. National Defense Industrial Association (USA), 2008

[161] NIST. Federal Information Processing Standards Publication (FIPS PUB) 200: Minimum Security Requirements for Federal Information and Information Systems. March 2006. Available at http://csrc.nist.gov/publications/fips/fips200/FIPS-200-final-march.pdf.

[162] NIST. NIST SP 800-33, Underlying Technical Models for Information Technology Security, December 2001

[163] NIST. NIST Special Publication 800-27: Engineering Principles for Information Technology Security (A Baseline for Achieving Security). Revision A, June 2004. Available at http://csrc.nist.gov/publications/nistpubs/800-27A/SP800-27-RevA.pdf.

[164] OPEN Process Framework. Safety Cases. [Online Document] 5 Nov 2005. [cited on: 13 Feb 2007] Available HTTP: http://www.opfro.org/index.html?Components/WorkProducts/SafetySet/SafetySet.html~Contents

[165] OPSI. The Offshore Installations (Safety Case) Regulations 2005. [Online Document] 9 Nov 2005. [cited on: 13 Feb 2007] Available HTTP: http://www.opsi.gov.uk/si/si2005/20053117.htm

[166] Park, J., Montrose, B., and Froscher, J. "Tools for Information Security Assurance Arguments." DARPA Information Survivability Conference & Exposition II, 2001. DISCEX '01. Proceedings. 2001.

[167] Petroski, Henry. Design Paradigms. Cambridge University Press, 1994.

[168] Prasad, D., Dependable Systems Integration using Measurement Theory and Decision Analysis, PhD Thesis, Department of Computer Science, University of York, UK, 1998.

[169] Prieto -Diaz, Ruben, Common Criteria Evaluation Process, Commonwealth Information Security Center Technical Report CISC-TR-2002-003, 2003. http://www.jmu.edu/cisc/research/publications/CCevaluationProcesTR03-5.pdf

[170] PSM Safety & Security TWG. "Security Measurement." Nov 2004.

[171] Pullum, L. L. Software Fault Tolerance, Artech House, 2001.

[172] Randell, B., and Koutny, M. Failures: Their Definition, Modelling and Analysis. School of Computing Science, Newcastle University CS-TR NO 994, Dec 2006 Randell, B., and Rushby, J. M. Distributed Secure Systems: Then and Now. CS-TR No 1052 School of Computing Science, Newcastle University, Oct 2007

[173] Rechtin, E. Systems Architecting of Organizations: Why Eagles Can't Swim. Boca Raton, FL: CRC Press, 2000.

[174] Redwine, Samuel T., Jr. (Editor). Software Assurance: A Guide to the Common Body of Knowledge to Produce, Acquire, and Sustain Secure Software Version 1.1. US Department of Homeland Security, September 2006.

[175] Redwine, Samuel T., Jr. “The Quality of Assurance Cases” Workshop on Assurance Cases for Security: The Metrics Challenge, International Conference on Dependable Systems and Networks, 2007.

[176] Redwine, Samuel T., Jr., and Noopur Davis (Editors). Processes for Producing Secure Software: Towards Secure Software. vols. I and II. Washington, D.C.: National Cyber Security Partnership, 2004. Available at http://www.cigital.com/papers/download/secure_software_process.pdf.

[177] Riguidel, Michel, Gwendal Le Grand, Cyril Chiollaz, Sved Naqvi, Mikael Formanek, “D1.2 Assessment of Threats and Vulnerabilities in Networks”, Version 1.0. European Union Security Expert Initiative (SEINIT), 31 August 2004. Available at http://www.seinit.org/documents/Deliverables/SEINIT_D1.2_PU.pdf

[178] Ross, Karol G., Jennifer L. Shafer, and Garry Klein. “Professional Judgements and ‘Naturalistic Decision Making’.” p. 403-420 in [47].

[179] Ross, Ron et al. Recommended Security Controls for Federal Information Systems, NIST Special Publication 800-53, Feb. 2005. Available at http://csrc.nist.gov/publications/nistpubs/800-53/SP800-53.pdf. and its Revision 1 Draft available at Available at http://www-08.nist.gov/publications/drafts/800-53-rev1-ipd-clean.pdf.

[180] SAE JA1000 Reliability Program Standard, SAE International, June 1998. Available at http://www.sae.org

[181] SafSec Project. SafSec Methodology: Guidance Material: Integration of Safety and Security. Available at: http://www.praxis-his.com/safsec/safSecStandards.asp.

[182] SafSec Project. SafSec Methodology: Standard: Integration of Safety and Security. Available at: http://www.praxis-his.com/safsec/safSecStandards.asp.

[183] Saltzer, J. H. and M. D. Schroeder, “The protection of information in computer systems,” Proceedings of the IEEE, vol. 63, no. 9, pp. 1278-1308, 1975. Available on-line at http://cap-lore.com/CapTheory/ProtInf/

[184] Seminal Papers - History of Computer Security Project, University of California Davis Computer Security Laboratory. Available at: http://seclab.cs.ucdavis.edu/projects/history/seminal.html

[185] Serene. “Safety argument.” [Online Document] [cited on: 13 Feb 2007] Available HTTP: http://www2.dcs.qmul.ac.uk/~norman/SERENE_Help/sereneSafety_argument.htm

[186] Severson, K. “Yucca Mountain Safety Case Focus of NWTRB September Meeting.” United States Nuclear Waste Technical Review Board. Aug 2006.

[187] Sieck, Winston R., and Gary Klein. “Decision making.” p. 195-218 in [45].

[188] Software and Systems Engineering Vocabulary (sevocab). Available at www.computer.org/sevocab/

[189] Software Engineering Institute. Assurance Case and Plan Preparation. [Online Document] [cited on: 26 Feb 2008] Available at http://www.sei.cmu.edu/pcs/acprep.html.

[190] Sommerville, Ian. Software Engineering, 8th ed., Pearson Education, 2006.

[191] Stoneburner, Gary, Hayden, Clark and Feringa, Alexis. Engineering Principles for Information Technology Security (A Baseline for Achieving Security), Revision A, NIST Special Publication 800-27 Rev A, June 2004.

[192] Storey, Neil. Safety-Critical Computer Systems. Addison Wesley, 1996.

[193] Strunk, E. and Knight, J. "The Essential Synthesis of Problem Frames and Assurance Cases." IWAAPF'06, Shanghai, China. May 2006.

[194] Swiderski, F. and W. Snyder, Threat Modeling, Microsoft Press, 2004.

[195] U.S. NRC. "Quality Assurance Case Studies at Construction Projects."

[196] UK CAA. CAP 670 Air Traffic Services Safety Requirements. UK Civil Aviation Authority Safety Regulation Group, 12 June 2003 amended through 30 June 2006.

[197] UK CAA CAP 730 Safety Management Systems for Air Traffic Management A Guide to Implementation. UK Civil Aviation Authority Safety Regulation Group, 12 September 2002.

[198] UK CAA CAP 760 Guidance on the Conduct of Hazard Identification, Risk Assessment and the Production of Safety Cases For Aerodrome Operators and Air Traffic Service Providers, 13 January 2006.

[199] Vanfleet, W. Mark and R. William Beckwith, Dr. Ben Calloni, Jahn A. Luke, Dr. Carol Taylor, and Gordon Uchenick. "MILS:Architecture for High Assurance Embedded Computing," Crosstalk. August, 2005.

[200] Viega, J., The CLASP Application Security Process, Secure Software, 2005. Available at http://www.securesoftware.com

[201] Viega, J., The CLASP Application Security Process, Secure Software, 2005. Available at http://www.securesoftware.com.

[202] Viega, John, and Gary McGraw, Building Secure Software: How to Avoid Security Problems the Right Way, Reading, MA: Addison Wesley, 2001.

[203] Walker, Vern R. "Risk Regulation and the 'Faces' of Uncertainty," Risk: Health, Safety and Environment. p. 27-38, Winter 1998

[204] Ware, Willis H. Security Controls for Computer Systems (U): Report of Defense Science Board Task Force on Computer Security, The RAND Corporation, Santa Monica, CA (Feb. 1970).

[205] Weaver, R. "The Safety of Software – Constructing and Assuring Arguments." Doctorial Thesis – University of York: Department of Computer Science. 2003.

[206] Weaver, R., Fenn, J., and Kelly, T. "A Pragmatic Approach to Reasoning about the Assurance of Safety Arguments." 8th Australian Workshop on Safety Critical Systems and Software (SCS'03), Canberra. 2003.

[207] Whittaker, J. A. and H. H. Thompson. How to Break Software Security: Effective Techniques for Security Testing. Pearson Education, 2004.

[208] Williams, J. and Schaefer, M. "Pretty Good Assurance." Proceedings of the New Security Paradigms Workshop. IEEE Computer Society Press. 1995.

[209] Williams, Jeffrey R. and George F. Jelen, A Framework for Reasoning about Assurance, Document Number ATR 97043, Arca Systems, Inc., 23 April 1998.

[210] Yates, J. Frank, and Michael D. Tschirhart. "Decision-Making Expertise." p. 421-438 in [47]

[211] Yee, Ka-Ping , "User interaction design for secure systems," In Proceedings of the 4th International Conference on Information and Communications Security, Springer-Verlag, LNCS 2513, 2002.

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

ISO/IEC JTC1/SC7 /N4201

2009-01-21

Document Type	Comments Disposition Report
Title	Comments disposition report, PDTR 15026-1.2, Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary
Source	WG7
Project	15026-1
Status	Final
Reference	N4110, N4199
Action ID	FYI
Distribution	AG
No. of Pages	6
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

Document Type	PDTR Comment disposition report
Title	ISO/IEC PDTR comment disposition
Source	ISO/IEC JTC 1/SC 7/WG xx Convener
Status	Final
Reference	ISO/IEC JTC 1/SC 7 N1153
Action ID	FYI or ACT

Document information

Balloted document

ISO/IEC document code	ISO/IEC
Document title	Systems and software engineering — Systems and sof
Document status	PDTR
WG document code	N1104
SC7 document code	
JTC1 document code	

Ballot summary

Ballot Type	PDTR
Ballot summary document code	ISO/IEC JTC 1/SC 7 N1153
Ballot status	Approved

Comment disposition report

WG document code	ISO/IEC JTC 1/SC 7/WG 7 N1153
Report date	11/7/2008
Report status	Final
Generated by	Comment database Version 4.2.1 on 1/15/2009 8:52:40 PM

Report selection criteria

National Body

Category

Clause

Outcome

No outcome

Classification

Issue

Status

Filter 1

Filter 2

Comments selected: 30

ID	Cat	Clause	Para	Comment and rationale	Proposed text	Outcome	Disposition
GBR-1	GT			UK vote on Part 1 is to 'Approve with comment' . UK request that our previous comments (on the pre-PDTR, supplied to the Editor prior to the Berlin SC7 Plenary) shall be addressed.		A	Comments will be reviewed.
JPN-4	GE			A way of the numbering of listed items should be described with unified manner and be easy to read.□The followings are examples which are extremely difficult to read:□- a), b)… following the forth layer "9.2.2.2";□- 1), 2)… following the second layer "10.2".	For example, alphabetical a), b)… should be following the third layer clause number "0.0.0".□	AIP	Unclear what suggestion intends. However, could be resolved as part of general edit.
JPN-5	GT			It may be helpful to send questionnaires to each of NBs, which are asking the extent of completion of this drafted document (Draft Maturity Level). Such way was used when the ISO/IEC 12207 and 15528 were under developed.		AIP	This may be helpful.
IE3-1	TL	10		Clause 10 needs to be modified to reflect the most current thinking about Part 3		A	
IE3-2	E	10		Clause 10 should have figures/diagrams to help reader understand		A	
USA-20	E	9.1		There are two sections numbered as 9.1.	Correct numbering.	A	
JPN-2	TH	9.1	2	When a supplier is contracted with an acquirer to develop system or software, acquirer's requirements specifications are often provided ambiguously. So, following (a) and (b) become important. □□a) Acquirer is responsible to provide specified integrity level and assurance requirements of the target system unambiguously enough to determine the assurance case. (See Attachment No.1-1)□b) When integrity level is "high", acquirer and supplier agree with tradeoffs of the investment and cost. (See Attachment No.1-2)	Add followings to 9.1:□□When a supplier is contracted with an acquirer to develop system or software, followings shall be ensured:□□a) Acquirer is responsible to provide specified requirements unambiguously enough to determine the integrity level and the assurance claims of the target system. □b) When integrity level is "high", acquirer and supplier agree with tradeoffs of the investment and cost.	R	That something is true within the assurance case is within the scope of this IS Part 2. Who originally supplies the contents is not within scope. This is consistent dispostion of JP-6 on Part 2.
JPN-1	GT	9.2.3	Figure2	Japan agrees that selecting the top-level claim is not within the scope of ISO/IEC 15026, as described in 9.2.1 Introduction. At a point of view of security or functional safety, is the structure of claim shown in Figure2 consistent and appropriate?□□See JP-3, the comment for ISO/IEC 15026 Part2.	It is necessary to liaise with the other SCs who handle the security, functional safety etc., so that ISO/IEC 15026 and its properties are consistent.	AIP	Liasing could be benefitial.

ID	Cat	Clause	Para	Comment and rationale	Proposed text	Outcome	Disposition
USA-1	GT	9.3.2.1,□lines 432-435,□and throughout		Generally wordy and tough to read smoothly, refer to following sentence as a particularly unruly example, "An argument needs to argue that what supports it is relevant to supporting the claim and that it meaningfully combines what supports it into support for the claim whose meaning and uncertainty reflect those of what supports it and the nature of the argument." (from 9.3.2.1)□□Another example is in lines 432-435.□ □It is unclear what the author intended in this instance.	Cannot provide replacement text, since the author's intended message is not clear. Change this specific instance to make it more readable and look for other overly wordy sentences.	A	General edit will be conducted.
ISR-1	TH	ALL		Don't accept rejection of comments on previus ballot. Some standards should be divided into parts and some shouldn't. This one shouldn't. Partial conformance, for example, sounds horrific.		OBE	
USA-21	GT	ALL		This document has too much redundant information. For example clause 9.1 goes into a lot of detail that is then covered again in the subsequent clauses. In general, the document is too large with too much tutorial detail.	Review for areas to remove redundancy. This document needs a significant amount of rewriting.	A	
JPN-3	GT	Bibliography		There are documents which are impossible to access or refer from user of any country.	It is possible that they are given in Bibliography, but such texts in the normative clauses of this document should be removed or moved to informative annexes, that refer to documents which are impossible to access or refer from user of any country.	R	No normative clauses exist in this document.
USA-2	GT	line 270		Definition of uses undefined set of terms; i.e., "Justified confidence". Need the concept of adequate or sufficient confidence.	Add definitions for "Confidence" and "Justifed Confidence".	AIP	Partially accept. Concept of adequate or sufficient confidence has a place in report. Offical definitions for confidence and justified confidence are unneeded as common definition suffices.
USA-3	E	line 283		What is meant by 'clearly' link?	Clarify the phrase "clearly link".	A	Accept. Replace "clearly" with "logically".
USA-5	TL	line 344		Definiton of uncertainty as lack of certainty adds little to the standard as it stands. The term is used in the same way as in common English usage. Clause 7.4 provides more insight into its use in this standard. It also already covers what is in the note of the definition.	Delete the definition.	R	Reject. While this is layman's usage, several communities give it specialized definitions and for users of the standards from these communities the definition needs to be explicit and stated "offically".

ID	Cat	Clause	Para	Comment and rationale	Proposed text	Outcome	Disposition
USA-7	E	line 437	Page 5, para 5.3, line 437	Typo, clarity should be clarify	Change to "clarify"	A	
USA-8	E	line 439		Shareholder should read stakeholder	stakeholders	A	
USA-10	E	line 460		States: "Across the life cycle systems or software can have a number of stakeholders who might or be perceived as affect or being affected by the product including through product-related activities." Awkward sentence structure makes it hard to understand.	Improve sentence structure and grammar.	A	
USA-11	TL	line 533	line 533, and on…	Suggest including statistical confidence as a highly used measure of assurance	Add the following as an example for the second bullet:□"e.g., using statistical confidence as a measure of assurance".	A	Will add material on statistical confidence somewhere in report.
USA-12	E	line 551		Definition of assurance case - make consistent with earlier definition.	Re-phrase to make consistent.	A	
USA-13	TL	line 554	after line 554	Suggest including example of the reliability case methodology as a type of assurance case, a reference to SAE JA-1000 would be recommended	* A type of assurance case used regularly is the Reliability, Availability and Maintainability or RAM Case. This approach, documented in SAE JA1000 prescribes three objectives for the assurance of reliability; (1) demonstrate understanding of requirements, (2) develop and execute a plan to achieve those requirements, and (3) provide progressive assurance and evidence of compliance	A	
USA-14	TL	line 566		This statement avoids the question of how to decide when to use an assurance case. Should the standard provide guidance on this?	General considerations should be added to aid the selection of an assurance case. Examples include risk assessments, customer direction, organizational practices, industry regulatory requirements, …	A	
USA-15	GE	line 646		Example: Language of this standard is rather discursive and statements are qualified and hedged - needs tightening up	Replace the whole first paragraph with "The purpose of an assurance case is to provide assurance about product or service properties to parties not closely involved in the associated technical development or service processes, particularly for purposes of audit."	A	Accept general comment, but replacement text may be different.
USA-17	E	line 690		The objectivity of the assurance case depends on the objectivity of the cited evidence; an assurance case merely gives better visibility of the arguments used. 'Justified degree of confidence' is confusing.	Add definitions for "Confidence" and "Justifed Confidence".	A	Edit to improve but not add definitions. See comment US 2.

ID	Cat	Clause	Para	Comment and rationale	Proposed text	Outcome	Disposition
USA-18	TL	line 693		This para argues that the assurance case is central to rational use of products where uncertainty and consequence are serious concerns. Not sure this claim has been justified.	Not clear that this paragraph is necessary, as much of it is redeundant. Delete the paragraph.	A	Editors consider and edit as appropriate.
USA-4	GE	lines 291, 302, and throughout		In line 291, the note is not numbered, even though it is the second note for the definition. In line 302, there is a second Note 2 for this definition.	Go through the document and fix all incorrect Note labels.	A	
USA-6	TL	lines 403-425		We don't understand the meaning of the subclause title and the lead-in paragraph. Without this context, we don't understand the rest of this clause either.	No replacement text can be provide, since we don't understand the intent of the clause. Clarify the clause.	A	Accept. Clarify purpose of subclause.
USA-9	E	lines 446-448		An unnecessary break in the sentence occurs that makes the paragraph hard to read.	Remove the extra line breaks.	A	
USA-16	TL	lines 651-661		Lines 651 through 661 are redundant with the defintiion in Clause 3, as well as providing a little clarification.	Incorporate the clarifying information into the definition and delete here.	A	Accept incorporated any clarifying information in Notes related to definition. Some coverage of assurance case is neeeded here in order for the report to flow properly.□Category added as a TL during database loading
USA-19	TL	lines 718-740		This text is both confusing and somewhat out of scope for this document.	Delete the text.	A	Edit to make information clear and review for redundancy.

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

ISO/IEC JTC1/SC7 /N4203

2009-01-23

Document Type	Abstract
Title	Abstract, DTR 15026-1, Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary
Source	WG7
Project	15026-1
Status	Final
Reference	N4202
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

Abstract ISO/IEC TR 15026-1

This Technical Report provides aid to users of International Standard ISO/IEC 15026 on systems and software assurance. This Technical Report defines terms. It establishes a basis for shared understanding of concepts and principles central to the International Standard including an extensive and organized set of concepts and their relationships. These cover general concerns, assurance cases, integrity levels, and assurance in the lifecycle.

G8 Explanatory Report

EXPLANATORY REPORT	DTR 15026-1 Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary
ISO/IEC JTC 1/SC 7 **N 4202**	
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: 2008-10-26
by postal ballot initiated on: 2008-07-25
P-members in favour: 15 P-members voting against: 3 P-members abstaining: 1 P-members who did not vote: 8

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