

Citation for this paper is as follows:

Henderson-Sellers, B., Gonzalez-Perez, C., McBride, T. and Low, G.C. 2014, An ontology for ISO software engineering standards: 1) Creating the infrastructure, *Computer Standards and Interfaces*, **36(3)**, 563-576

An Ontology for ISO Software Engineering Standards: 1) Creating the Infrastructure

Brian Henderson-Sellers^{1*}, Cesar Gonzalez-Perez², Tom McBride¹ and Graham Low³

¹Faculty of Engineering and Information Technology, University of Technology,
Sydney, PO Box 123, Broadway, NSW 2007, Australia

²Institute of Heritage Sciences (Incipit)
Spanish National Research Council (CSIC), Santiago de Compostela, Spain

³School of Information Systems, Technology and Management,
University of New South Wales, Sydney, NSW 2052, Australia

¹[brian.henderson-sellers,tom.mcbride}@uts.edu.au](mailto:{brian.henderson-sellers,tom.mcbride}@uts.edu.au), ²cesar.gonzalez-perez@incipit.csic.es,
³g.low@unsw.edu.au

*Corresponding author. Tel +61 2 9514 1687

Abstract. Software engineering standards developed under the auspices of ISO/IEC JTC1's SC7 have been identified as employing terms whose definitions vary significantly between standards. This led to a request in 2012 to investigate the creation of an ontological infrastructure that aims to be a single coherent underpinning for *all* SC7 standards, present and future. Here, we develop that necessary infrastructure prior to its adoption by SC7 and its implementation (likely 2014). The proposal described here requires, firstly, the identification of a single comprehensive set of definitions, the Definitional Elements Ontology (DEO). For the scope of an individual standard, only a subset of these definitional elements will be needed. Once configured, this definitional subset creates a Configured Definitional Ontology or CDO. Both the DEO and the CDO are essentially foundational ontologies from which a domain-specific ontology known as a SDO or Standard Domain Ontology can be created. Consequently, all such SDOs are conformant to a CDO and hence to the single DEO thus ensuring that all standards use the same ontological base. Standards developed in this fashion will therefore be not only of a higher quality but also, importantly, interoperable.

Keywords: conceptual modelling; international standards; ontologies

1. Introduction

The International Organization for Standardization (ISO), based in Geneva, is responsible for the creation and maintenance of a large suite of international standards (IS) and technical reports (TR) for software engineering (SE). However, the semantics of the terms used in these various SE standards can often be contradictory, or at best misaligned, across pairs of similarly-focussed standards. For example, within the SC7 (the sub-committee responsible for software engineering standards) community, Rout [1] analyzed a number of standards, term by term, identifying by number the standards in which these terms appear together with their (disparate) definitions. Other ISO work, outside of SC7, has led to the publication of ISO 704¹, currently in its third edition (2009): "Terminology work – principles and methods".

The early 2000s saw SC7 standards being developed increasingly in 'stovepipes', leading to the realization that these various standards need to be consistent with each other in terms of terminology, structure and semantics, resulting in significant discussions be-

¹ All ISO standards cited are listed separately at the end of the paper

tween working groups (WG) within SC7 to resolve terminology differences e.g. WG6 and WG13 needed to resolve 'metric' and 'measure' e.g. see discussion in [2]. On a larger scale, initial conceptual differences, possibly among other things, led to quite different process standards for software (ISO/IEC 12207) and systems (ISO/IEC 15288). More recently, SC7 (notably WG7) has attempted to reconcile definitions within a harmonization project to align ISO/IEC 12207 and ISO/IEC 15288, requiring considerable effort (see also [3]). Also over the last several years, a new International Standard (ISO/IEC 24744) has been created specifically to create a set of definitions of process-focussed terms and their interrelationships, effectively creating a foundational ontology [4].

At the beginning of 2012, McBride et al. [5] produced a document for the 2012 SC7 Plenary meeting (Korea) entitled "The growing need for alignment" (also as a PowerPoint™ presentation to the Strategic Planning meeting of SC7 on 19-20 May 2012). The themes presented there included:

- Increasing recognition that standards are becoming multi-disciplinary and that there is no guidance for a new team when building a standard to ensure it is compatible with other SC7 standards. They need guidance for "What kind of standard do you want to develop", especially, guidance about the type, purpose and applicability.
- Clashes of terminology and subtle clashes in semantics observed in current standards.
- The recent impacts on software engineering standards due to the enactment of external legislation such as Sarbanes-Oxley Act [6].

These factors suggest that:

- There is a need to move from serendipitous knowledge of such problems to organizational (SC7-level) solutions.
- Each standard should be tightly focussed e.g. requirements standards are different from guidance standards.
- An ontology/taxonomy should be produced, in particular using ISO 704:2009 – Terminology work – principles and methods.

Resulting from the discussion of these proposals at the May 2012 meeting of SC7, a Study Group was created, chaired by the first author of this paper and charged with investigating the potential utility of ontologies for rationalizing SC7's suite of software engineering International Standards.

SC7 is not alone in addressing concerns on non-interoperability [7]. Almost any successful organization will gradually introduce more products and eventually face the need to rationalize them in order to present potential customers with a suite of products that can work together and that do not contradict or compete with each other unnecessarily. Organizations also find it necessary to rationalize their production methods in order to avoid supporting unique development and production methods, tools and infrastructure. For SC7, a rationalised product set is more important than rationalised production methods. Even so, JTC1 (Joint Technical Committee 1) has found it necessary to impose a template standard, a standard form for ballot comments and a uniform balloting system.

Like all products, standards are frequently used in combinations. Organizations want to use high level standards for systems development, for example, along with standards

that expand on specific processes such as software maintenance or risk management. Organizations also want to combine standards from different sources; quality management with software development or CoBIT with software development and related processes. Harmonious combination of standards is aided when the standards start with consistent concepts and terms. This is not significantly different from organizations sourcing product components from different suppliers. They too need agreements on the basics like voltages, measurement scales, thread types and screw sizes.

Muddling through, resolving conceptual differences as they are noticed will not suffice in a future where standards are used in combinations, sometimes from disparate sources.

Software, systems, IT Service management, IT governance are all sufficiently different domains, such that considerable effort is needed to avoid mutual misunderstanding of fundamental concepts. That there is now considerable effort being made to split the “management” related standards off into another JTC1 committee argues that this effort was either not made or was insufficient. Splitting into another JTC1 committee won’t solve the problem of mutual misunderstanding but will increase the difficulty of developing standards that can work together.

SC7 needs to manage its strategic direction, as does any organization. The clarity and effectiveness of that strategy depends, in part, on how clearly the conceptual foundations of the different families of standards are articulated, how well the relationships are understood between those families to other standards within SC7 and standards external to SC7. This is much easier to achieve when the concepts and terminology are universally agreed.

In this paper, which offers a significant extension to the ideas proffered in [8], we present the research-oriented results of the material evaluated by the members of this Study Group and the proposal that was made to the SC7 overseeing committee (SWG5) at the SC7 Plenary meeting in Montréal in May 2013 and the subsequent revisions to create its “Second Report” (circulated to committee members July 2013). At that May 2013 plenary meeting, further work was also commissioned by SWG5 in order to create a work plan prior to the development of the first report into a New Work Item Proposal (NWIP) at the next international SC7 meeting.

Our overall hypothesis is simply that SC7 standards, as an interdependent suite of artefacts, could benefit by the application of conceptual modelling and ontological organization.

In summary, the problems to be addressed in this paper and the proposed solutions adopted by the Study Group and SWG5 are:

- (i) construction of clear, unambiguous and comprehensive definitions of all SC7 terminology;
- (ii) conformance of existing and new standards to this agreed ontological description of terminology;
- (iii) categorization of existing standards and their relationship to (i) and (ii).

The approach taken is that of a combination of conceptual modelling e.g. [9] and ontology engineering e.g. [10]. In Section 2 of this paper, we first identify from the research literature five possible flavours of ‘ontology’ that might be usefully transformed into SC7 standards i.e. research-underpinned technology transfer (each of these five being discussed in detail in five subsections). Section 3 discusses how these ideas might be implemented in the context of ISO’s SC7 committee and its request in May 2013 for a detailed work plan. Section 4 discusses how this proposal can be maintained in future years and its influence on how new standards will need to be developed; whilst Section 4 gives both our recommendations (made to SC7) together with our overall conclusions.

1.1 Current situation for SC7 standards

SC7 already has a vocabulary (ISO/IEC 24765 SEVOCAB), a definition of process (TR 24774), several process reference models (ISO 12207, ISO 15288, ISO 19770-1, ISO 29119-2, ISO 20000-4) and several detailed processes (ISO 14764, ISO 16085, ISO 15939, etc.). In the example shown in Figure 1, TR 24774 is specified using terms defined in the ISO vocabulary. SC7 process reference model standards, e.g. ISO 12207, ISO 15288, ISO 19770-1, define or describe processes using the terms and structure defined in TR 24774 and there are several standards, of which ISO 15939 is an example, that are an expanded description of a process used in or referenced in SC7 process reference models, e.g. ISO/IEC 15939. This structure has simply emerged, illustrating that a hierarchy of concepts, configured concepts, standards based on those definitions and instantiations of those defined processes seems a natural thing to do.

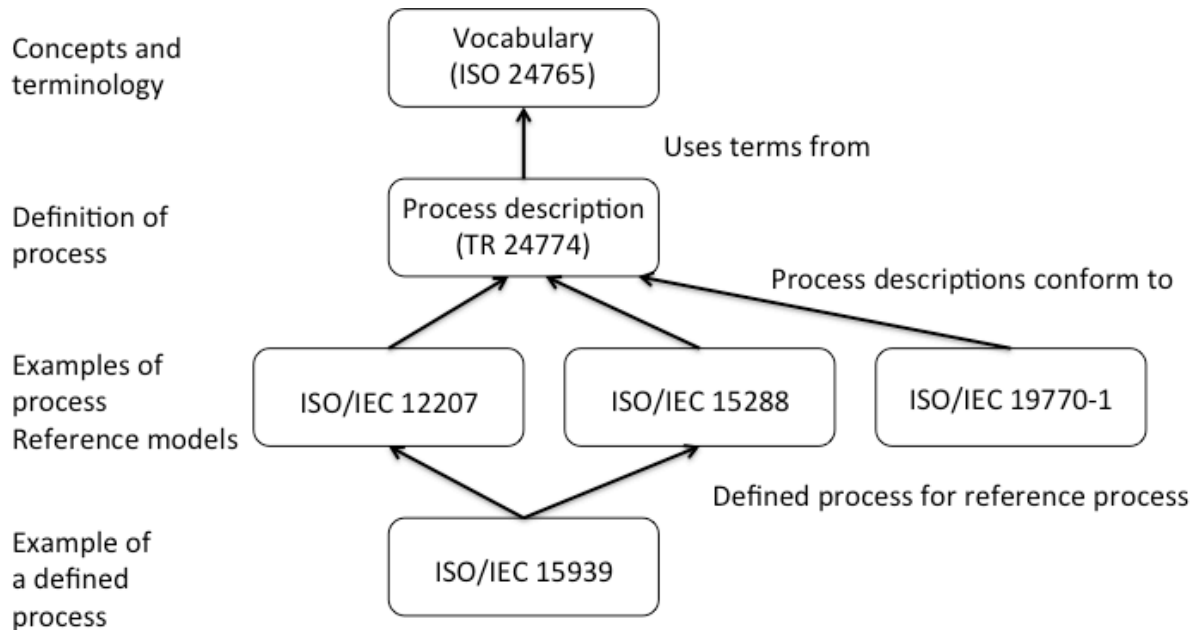


Figure 1 SC7 has some information regarding vocabulary, process definitions, process reference models and defined processes.

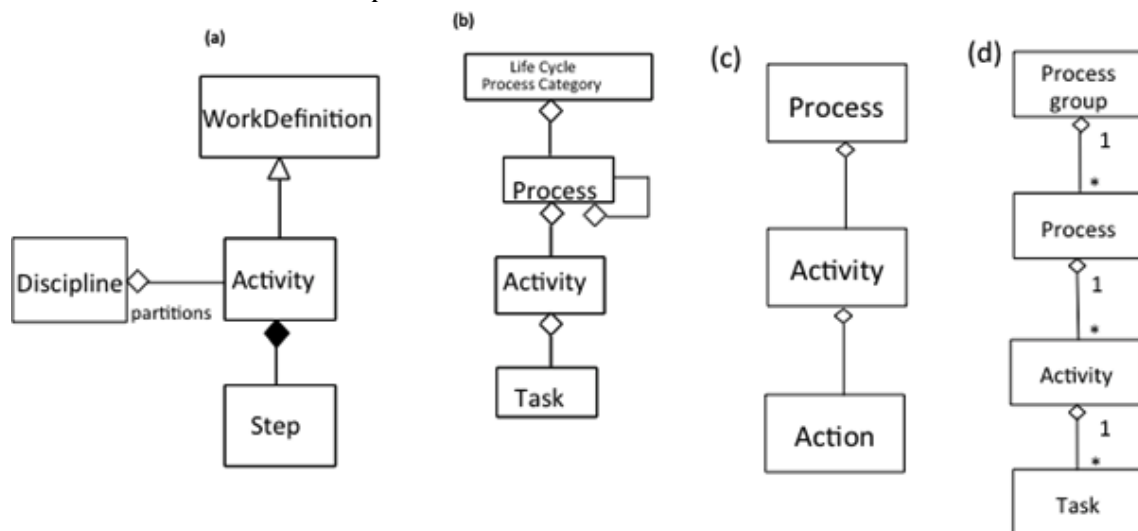
ISO 24765 (SEVOCAB) was not intended to be a collection of precise definitions, simply a collecting together of existing definitions. Yet SC7 needs to know which definitions should apply in its domain. Additionally, SC7 needs a way to adopt a precise definition to the particular circumstances. For example, a process may be defined as “set of interre-

lated or interacting activities that transforms inputs into outputs” but that definition needs to extend to other attributes of processes in order to accommodate the interests of process assessment.

TR 24774 has guidelines for process descriptions that do not try to include every conceivable interest in processes. So far, TR 24774 accommodates process performance (e.g. ISO 15288) and process assessment (ISO 15504/33002) but not process governance. Nor should it. These divergent needs will come with time and might extend TR 24774 or might result in another set of guidelines for process description. In either case, there needs to be consensus about just what a process is, and precisely what the different attributes of a process are.

The hierarchy exists but the problems described earlier attest to its limitations.

Secondly, SC7 is increasingly being presented with PASs (Publicly Accepted Standards), many from the Object Management Group (OMG). Yet, there is no guarantee that the vocabulary (and ontology) utilized in a PAS will be consistent with that of SC7. For example, Figure 2 contrasts the implicit definitions of terms in OMG’s SPEM standard (Software Process Engineering Metamodel: [11]) and similarly focussed standards of SC7 (see also discussion of this diagram in [12]). It is clearly seen that these are different across both SPEM and the process-focussed standards of SC7.



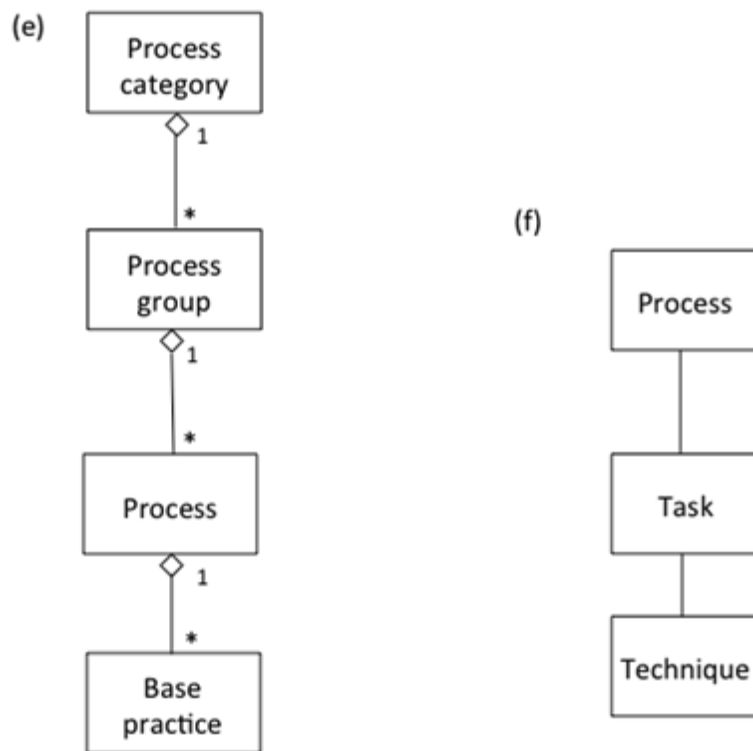


Figure 2 Comparison of the portions of the underlying ‘ontology’ for various process-focussed standards: (a) SPEM V1 [11], (b) ISO/IEC 12207, (c) ISO/IEC 15288 (2002 version), (d) ISO/IEC 15288 (current version), (e) ISO/IEC 15504 and (f) OOSPICE [13] (after [3] figure 5, ©2012) (With kind permission of Springer Science+Business Media.)

2. Proposals: five different modelling approaches

In order to develop an appropriate and ontologically valid infrastructure to underpin all of SC7’s SE standards, we first identified five distinct areas where conceptual modelling and ontologies might be helpful for reorganizing SC7 standards. It is important to note that these five areas relate to different domains, as illustrated in Figure 3: (i) the definitional domain (ontologies A, B and E below), such ontologies are often called foundational ontologies or upper-level ontologies [14, 15]; (ii) the modelling domain (ontologies C and D); and (iii) the practice domain – wherein practice-based ontologies are the enactment of a process-focussed ontology from the modelling domain. Note that in Figure 3, some existing SC7 standards have been tentatively allocated to these different domains.

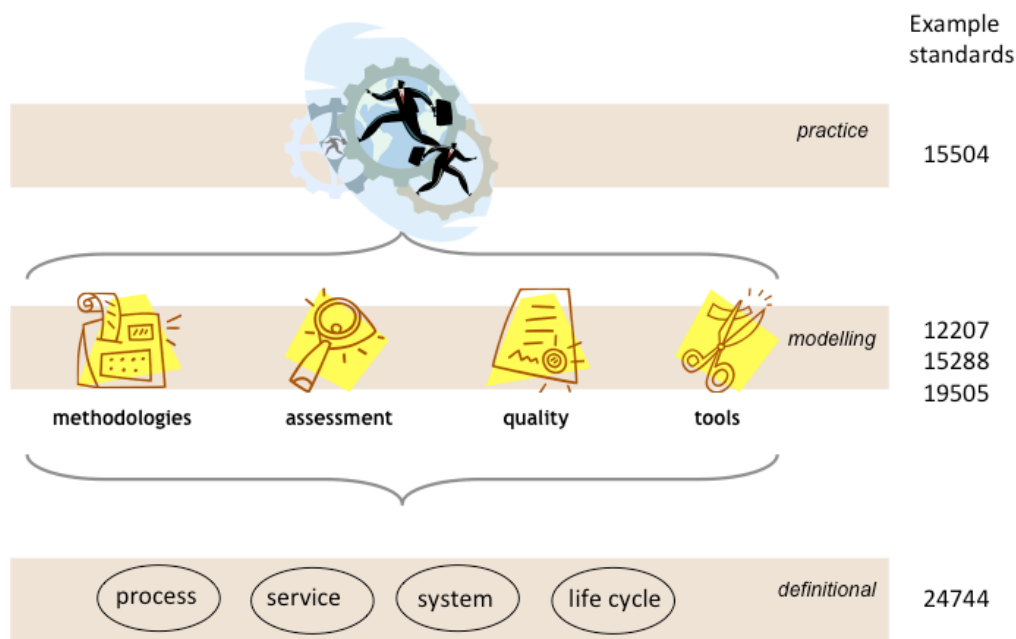


Figure 3 The proposed three level architecture (adapted from that originally created for ISO/IEC 24744 but with slightly modified names for the three domains)

We now consider each of these 5 originally proposed areas in detail.

(A) 'Definitional Elements Ontology' (DEO): although not an ontology in the strictest sense, the DEO provides definitions of individual concepts/terms, together with constraints that dictate how they must be related and hence configured in some future definitional standard (see below). At present, SC7 relies on each working group defining the terms it needs. The SC7 vocabulary (ISO 24765) is a collection of terms from SC7 standards that are frequently duplicated but slightly different in expression or meaning and has been identified as a valuable input to creating the standardized DEO together with the metamodel/ontology for software engineering specified in ISO/IEC 24744 (itself currently a domain-specific CDO – see below).

(B) 'Configured Definitional Ontology' (CDO): a specific configuration of the elements in the DEO – either as a standalone standard (à la ISO/IEC 24744) or as an intermediate template for the creation of a 'Standard Domain Ontology' (Type C – see below) a.k.a. an International Standard (IS). A CDO gives structure and order to the concepts used and necessary in a specific domain. For example, ISO/IEC 15504 needs to define a process in terms of its purpose and outcomes. Without those two concepts, process assessment could not be consistent or rigorous. It happens that processes within SC7 are comprised of more than their purpose and outcomes and, as the impact of other domains become evident within SC7, a larger configured definitional ontology becomes more important. An example of such an 'other domain' is that of governance. The requirements for auditable governance will emerge over time and these, like that of quality, will introduce new activities, new work products, and possibly new relationships between different processes. All of these may throw up new concepts that need to be evaluated for possible addition to the DEO.

A CDO could itself be standardized within SC7 (the only example to date being ISO/IEC 24744). However, it is more common in SC7 to standardize an instance of a CDO. Since CDOs consist of concepts, an instance of a CDO would be composed of occurrences of these concepts; the latter being an SDO (see below). For example, a CDO may contain the Task concept, for which sample instances might be “Unit test Java class” and “Write use case” (as seen, for instance, in ISO/IEC 12207).

(C) ‘Standard Domain Ontology’ (SDO): a number of instances conforming to the CDO template. These follow a specific configuration as determined by the CDO. A Standard Domain Ontology is highly specific to one particular situation e.g. process management, services, governance (à la ISO/IEC 12207, which is a SDO that is a ‘process model’ for the domain of processes, activities and tasks). A number of SC7 working groups are developing process reference models, e.g. ISO/IEC 19770-1 (Asset management), ISO/IEC/IEEE DIS 29119-2 (Software Testing), ISO/IEC TR 20000-4 (Service management).

(D) ‘Standards Relationship Diagram’ (SRD): a relationship diagram showing how current and future SC7 standards could and should be linked. Such a diagram is in effect an ‘ontology of ontologies’. The construction of such a diagram would rely on identification of the ontological underpinnings at the cross-standard level of detail. In other words, when we become able to view standards (or at least their conceptual core) as ontologies, we can use well-known ontology mapping techniques to create an SRD. It shows relationships between ontologies and was initially proposed as a valuable tool for creating a coherent suite of SC7 standards. In the revised report, it was realized that it would serve better as input rather than output (see below).

(E) (more futuristic) ‘Advanced Foundational Ontology for Standards’ (AFOS): an extended formalism to the Definitional Elements Ontology in terms of standard concepts from the ontological engineering literature, such as sortals and moments [16,17,12]. [These concepts are important in the context of work products and, in particular, modelling languages (SC7 WG19’s remit). For example, future versions of UML (ISO/IEC 19505) and UML for RMODP (ISO/IEC 19793) are two standards that will be impacted. Whether they can be also applied to process-focussed standards is less obvious.]

The interconnections between these five suggested ‘ontologies’ is depicted graphically in Figure 4. The core of the proposal is the creation, in the “definitions layer” of the DEO and particular configurations of subsets: CDOs. From each CDO (only one is shown in Figure 4), one or more ‘instance(s)’ or SDO(s) can be created, conformant to the concepts in the CDO. Each of these will be published as an ISO International Standard (IS) or Technical Report (TR). International ISO/SC7 documentation will also include the SRD as a visual depiction of how these various SDOs are interconnected. Finally, a future extension to the definitional DEO by the addition of foundational ontological concepts will be explored.

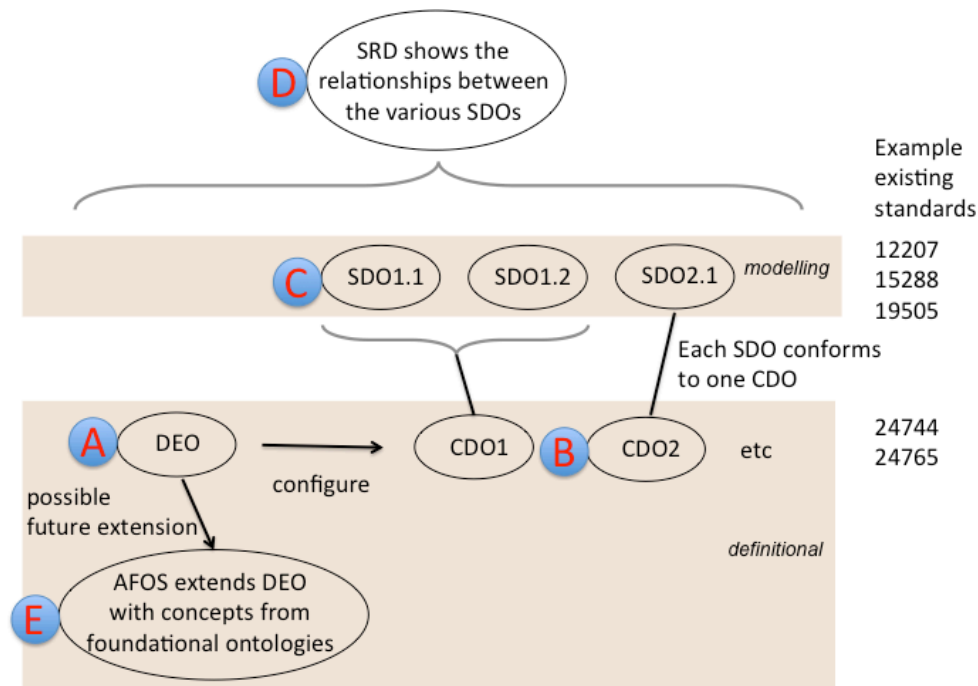


Figure 4 The relationship of the original five 'ontologies' (A-E) and the layered architecture of Figure 3.

After receiving feedback in May 2013 following our presentation to SWG5 and other working groups at the SC7 Plenary in Montréal, it was decided that a better structure would be to focus on the DEO and the derived CDOs. SDOs are then identified simply as being the suite of International Standards conformant to the CDOs.

In the following subsections we describe the original ideas and illustrate how they were later refined to a more compact and more "saleable" infrastructure – engendering adoption of the ontological structure by all working groups in SC7 is a socio-political challenge not a technical one (and therefore such issues are out of scope for this paper describing the technicalities of the proposed ontological infrastructure).

2.1 Definitional Elements Ontology (DEO)

As noted above, Rout [1] outlines the problem of conflicting definitions within SC7 standards. He notes that at the time of his survey (granted now more than a decade ago) that 198 of the 864 definitions he found in SC7 standards were duplicate or overlapping terms and that there were a large number of definitions that needed alignment i.e. slight differences in definitions for apparently the same (key) concept e.g. process, service, system, life cycle.

One highly visible and challenging definition that is urgently needed in order to ensure SC7 standards are both internally consistent and consistent with other industry usage is that of 'method' versus 'process'. SC7 standards prefer the word 'process' rather than 'method'; whereas in other software engineering and information systems disciplines, 'method' (a.k.a. 'methodology') has prominence. Whether this terminological difference entails an underlying conceptual difference is debatable. The underlying rationale is that

there are three predominant areas of interest to software engineering and information systems development: work products (e.g. documents, applications), work units (e.g. activities, tasks) and producers (people and tools). Producers execute work units to create/modify work products. ‘Method’ or ‘methodology’ is usually seen as the combination of these three realms, whereas ‘process’ is understood to refer to work units only (see e.g. clause 3.2 of ISO/IEC 24744).

For widespread adoption of SC7 standards, a single framework that both defines ‘process’ and ‘method’ and relates them to each other and to commonly used terms such as ‘process model’ and ‘process reference model’ is vital. Figure 5 offers one such compromise [18] in which ‘method’ is equated to ‘process model’ and ‘process’ is used for the enactment of that process model. Note that this is not in agreement with current usage in standards such as ISO/IEC 12207, which uses the term ‘process’ to mean an element in the process model of Figure 5. As an alternative and to achieve better conformance, SC7 might consider retaining the use of the term ‘process’ to mean an element in a process model but then relabel the idea of a process (or process model or method) when it is enacted on a real project as something other than ‘process’ (as is currently shown in Figure 5).

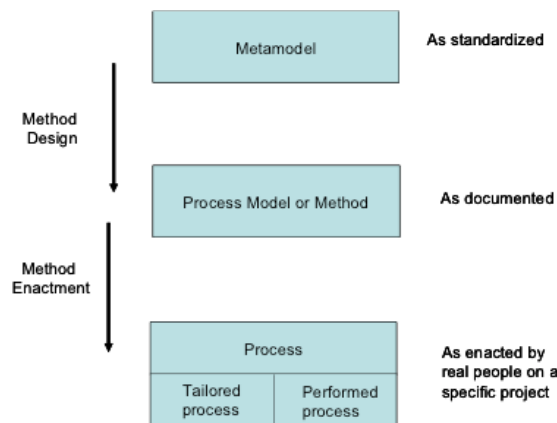


Figure 5 One proposed framework to support terminology of process, method, process model etc. (after [18] figure 2, ©2011) (With kind permission of Springer Science+Business Media.)

More recently, SC7 (notably Working Group 7) has attempted to reconcile definitions within the harmonization project to align ISO/IEC 12207 and ISO/IEC 15288. Also over the last several years, a new International Standard, ISO/IEC 24744, has been specifically developed to create a set of definitions of method-focussed terms and their interrelationships, effectively creating an upper level ontology (an example of a CDO – see Section 2.2) that can be used for the creation of instances of the elements in the CDO that, together, constitute a method(ology) or process model (Figure 5) and a process reference model (examples of an SDO).

Each concept needs:
1) Name – either 1) standard or 2) a pointer to a standard (i.e. a synonym)
2) One sentence definition
3) Any necessary properties
4) Categorization label or labels chosen from the standard set
5) Possible synonyms – if known beforehand
Example:
1) Process
2) A set of activities that transform inputs to outputs
3) Purpose; Outcomes; Activities: Tasks
4) Life Cycle
5) Activity, Work Unit, Procedure

Figure 6 A possible definitional template (upper) with an example (lower) of how this would look in practice in a Definitional Elements Ontology

The proposal for the Definitional Elements Ontology (DEO) is that each definition (concept) adheres to a pre-determined template adhering to a given specification, which includes a name, a definition, and a set of properties. One possibility (just as an example) could be that shown in Figure 6. The upper part of this figure is the template for the concept with an example (here for the concept of Process) in the lower part. The collection of all the 'definitional standards elements' form an interconnected network – the DEO.

The concepts in the DEO will be taken from a combination of several existing ISO sources:

- ISO/IEC 24765 (SEVOCAB), which will be used to make sure that the concepts in the DEO are abstract enough to cater for the needs of all SC7 Working Groups.
- ISO/IEC 24744 (SEMDM), which will be used as the basis for an ontology for software engineering [4].

These two major sources will be augmented with concepts from additional sources. Since ISO/IEC 24765 (SEVOCAB) already takes into account concepts (and definitions) from a large number of standards and technical reports, it is only necessary to examine those additional sources that are not explicitly treated by SEVOCAB. Study group members have identified these additional sources as follows.

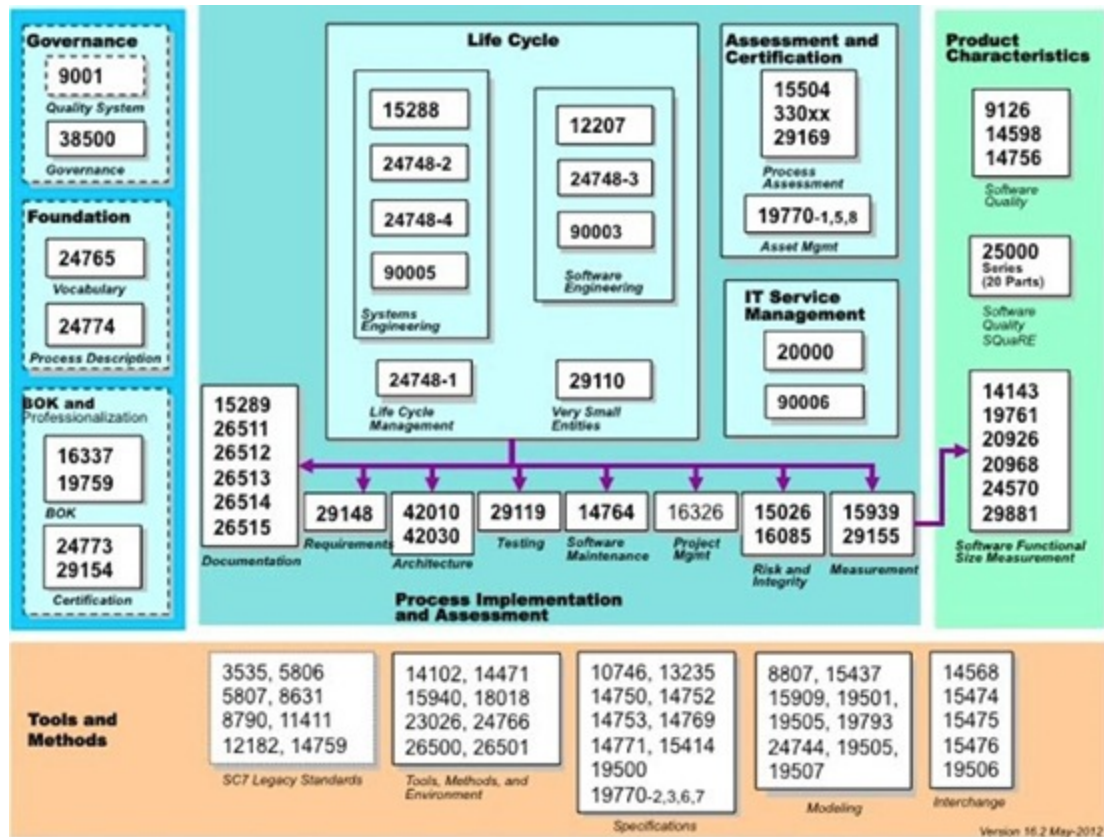


Figure 7 Categorization of SC7 standards (kindly supplied by ISO/IEC/JTC1/SC7). [Note that 26501 appears to be a typographical error in that no such standard exists on the ISO webpage.]

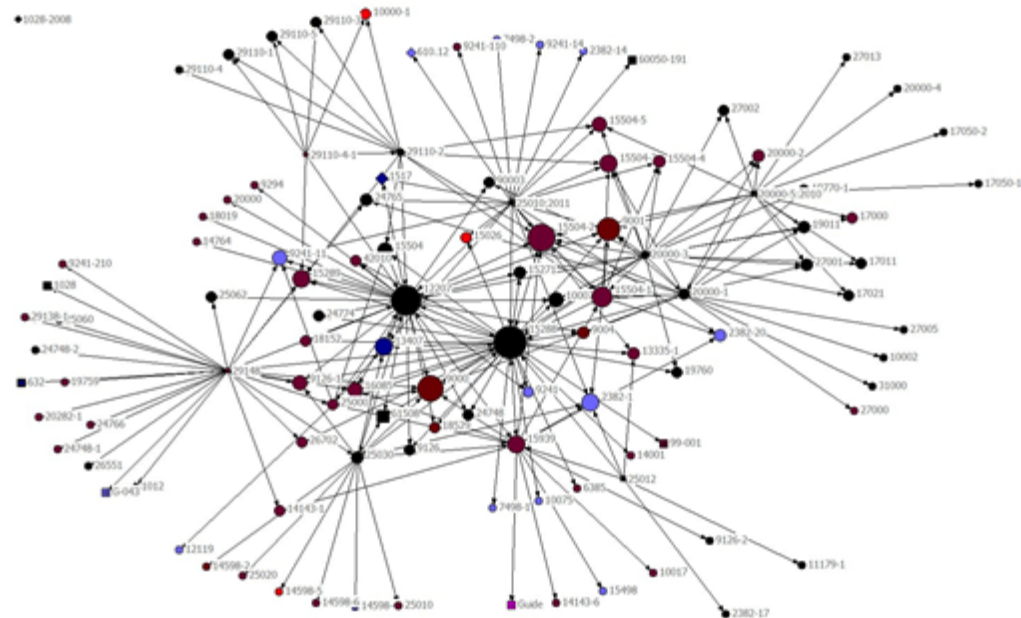


Figure 8 Dependency diagram amongst SC7 standards (October 2011; kindly supplied by ISO/IEC/JTC1/SC7).

Table 1 Documents in development or review.

Number	Title
16337	Systems Engineering -- Systems Engineering Handbook

Number	Title
23026	Software Engineering -- Recommended Practice for the Internet -- Web Site Engineering, Web Site Management, and Web Site Life Cycle
26500	Information technology -- Application management -- Requirements for application management
29110	Software engineering -- Lifecycle profiles for Very Small Entities (VSEs)
29119	Software and systems engineering -- Software testing
29154	Guide for the application of ISO/IEC 24773 -- Certification of Software Engineering Professionals
29155	Systems and software engineering -- Information technology project performance benchmarking framework
29169	Information technology -- The application of conformity assessment methodology to process capability and organizational maturity
33000 series	Software and Systems Engineering - Process assessment
42030	Systems and Software Engineering -- Architecture Evaluation

Table 2 Documents requiring further evaluation regarding potential usefulness.

Number	Title
8807	Information processing systems -- Open Systems Interconnection -- LOTOS -- A formal description technique based on the temporal ordering of observational behaviour
14750	Information technology -- Open Distributed Processing -- Interface Definition Language
18018	Information technology -- Systems and software engineering -- Guide for configuration management tool capabilities
26511	Systems and software engineering -- Requirements for managers of user documentation
26512	Systems and software engineering -- Requirements for acquirers and suppliers of user documentation
26513	Systems and software engineering - Requirements for testers and reviewers of user documentation
26515	Systems and software engineering -- Developing user documentation in an agile environment

Figures 7 and 8 have been obtained from SC7 sources and have provided a list of 116 source documents. Most of these were used as input to create SEVOCAB. Some standards that are not thus included – and hence we recommend for consideration – are ISO/IEC 15437, 18152, 18529, 24773, 24774, 29148 and 42010. In addition, we have identified 11 documents in development (Table 1). There are also a small number of documents that may or may not be useful in this project; these are listed in Table 2. These elements are elicited from those in the “Terminology” section of these standards. In addition, some standards (one in particular: ISO/IEC 24744) has at its core a number of additional conceptual definitions. For each of the 69 concepts in ISO/IEC 24744 SEMDM, there might be up to 7 corresponding definitions in ISO/IEC 24765 SEVOCAB]. Rationalizing these multiple definitions could take, on average, several hours for each term. Thus the new work-plan (to be delivered in November 2013) will include an overall estimate of the likely effort involved in implementing the research ideas described here.

Necessarily, the DEO is highly abstract, in order to guarantee that it covers everything under the scope of SC7. Therefore, the DEO cannot be utilized without change for all standards. Rather, it needs to be tailored to the specific needs of each working group or standard.

In addition, it might be decided to split such an enhanced document into two: a definitional (DEO) document (or part thereof) plus a Technical Report that gives advice on how to write descriptions successfully.

The specifics of the tailoring mechanism are given in the next section (especially Section 2.2).

The DEO is an infrastructure that allows for the integration of the individual CDOs. The DEO is a collection of definitions of individual concepts and terms, together with constraints dictating how they may be configured. Among other things, these constraints help to resolve the twin problems of a) same concept, different terms and b) same terms, different concepts.

As an ontology, the DEO focuses on structure in a specific domain rather than being a mere list of terms in that domain (as in ISO/IEC 24765). Also, the DEO is built for extension and reuse, so that CDOs can be constructed. At present, SC7 relies on each working group defining the terms it needs. The SC7 vocabulary (ISO 24765) is a collection of terms from SC7 standards that are frequently duplicated but slightly different in expression or meaning. It is not an ontology.

In summary, the Definitional Elements Ontology is proposed as the cornerstone for the creation of future standards (right hand side of Figure 3). This can be done in one of two ways: either a direct configuring of the elements in the Definitional Elements Ontology to create a Configured Definitional Ontology (Section 2.2) or by deriving a number of instances of several of the definitions in the Definitional Elements Ontology to create an International Standard or Technical Report (Section 2.3).

The research methodology is thus to identify core concepts from these two main sources (ISO/IEC 24765 and 24744) and to abstract away to a core definition from which all the other definitions can be derived in their individual CDOs. Although the elements in the DEO are well connected, zero-to-many cardinalities on certain relationships permit ‘pruning’ of the DEO when it is necessary to create a tailored and more specific standard. Such tailored standards may be the set of standards within a single working group that therefore all use the same configuration (CDO) or within a working group as a “family” of standards (see Figure 10 below) or at the individual standard level.

2.2 Configured Definitional Ontology (CDO)

A ‘Configured Definitional Ontology’ can be created by selecting a number of the definitional standards fragments in the DEO and configuring them into a complete upper level ontology (the CDO). A CDO is created by using a combination of the two following mechanisms:

- Discarding areas of the DEO that are not relevant.
- Adding new concepts, as refinements of those in the DEO.

These two mechanisms are illustrated in Figure 9 and described in detailed in the two subsections below.

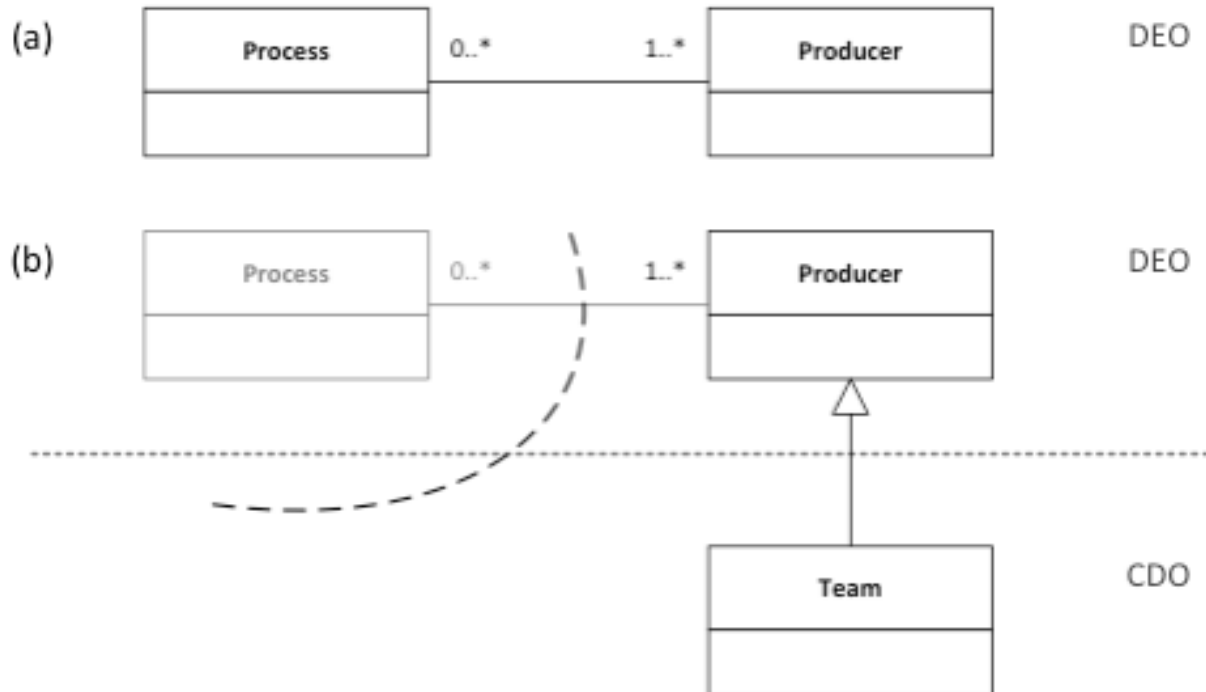


Figure 9 Mechanisms for tailoring the DEO. (a) shows the DEO in its original form. (b) shows the DEO as being used by a particular CDO: a section of the DEO is being discarded and a subclass has been added.

2.2.1 Discarding areas of the DEO

Although the elements in the DEO are all interconnected, associations between two concepts may have a cardinality of zero; this means that for any occurrence of the concept on one side of the association, there may be multiple or no occurrences on the opposite side. For example, Figure 9 shows concepts **Process** and **Producer** plus an association between them with a zero-to-many cardinality on the **Process** side. This means that any particular producer (such as software developer Jane) may have no processes at all. For this reason, a project focussing on modelling producers could legitimately discard the **Process** concept altogether and use only the **Producer** concept. However, a project focussing on processes could not discard the **Producer** concept, since every possible process occurrence would need at least an associated producer.

2.2.2 Adding elements to the DEO

As well as discarding unwanted sections of the DEO, new concepts can be introduced into a CDO if necessary. This is often the case when specialized definitions are needed that are not present in the abstract DEO.

It is a necessary restriction that an introduced concept may not contradict an existing concept. Instead, newly added concepts are introduced by refining an existing concept in the DEO through specialization. This means that the newly introduced concept must

be semantically “a kind of” the base concept. For example, Figure 9 shows Team as a specialized concept that has been introduced as a kind of Producer.

2.2.3 Chaining CDOs

CDOs are always created in a particular context, which determines the relevant set of concepts. This context may be a working group, a family of standards, or even a particular standard. Figure 10 shows how the DEO is tailored to two different SC7 working groups by creating two CDOs through the mechanisms described above. In turn, one of these CDOs is depicted as being further refined into a suite of CDOs that are specific for families of standards within that working group. And then again, these are refined into standard-specific CDOs.

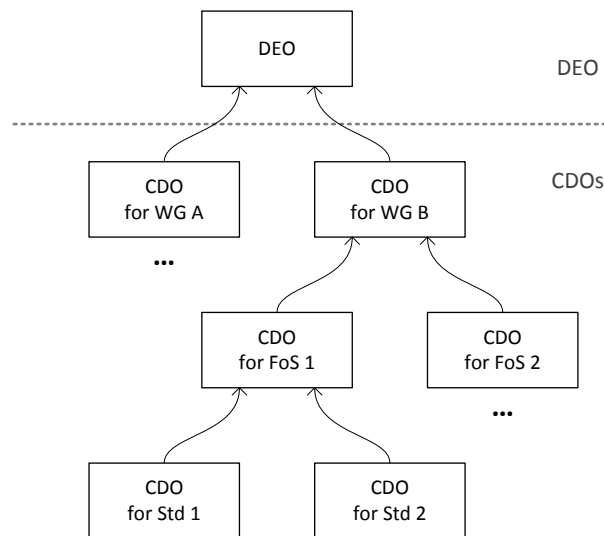


Figure 10 Chaining CDOs at various levels. The top row of CDOs corresponds to work groups. The middle row corresponds to families of standards. The bottom row corresponds to specific standards. The arrows mean “conforms to”.

In the chain of CDOs, detail is added gradually so that only the concepts that are relevant to a particular context (either a working group, a family of standards or a single standard) are included. Care must be taken so that under- or over-specification is avoided.

The structure shown in Figure 10 is just an example; there is no need to use every level shown. For example, a CDO for a particular standard could derive directly from the DEO.

This chaining mechanism ensures that each working group retains ownership of its own set of concepts, while:

- Maintaining the interoperability with other working groups through the shared DEO.
- Permitting individual standards to refine these concepts into more specific ones.

2.2.4 Example CDO

Although ISO/IEC 24744 is a major source for the creation of the DEO, the DEO, once constructed, will be more abstract and therefore have less detail than the current IS/IEC 24744. The current International Standard is thus retained in its current form, which is therefore an example of a CDO (Figure 11). Unlike ISO/IEC 24774, it does not, on pur-

pose, offer any guidelines on writing these. Thus, again, we recommend the distinction between (i) an International Standard that offers definitions (DEO) together with a configuration (turning the DEO into a CDO) and (ii) a Technical Report that offers advice on writing the elements in that CDO.

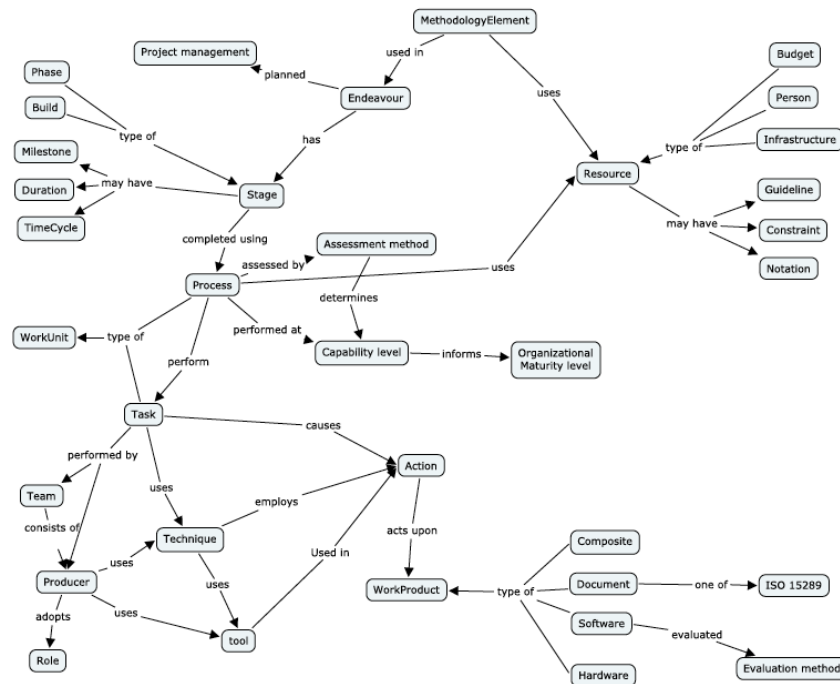


Figure 11 Exemplar configured standard or upper level ontology (CDO) (actually ISO/IEC 24744) depicted graphically to show relationships between standards fragments

2.3 Standard Domain Ontology (SDO)

Strictly speaking, an SDO is not a single ontology but a suite of domain ontologies, each targetted at a particular IT domain of interest. An SDO becomes the core of an International Standard (or a TR). It is important to discriminate between the definitions (in the definitional domain – the DEO and the CDO: foundational ontologies) as compared to examples (instances) of those definitions (in the modelling domain, a domain ontology – as outlined in this section).

In any SDO, it is likely that only a small subset of the definitions in the DEO repository would be needed. For example, in a process-focussed standard such as ISO/IEC 12207, it may only be necessary to utilize definitions for, say, Process, Activity, Task (Figure 12) and their inter-relationships, which provide the template for the creation of instances of these definitions that are then put together as a Configured Definitional Ontology (CDO), wherein each 12207 ‘process’ (here, the Training Process (Figure 13) is just one example) can be assessed for conformance to the definition of Process given in the DEO – or CDO if a CDO is used. In other words, a check that all the elements of the new standard

(an SDO) are predetermined by the definition of 'Process' in the CDO (e.g. as in Figure 12), this definition having been selected from all the elements in the DEO.

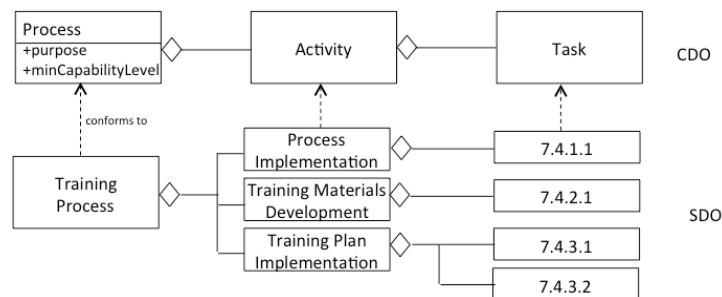


Figure 12 A portion of a CDO and one exemplar SDO element (Training Process). The numbers in the right hand side boxes relate to clauses in ISO/IEC 12207 (after [3] figure 9, ©2012) (With kind permission of Springer Science+Business Media.)

7.4 Training process

The Training Process is a process for providing and maintaining trained personnel. The acquisition, supply, development, operation, or maintenance of software products is largely dependent upon knowledgeable and skilled personnel. For example: developer personnel should have essential training in software management and software engineering. It is, therefore, imperative that personnel training be planned and implemented early so that trained personnel are available as the software product is acquired, supplied, developed, operated, or maintained.

List of activities. This process consists of the following activities:

- 1) Process implementation;
- 2) Training material development;
- 3) Training plan implementation.

7.4.1 Process implementation. This activity consists of the following task:

7.4.1.1 A review of the project requirements shall be conducted to establish and make timely provision for acquiring or developing the resources and skills required by the management and technical staff. The types and levels of training and categories of personnel needing training shall be determined. A training plan, addressing implementation schedules, resource requirements, and training needs, should be developed and documented.

7.4.2 Training material development. This activity consists of the following task:

7.4.2.1 Training manuals, including presentation materials used in providing training, should be developed.

7.4.3 Training plan implementation. This activity consists of the following tasks:

7.4.3.1 The training plan shall be implemented to provide training to personnel. Training records should be maintained.

7.4.3.2 It should be ensured that the right mix and categories of appropriately trained personnel are available for the planned activities and tasks in a timely manner.

Figure 13 Example process from the earlier version of ISO/IEC 12207 showing the full details of the training process (Reproduced with permission from SAI Global under Licence 1308-c098) [In the more recent version (2008) this has been updated as the Human Resources Management process (clause 6.2.4)]. All ISO standards are obtainable from <http://www.iso.org> and from <http://www.saiglobal.com>

Thus, standards that offer a 'process model' or a 'process reference model' are in the realm of the SDO. Examples of SDOs in current SC7 standards are ISO/IEC 12207, 15504 and 15288.

2.4 Standards Relationship Diagram (SRD)

Originally, it was proposed to create a Standards Relationship Diagram (SRD) as a domain ontology of these produced standards. One example is depicted in Figure 8, which shows how standards are inter-related as a dependency network diagram or social network of standards. Since each standard normally lists other normative standards or related standards, these relationships can be plotted here using UCINET as a social network. Figure 8 shows which standards are affected by revisions to any one other standard. For example, if ISO/IEC 12207 changes, which standards are affected? Social networks are good for illustrating such questions. However, in May 2013 it was realized that such an SRD could more usefully be regarded as an input, not an output, to this project (as noted above in Section 2) since with the revised DEO/CDO structure, influence connections are “upward” via the DEO, not “sideways” as in Figure 8.

Categorization of standards is consequently required; for example, SC7 categories of Governance, Foundation, Body of Knowledge (BOK), Life Cycle, Assessment & Certification, Product, Process and Tools & Methods. These categories are also assumed in a 2013 proposal to create an up-to-date revision of ISO/IEC TR 12182 Information Technology – Categorization of Software – 1998.

2.5 Advanced Foundational Ontology for Standards (AFOS)

Particularly in the domain of modelling languages, it has been pointed out in the literature [12, 19] that the finest grained concepts in a DEO (often expressed as a metamodel as in ISO/IEC 19505) is in fact coarse and ignores much of the literature in ontology engineering. There have thus been suggestions to extend the metamodel defining a modelling language like UML, as suggested in Figure 14.

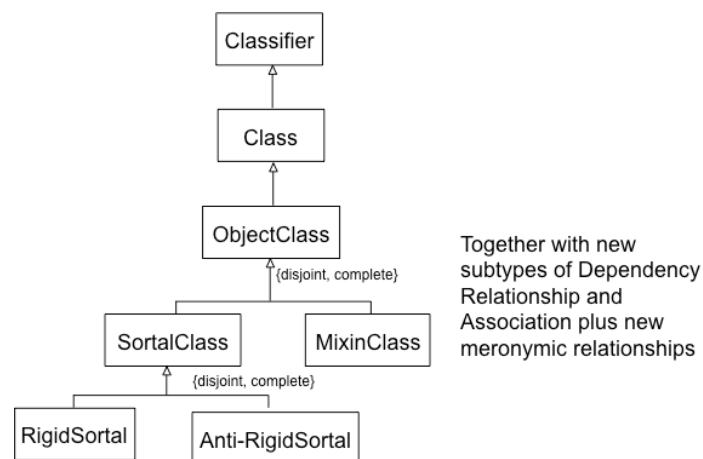


Figure 14 One proposal to incorporate ontological engineering ideas into a product-focussed standard such as ISO/IEC 19505 [which currently only contains Classifier & Class]. (after [12] figure 7.20, ©2012) (With kind permission of Springer Science+Business Media.)

An Advanced Foundational Ontology for Standards (AFOS) is predominantly useful for more sophisticated conceptual modelling and is relevant to product-focussed standards, several of which are OMG-adopted standards, such as ISO/IEC 19505 (UML).

3. Implementation and initial validation

These ideas are readily implementable. The first implementation step is to identify and agree upon definitions of a wide variety of concepts used in software engineering. These constitute the DEO (Definitional Elements Ontology). As noted above, the DEO definitions will need to be collected from various sources: ISO standards 24744 and 24765, supplemented by those ISO standards not contributing to ISO/IEC 24765.

The importance of having a DEO applicable to all SC7 standards can be demonstrated by consideration of what happens when a NWI proposes a standard in a ‘new’ domain. Advocates of this new domain standard are likely to have different tacit understanding of the terms they use. It is thus critical that a first step is to evaluate the definitions (semantics) of all terminology in the proposed new domain and to ensure it is compatible with the SC7 DEO. In some circumstances, for example when brand new technologies arise, there may be a need to add extra elements to the DEO but this should be done carefully to ensure they are well defined and there are no clashes (contradictions or overlaps) with existing DEO definitions. In other cases, synonym mappings may be usefully identified in order to retain consistency in both the domain-specific standard (thus Working Groups retain ‘ownership’) and also the coherence of the elements in the DEO.

Another important benefit of the proposed approach is that, as explained above and thanks to the CDO chaining mechanism, each working group continues to own the concepts and terms that it introduces but with the obligation that these concepts and terms conform to (either directly or via refinement) concepts in the DEO. Once a concept is in the DEO, a working group can use synonyms of the terms or refinements of the concepts in their work as they see fit.

It is an important recommendation that the route of a single comprehensive DEO MUST be taken. Whilst apparently cumbersome, it is easy to select only relevant parts of such an International Standard when creating a CDO or an SDO for instance. Techniques have been explored by e.g. [20, 21]; see also [22]. This is known as SMME (situational meta-model engineering), and closely parallels the ideas of situational method engineering e.g. [23].

It is vital to avoid the route of multiple independent CDOs, since this mimics the diversity (and incompatibilities) identified in the current suite of SC7 standards.

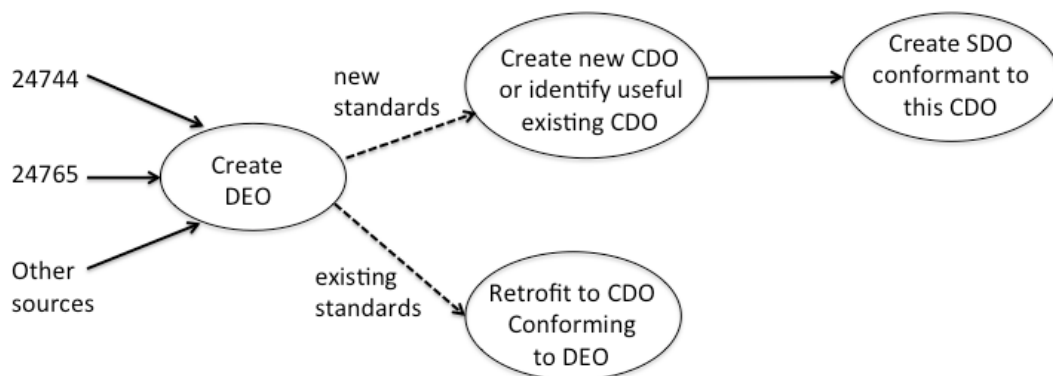


Figure 15 Schematic illustrating the creation of an SDO conformant to a CDO for both new standards and for existing standards

To create an SDO (Figure 15), it is necessary first to delineate (i.e. scope) the area of interest (e.g. process, services, governance) and then identify the concepts relevant to this area of interest. Then the standards developer needs either (i) to identify in the DEO repository those definitions that will support the new standard and then create an interim configuration analogous to a CDO (an interim work product) or (ii) select an existing International Standard (a CDO) that already covers the domain of interest.

In both cases, the SDO then consists of several instances conformant to each element in the CDO just created (case i) or identified (case ii). An example has been shown in Figure 12 (lower part).

Strictly speaking an SRD simply documents the existing situation, visualized differently. With contemporary search tools, such links and commonalities are easily obtained. Currently, inter-standards dependencies rely on the standards writing identifying, in words, ‘those standards that may need to be changed consequent on a change in this standard’. Whilst Figure 8 illustrates a dependency diagram constructed as a result of this kind of cross referencing, a true (and more useful) SRD will seek/identify links at a much finer granularity. For example, we could identify the use of ‘Task’ in ISO/IEC 12207 (Figure 12) and, using ontology matching techniques e.g. [24], attempt a match with ‘Task’ in ISO/IEC 24744 and in ISO/IEC 24774 (wherein both words appear). In other words, these dependencies can be fully automated via a database that contains *all* DEO elements and list of all the SC7 standards that refer to these DEO elements/relationships. Once the DEO has been standardized (as an IS), then the current Terminology sections of all SC7 standards is replaced by a statement of how the standard conforms to the DEO.

Typically, a domain ontology is a structured hierarchy showing how various elements are defined with respect to other elements. In order for this to be feasible, characteristics of the *overall standard* need to be addressed i.e. there needs to be a definitional template for each category of standard. This has been acknowledged already in ISO documents as in [25]. This document suggests several templates depending upon the categorization of the standard.

Another validation exercise involves investigating the practical use to which ISO standards have been previously put, using documented case studies. Some examples include the creation of a Spanish maturity model based on ISO standards [26], the application of ISO/IEC 24744 to traceability [27] and the utilization of RM-ODP, ISO/IEC 10746 [28] to intelligent transport systems [29]. These case studies will be used to evaluate whether the proposed revisions to SC7 standards will still permit (i.e. can replicate) these specific application areas.

4. Maintenance, responsibilities and future adoptions

Once the DEO has been standardized as either an International Standard or a Technical Report within SC7, three processes begin in parallel:

- The DEO is maintained on a continuous basis
- Existing standards get reconciled against the DEO
- New standards will need to be aligned to the DEO

as described in the following subsections.

4.1 Maintaining the DEO

Although the DEO will be a standard and therefore should be relatively stable over a period of years, it is inevitable that, eventually, changes will need to be introduced. It is recommended that SWG5 take responsibility for maintaining the DEO, incorporating the changes that they see fit from feedback obtained, either proactively or reactively, from the different working groups in SC7.

4.2 Reconciling existing standards

We are aware that altering existing standards to conform to the DEO may entail a significant amount of work in some cases. It is recommended that existing SC7 standards are considered for DEO reconciliation whenever they become open for periodical review. The owner working group should be responsible for this task.

Reconciliation entails re-describing the standard as a CDO and, if necessary, a collection of occurrences of said CDO. If the focus of standard is the occurrences of concepts (e.g. ISO/IEC 12207), then an explicit CDO is still necessary as part of the reconciled version in order to document the semantics of said occurrences.

In any case, reconciling a standard with the DEO would only mean an internal refactoring with no changes whatsoever to the semantics of its prescriptive parts. In other words, the concepts in the reconciled version of the standard may need to refer to parent concepts in the DEO, but otherwise remain unchanged. Often, it will be necessary to map the standard's terms and definitions to the SC7 DEO, revise the standard to use standard terms where possible and migrate terms to the SC7 DEO over time.

The effort required to accomplish this can be extremely variable and should be the responsibility of the overseeing working group.

4.3 Creating new standards

Any new standard that is created after the DEO is in place should be strongly encouraged to be fully compliant with the DEO. This means that the concepts in it must be a CDO and, for an SDO-focussed IS (e.g. ISO/IEC 12207), the elements must be instances of that CDO. In this case, an explicit CDO is still necessary as part of the new standard in order to document the semantics of said occurrences. This statement will replace the content of the "Terms and definitions" section present in all current SC7 standards.

4.4 Guidance and tools

Once all of the above has been implemented, there will be a need to provide guidance for would-be standards developers with regard to the process to follow in relation to CDO development, CDO instantiation etc. A flow chart or decision tree may be useful in order to guide potential standards builders.

In addition, it is suggested that ontology management tools be employed so that relationships between standards can easily be tracked and the DEO/CDOs structure be easily maintained.

5. Recommendations and conclusion

5.1 Recommendations

Once the above ontological structure has been implemented, there will be a need to formalize these not only in revised standards such as the Standard Domain Ontology standards of ISO/IEC 12207 etc. (ensuring they conform to the elements in the DEO and to one specific CDO) but also to introduce new standards and technical reports for would-be standards developers as guidelines to their standards activities.

A new suite of standards is thus required to explain and formalize all this.

- i) There is a need for a standard (or Technical Report) to explain how (a) to put together and (b) how to tailor such a standard.
- ii) Because the spectrum of standards is broad and we can't change accepted terminology in all subdomains, there MUST a synonym list so that non-standard standards can relate back to the standard vocabulary.

5.2 Conclusion and Future Work

We have shown how conceptual modelling and ontology engineering can be efficacious in reconciling otherwise disparate International (ISO) Standards – here the software engineering standards developed under the auspices of SC7. We have applied ontological thinking to the various existing standards and identified four areas in which ontological engineering can contribute immediately to increasing the quality of the SC7 suite of software engineering standards plus one more futuristic technique. The ontologies thus identified can be considered parallel to well-known situational method engineering strategies [30] leading to a repository of definitional standards (a DEO), a configuration of those elements into a CEO – both at the 'meta' level. For the software developer, there are then configured modelling-focussed ontologies (CDO), which underpin a large number of the SC7 standards. Finally, there is interest in understanding how these standards relate to each so that when changes occur in one standard, any interlinked standard is readily identified and similarly updated. These recommendations will form part of a second submission to be made to the SC7 committee and its working groups as a prelude to the creation and endorsement of a NWI (New Work Item) in 2013-14, which will finally culminate in a new IS (the DEO).

The next step in our research is to implement this infrastructure and create a comprehensive DEO. As discussed earlier, this requires identifying and clarifying a core set of concepts at a relatively abstract level so that they can be readily used to create a suite of domain-specific CDOs; some of which will need to be created as further 'proof of concept' and validation of the ontological approach developed in this paper.

6. Acknowledgements

We wish to acknowledge the comments from the SC7 Study Group members on the several drafts of the first report prior to the submission to SC7/SWG5 in March 2013.

References

1. T.P. Rout, Consistency and conflict in terminology in software engineering standards, In: Procs. Fourth IEEE Int. Symposium and Forum on Software Engineering Standards (ISESS'99), IEEE Computer Society, 67-74 (1999)
2. B. Henderson-Sellers, Object-Oriented Metrics. Measures of Complexity, Prentice Hall, NJ, 234pp, 1996.
3. B. Henderson-Sellers, Standards harmonization: theory and practice, *Software and System Modeling*, 11(2) (2012) 153-161.
4. C. Gonzalez-Perez, B. Henderson-Sellers, An ontology for software development methodologies and endeavours, In: C. Calero, F. Ruiz and M. Piattini (Eds.), *Ontologies in Software Engineering and Software Technology* Springer-Verlag, 2006, pp. 123-152.
5. T. McBride, B. Henderson-Sellers and reviewers, The growing need for alignment, ISO/IEC JTC1/SC7 N5507 (12 May 2012)
6. Wikipedia, http://en.wikipedia.org/wiki/Sarbanes%E2%80%93Oxley_Act (accessed 17 March 2013)
7. C. Pardo, J.P. Pino, F. Garcia, M. Piattini, M.T. Baldassarre, An ontology for the harmonization of multiple standards and models, *Computer Standards & Interfaces*, 34 (2012) 48-59.
8. B. Henderson-Sellers, T. McBride, G.C. Low, C. Gonzalez-Perez, Ontologies for international standards for software engineering, In: Procs. ER2013, Hong Kong, Springer (2013)
9. L. Delcambre, R. Kaschek (eds.), *The Evolution of Conceptual Modeling*. LNCS 6520, Springer, Berlin, 2011.
10. G. Guizzardi, Theoretical foundations and engineering tools for building ontologies as reference conceptual models. *Semantic Web* 1 (2010), 3-10.
11. OMG, Software Process Engineering Metamodel Specification, Version 1.0, Object Management Group, formal/02-11-14, 98pp (2002).
12. B. Henderson-Sellers, On the Mathematics of Modelling, *Metamodelling, Ontologies and Modelling Languages*. SpringerBriefs in Computer Science, Springer (2012).
13. C. Gonzalez-Perez, T. McBride, B. Henderson-Sellers, A metamodel for assessable software development methodologies, *Software Quality Journal* 13(2) (2005) 195-214.
14. B. Henderson-Sellers, Bridging metamodels and ontologies in software engineering. *Journal of Systems and Software* 84(2) (2011) 301-313.
15. G. Guizzardi, G. Wagner, Towards ontological foundations for agent modelling concepts using the Unified Foundational Ontology (UFO). In: P. Bresciani, P. Giorgini, B. Henderson-Sellers, G. Low, M. Winikoff (Eds.), *Agent-Oriented Information Systems II*, LNAI 3508, Springer, Berlin, 2005, pp. 110-124.
16. G. Guizzardi, G. Wagner, N. Guarino, M. van Sinderen, An Ontologically Well-Founded Profile for UML Conceptual Models. In: A. Persson, J. Stirna (Eds.), *16th International Conference on Advances in Information Systems Engineering (CAiSE'04)*, Latvia. LNCS vol. 3084, Springer-Verlag, Berlin, 2004, pp. 112-126.
17. G. Guizzardi, L. Ferreira Pires, M. van Sinderen, An Ontology-Based Approach for Evaluating the Domain Appropriateness and Comprehensibility Appropriateness of Modeling Languages. In: Briand, L., Williams, C. (Eds.), *Model Driven Engineering Languages and Systems*. LNCS vol. 3713, Springer-Verlag, Berlin, 2005, pp. 691-705.
18. T. McBride, B. Henderson-Sellers, A method assessment framework. In: J. Ralyté, I. Mirbel, R. Deneckère (Eds.), *Engineering Methods in the Service-Oriented Context*, Springer, 2011, pp.64-76
19. G. Guizzardi, *Ontological Foundations for Structural Conceptual Models*. Enschede, The Netherlands (2005)
20. C. Hug, *Méthode, modèles et outil pour la méta-modélisation des processus d'ingénierie de systèmes d'information*, PhD thesis, Université Joseph Fourier - Grenoble I, France (2009)
21. C. Hug, A. Front, D. Rieu, B. Henderson-Sellers, A method to build information systems engineering process metamodels. *J. Systems Software* 82(10) (2009) 1730-1742.
22. S. Sen, N. Moha, B. Baudry, J.M. Jezequel, Metamodel pruning. In: Procs. ACM/IEEE 12th International Conference on Model Driven Engineering Languages and Systems (MODELS'09), Springer, 2009, pp. 32-46.
23. B. Henderson-Sellers, J. Ralyté, P. Ågerfalk, M. Rossi, *Situational Method Engineering*, Springer-Verlag, Heidelberg. 2013.
24. M. Mohsenzadah, F. Shams, M. Teshnehlab, A new approach for merging ontologies, *World Academy of Science and Technology* 4 (2005) 153-159.
25. Taxonomy of Standards Document, dated 30 August 2011
26. J. Garzás, F.J. Pino, M. Piattini, C.M. Fernández, A maturity model for the Spanish software industry based on ISO standards, *Computer Standards & Interfaces* 35 (2013) 616-628.
27. A. Espinoza, J. Garbajosa, A study to support agile methods more effectively through traceability, *Innovations Syst. Softw. Eng.* 7 (2011) 53-69.

28. H. Kilov, P.F. Linington, J.R. Romero, A. Tanaka, A. Vallecillo, The Reference Model of Open Distributed Processing, Foundations, experience and applications, Computer Standards & Interfaces 35 (2013) 247-256.
29. I. Román, G. Madinabeitia, L. Jimenez, G.A. Molina, F. Pino, M. Piattini, C. Manuel, Experiences applying RM-ODP principles and techniques to intelligent transportation system architectures, Computer Standards & Interfaces 35 (2013) 338-347.
30. B. Henderson-Sellers, J. Ralyté, Situational Method Engineering: state-of-the-art review, Journal of Universal Computer Science 16(3) (2010) 424-478.

ISO International Standards cited in text. These are all obtainable from the publisher (ISO) or via <http://www.iso.org> or from a national standards body in your country of residence.

ISO 704 Terminology work - Principles and methods. Third edition (2009)
 ISO 10241 International terminology standards — Preparation and layout (1992)
 ISO/IEC 10746 Information technology -- Open distributed processing -- Reference model: Foundations (2009)
 ISO/IEC 12207 Systems and software engineering -- software life cycle processes (Amendment 1995, 60pp and 2008)
 ISO/IEC 14764 Software Engineering -- Software Life Cycle Processes -- Maintenance (2006)
 ISO/IEC 15288 Systems and software engineering -- system life cycle processes (2008)
 ISO/IEC 15437 Information technology -- Enhancements to LOTOS (E-LOTOS) (2001)
 ISO/IEC 15504 Information technology -- process assessment. Part 1: Concepts and vocabulary (2004)
 ISO/IEC 15939 Systems and software engineering -- Measurement process (2007)
 ISO/IEC 16085 Systems and software engineering -- Life cycle processes -- Risk management (2006)
 ISO/TS 18152 Ergonomics of human-system interaction -- Specification for the process assessment of human-system issues (2010)
 ISO/TR 18529 Ergonomics -- Ergonomics of human-system interaction -- Human-centred lifecycle process descriptions (2000)
 ISO/IEC 19505 OMG Unified Modeling Language (UML2.1.2) - Part 1 Infrastructure and Part 2 Superstructure (2012)
 ISO/IEC 19770-1 Information technology -- Software asset management -- Part 1: Processes and tiered assessment of conformance (2012)
 ISO/IEC 19793 Information Technology -- Open distributed processing -- Use of UML for ODP system specifications (2008)
 ISO/IEC 24744 Software Engineering -- Metamodel for Development Methodologies (2007)
 ISO/IEC/IEEE 24765 Systems and software engineering -- Vocabulary (2010)
 ISO/IEC 24773 Software engineering -- Certification of software engineering professionals -- Comparison framework (2008)
 ISO/IEC/IEEE DIS 29119-2.2 Software and systems engineering -- Software testing -- Part 2: Test process (DIS stage April 2012)
 ISO/IEC/IEEE 29148 Systems and software engineering -- Life cycle processes -- Requirements engineering (2011)
 ISO/IEC 42010 Systems and software engineering -- Architecture description (2011)

ISO/IEC Technical Reports cited

ISO/IEC TR 12182 Information Technology -- Categorization of Software -- 1998
 ISO/IEC TR 20000-4 Information technology -- Service management -- Part 4: Process reference model (2010)
 ISO/IEC DTR 24774 System and software engineering -- life cycle management -- guidelines for process description (2006)

Brief bios and photos

Brian Henderson-Sellers received the doctor of science (DSc) degree from the University of London in 2001 for his research contributions in object-oriented methodologies. He is the director of the Centre for Object Technology Applications and Research and a professor of information systems at the University of Technology, Sydney (UTS). He is the author or editor of 34 books and over 600 other publications.



Cesar Gonzalez-Perez is a Staff Scientist at Incipit · CSIC, where he leads a co-research line in software engineering and cultural heritage. His ultimate goal is to develop the necessary theories, methodologies and technologies to help us ascribe meaning to cultural heritage information. Previously, Cesar has worked in Spain and Australia both in industry and academia, and in the fields of conceptual modelling, metamodeling and method engineering. He has started three technology-based companies, he is an elected member of the steering committee of the Computer Applications and Quantitative Methods in Archaeology association, and he has authored over 50 academic publications.



Tom McBride is a senior lecturer at the Faculty of Information Technology at the University of Technology, Sydney. He has been involved in developing and teaching a number of subjects. His research interests include software engineering, design thinking for software and systems, and human-centred software development processes. He has worked in the software industry for more than 25 years with most recent experience in both project management and quality assurance. He is involved with software engineering standards development with both Standards Australia and the International Standards Organization where he has helped to develop several standards and is currently involved in revising process assessment related standards.



Graham Low received the BE and PhD degrees from The University of Queensland. He is an emeritus professor of information systems in the School of Information Systems, Technology and Management at the University of New South Wales. His research program focuses on the implementation and adoption of new technologies. This can take the form of new/modified approaches/techniques for information systems development such as methodological approaches to agent-oriented information systems design and management of the information systems design and implementation process.

