Coetzee S, Cox S and Herring J. Configuration management of a system of interdependent standards. *7th International Conference on Standardization and Information Technology (SIIT)*, Berlin, Germany, 28-30 September 2011.

Configuration management of a system of interdependent standards

Serena Coetzee[1], Simon Cox[2] and John Herring[3]
[1]*Centre for Geoinformation Science, Department of Geography, Geoinformatics and Meteorology, University of Pretoria, South Africa, serena.coetzee@up.ac.za*
[2]*CSIRO, Earth Science and Resource Engineering, Kensington, WA, Australia, simon.cox@csiro.au*
[3]*Oracle, New England Development Center, Nashua, NH, United States of America, john.herring@oracle.com*

ABSTRACT

ISO/TC 211, *Geographic information/Geomatics* develops the ISO 19100 series of geographic information standards in close collaboration with the Open Geospatial Consortium (OGC). Forty-nine ISO 19100 standards have been published to date, and several were recently approved for revision in accordance with ISO's review process. Dependencies among standards have to be managed as changes are introduced in revisions. Configuration management (CM) refers to identifying the configuration of a system at distinct points in time for the purpose of systematically controlling changes to the configuration, and maintaining the integrity and traceability of the configuration throughout the system life cycle. In this paper we explain the configuration management challenges for a 'system' of interdependent standards, which are expressed using standard documents, a Unified Modeling Language (UML) model and Extensible Markup Language (XML) schemas. We present two conceptual models and propose improved configuration management activities for ISO/TC 211. We conclude with a summary of results that are applicable in other ISO technical committees.

Keywords: standardization, configuration management, conformance testing, UML, XML, standard revision

INTRODUCTION

A spatial data infrastructure (SDI) aims to make spatial (geographic) data usable to as wide an audience as possible. The implementation of spatial data infrastructures in countries, regions and globally has created a demand for a suite of harmonized geographic information standards through which spatial data can be accessed, displayed, exchanged and analyzed. One example of an SDI is the INfrastructure for SPatial InfoRmation in Europe (INSPIRE), which aims to create a European Union (EU) spatial data infrastructure that will enable the sharing of environmental spatial information among public sector organizations and better facilitate public access to spatial information across Europe (Directive 2007/2). INSPIRE and other SDI implementations rely extensively on standards published by recognized bodies. Similarly, implementations of omnipresent digital geographic information on Internet Mapping sites (e.g. Google Earth, VirtualEarth and OpenStreetMap) and handheld devices (e.g. cell phones and GPS devices) contribute towards the demand for a suite of harmonized geographic information standards.

Standardization in the field of geographic information gained momentum in the 1990s when geographic information systems (GIS) matured and the advent of the Internet accelerated the sharing and exchange of information. In 1994, a technical committee for standardization in the field of geographic information was created in the International Organization for Standardization (ISO), the ISO/TC 211 *Geographic information/Geomatics* (www.isotc211.org). The first standard was published in 2000 and since then, a suite of standards for geographic information has been published; the first ISO number assigned is 19101 and the number of the latest project under development is 19160. More than half of the 49 published standards have undergone ISO's periodic systematic review in the past few years. Only four of these were confirmed (i.e. no revision required); it was decided that most of the others should be revised, based on the comments received in the systematic reviews. Some of these revisions have already commenced.

The Open Geospatial Consortium, Inc. (OGC) is an international industry consortium with more than 410 members, including companies, government agencies and universities (www.opengeospatial.org). The OGC, also founded in 1994, was one of the first liaisons approved by ISO/TC 211 in 1995. Since 2003 collaboration on geographic information standards between ISO/TC 211 and OGC is coordinated through a Joint Advisory Group (JAG). For example, the candidate OGC implementation standard for a gazetteer service (OGC, 2006a) is based on the conceptual model for gazetteers as defined in ISO 19112:2003, *Geographic information – Spatial referencing by geographic identifiers*. Another example is ISO 19128:2005, *Geographic information - Web Map Service Interface (WMS)*, which was originally developed in the OGC as part of the OGC Interoperability Testbed program. WMS was then submitted to ISO/TC 211, further developed, and is now published by the OGC as *OpenGIS® Web Map Server Implementation Specification 1.3.0* (OGC, 2006b).

In contrast to ISO, the OGC does not have a formal review of a given standard based on some time period. Rather, if an implementer or developer encounters a problem or has a suggestion for an enhancement or change, the OGC has a formal change or requirement request submission process, which is available to the public on the OGC website. These requests are used to trigger the revision process for a given OGC standard. For example, there are now seven outstanding change requests for KML (used in Google's mapping applications) and OGC is forming a new KML Standards Working Group (SWG) to manage the revision. Once the SWG is formed, the normal OGC standard development process commences, as described in the Technical Committee Policies and Procedures (OGC, 2010). This may include collaboration with ISO/TC 211 if appropriate. In practice this process results in constant review: an average life cycle of only a few years between standards version, comparable to ISO's three year systematic review.

There are many dependencies between standards from ISO/TC 211 and OGC. An analysis of dependencies of geographic information standards showed 167 normative references from one ISO/TC 211 standard to another, and 3 from an ISO/TC 211 standard to an OGC standard. A similar pattern applies within the OGC standards. A change to the normative content of one standard potentially affects dependent standards, therefore these interdependencies have to be carefully managed when changes are introduced in revisions of standards.

UML models and XML schemas are associated with many of the standards and also require careful management when standards are revised. As a result of revision there could be more than one version of each UML package and/or XML schema, so there is a risk that an implementation of SDI tools or a deployed infrastructure that claims conformance to a set of standards may be internally inconsistent, depending on the particular combination of versions of standards adopted. The Harmonized Maintenance Management Group (HMMG) of ISO/TC 211 maintains an integrated and consistent model of all UML elements in ISO 19100 standards. Everyone has read access to the model on the ISO/TC 211 website. The HMMG receive frequent comments on the model, suggesting that it is widely used in implementations. The XML Maintenance Group (XMG) in ISO/TC 211 was recently established to maintain the XML schemas produced in ISO/TC 211 standards.

Configuration management (CM) is the discipline of identifying the configuration of a system at distinct points in time for the purpose of systematically controlling changes to the configuration, and maintaining the integrity and traceability of the configuration throughout the system life cycle (Bershoff, 1997). Software configuration management (SCM) is one of the software engineering knowledge areas defined in The Guide to the Software Engineering Body of Knowledge (SWEBOK) (Abran et al., 2004) of the IEEE Computer Society. The SCM activities are: management of the SCM process, software configuration identification, software configuration control, software configuration status accounting, software configuration auditing and software release management and delivery. It could be argued that a system of interdependent standards is not a software system, as no executable software development takes place. However, most ISO/TC 211 standards describe some aspect of a software system (also evident in the UML and XML artefacts) and therefore the principles of software configuration management apply.

In this paper we explain configuration management challenges for a 'system' of interdependent standards, which are expressed as standard documents, a UML model and XML schemas. Subsequently, we present two conceptual models and propose improved configuration management activities for ISO/TC 211. We conclude with a summary of results that are equally applicable to other ISO technical committees.

CONFIGURATION MANAGEMENT CHALLENGES FOR A SYSTEM OF INTERDEPENDENT STANDARDS

In recent years there has been a substantial increase in the use of machine-readable artefacts in association with ISO/TC 211 and OGC standards. The UML model or parts thereof are, for example, used in model-driven development and XML schemas are used in various ways, such as for data validation. In most cases a standard has dependencies on other standards, therefore the overall coherence of the set of elements in a standard has to be considered, including the UML and XML components. This becomes particularly challenging when one or more standards are revised so that dependencies on previous and new versions get entangled. It became clear that a study of the issue related to the specific artefacts used in ISO/TC 211 standards was required in order to provide guidance to the various ISO/TC 211 coordination groups, project teams and editors.

The issue was first raised in OGC, where many of the standards had a generous scope, necessitating a ‘profile’ for practical implementation. The mechanism for restricting a general standard to a more specific subset was unclear, and implementations were sometimes not interoperable. An alternative model based on ‘core and extensions’ was proposed, which was then formalized into the ‘requirements-classes with dependencies’ model discussed below. A flow-on benefit has been an improvement in document quality. Previously, every OGC standard was specifying requirements and abstract conformance classes differently. This made compliance and interoperability testing extremely difficult. It is impossible to build a test suite for a standard if the requirements are not clear.

The ISO/TC 211 Harmonized Model Maintenance Group (HMMG) is responsible for the consistency of UML models across the suite of ISO/TC 211 standards. This is achieved by integrating UML from all the standards into a single harmonized UML model, the *Harmonized Model*. To date, harmonization has been substantial but incomplete: editors have often failed to make the models from their standards available to the HMMG in a timely manner so harmonization ahead of publication was not always possible. In addition, there are no detailed guidelines or best practices for editors on how to prepare the UML. Another complicating factor is the wide range of levels of detail and abstractions that the model exhibits. Some standards contribute UML to the Harmonized Model for ontological purposes only, while others contribute UML that represents an actual data record implementation. Since there are currently no rules or standards, unless a standard contributes UML that violates the rules of UML, or contributes UML that is flawed to some extent that prevents it from being integrated, it is accepted into the Harmonized Model.

Meanwhile, developments in the capability and affordability of tools for editing and storing UML models have made online collaboration and synchronization feasible. The Harmonized Model is now maintained as a set of XML Model Interchange (XMI) files in a single Subversion repository, to which all authorized parties have access. Easier access to the Harmonized Model has, however, made gaps and inconsistencies in the current state of the model visible to a wider audience. The community has a quite reasonable expectation of being able to access a consistent model state, and the teams preparing standards have a role in helping this come to pass.

A key issue here is the granularity of dependencies. ISO standards normatively reference each other as a whole. However, UML components are packaged in various ways, including classes, leaf packages, and packages containing other packages. While one standard (the dependent standard) might normatively reference another (the origin standard), a change to a specific UML package or class in the revision of the origin standard might not have an effect on the dependent standard at all. The same applies to other normative content in the standard. This shows that ISO standards do not provide dependency information at the granularity required for configuration management. The relationship between the re-usable elements in the UML representation and the elements within the document form of the standard also needs to be clarified. Finally, there is likely to be a requirement for access to prior consistent states of the Harmonized Model.

The key XML artefacts are XML Schema (XSD) and Schematron documents (containing XML constraints and validation rules) along with some XML instance documents that represent common resources such as code-lists and vocabularies. The ISO/TC 211-maintained XML resources are currently stored in multiple repositories and it is not clear which one of them is authoritative. The externally visible packaging for schemas is the XML namespace, identified by URI. However, most XML processors refuse to load the same namespace from different locations, even if the documents are identical. This is consistent with the provisions of the XML Schema and XML Namespace specifications (W3C, 2004; W3C, 2009), but effectively makes clones of a schema problematic. OGC established a set of rules for maintaining an XML repository over the last five years. This is not surprising as OGC has been dealing with XML much longer than ISO/TC 211. OGC addressed the repository issue through a somewhat fierce (but it seems unavoidable) policy that the 'schema location' is normative, as well as the XML namespace. In the context of the configuration management problem, it is necessary to define the relationship between the XML components and the corresponding UML and document elements. The ISO/TC 211 XML Maintenance Group (XMG) is responsible for establishing policies relating to XML resources, and it is expected to make recommendations in this area, including versioning, in a way that is consistent with the mappings to the other representations (Cox, 2011).

Finally, the ISO/TC 211 Program Maintenance Group (PMG) is responsible for overseeing the scope of projects within ISO/TC 211 to ensure harmonization and consistency of standards. The PMG does this by collecting and publishing inconsistencies in standards; by advising the committee to initiate amendments, corrigenda and revisions; and by monitoring the scope of new projects. This touches on a number of the activities of software configuration management, namely controlling change in a system, accounting on the status of the system configuration, and maintaining integrity and traceability of the system configuration.

CONCEPTUAL MODELS FOR CONFIGURATION MANAGEMENT OF A SYSTEM OF INTERDEPENDENT STANDARDS

The first model in this section represents normative dependency relationships between standards. It shows the different kinds of normative dependencies that can exist between ISO standards written according to the ISO/IEC Directives (ISO/IEC, 2004). The second model proposes a new structure for standards, focusing on their internal structure, that will allow improved configuration management. Both models are important because it will take time (a number of years at least!) to convert existing standards into the new structure.

An ISO standard consists of normative and informative elements. Normative elements describe the scope of the document and set out provisions of the standard, while informative elements provide additional information to introduce the content, explain the background and assist in understanding and using the document (ISO/IEC, 2004). Only changes to the normative content of a standard can possibly have an impact on other standards. Therefore, a normative dependency can be defined as *a relationship between two standards where a change to the normative content in the one standard affects the normative content of the other standard* (Coetzee, 2011). For configuration management of a system of standards, the normative dependencies between standards have to be managed to ensure integrity of the system.

Figure 1 shows a UML conceptual model for the representation of normative dependencies between standards. This conceptual model is the blueprint for the normative dependency database that is maintained by the ISO/TC 211 PMG (ISO/TC 211 PMG, 2011). Dependencies among published ISO/TC 211 standards, as well as standards under development are included. In the model the Edition class represents a specific edition (publication) of a document and thus always has a year of publication associated with it. If there is more than one edition of the same document, the editions are associated with a single document. For example, ISO 19111 is an instance of the Document class and ISO 19111:2003 and ISO 19111:2007 are two instances of the Edition class, associated with the ISO 19111 document.

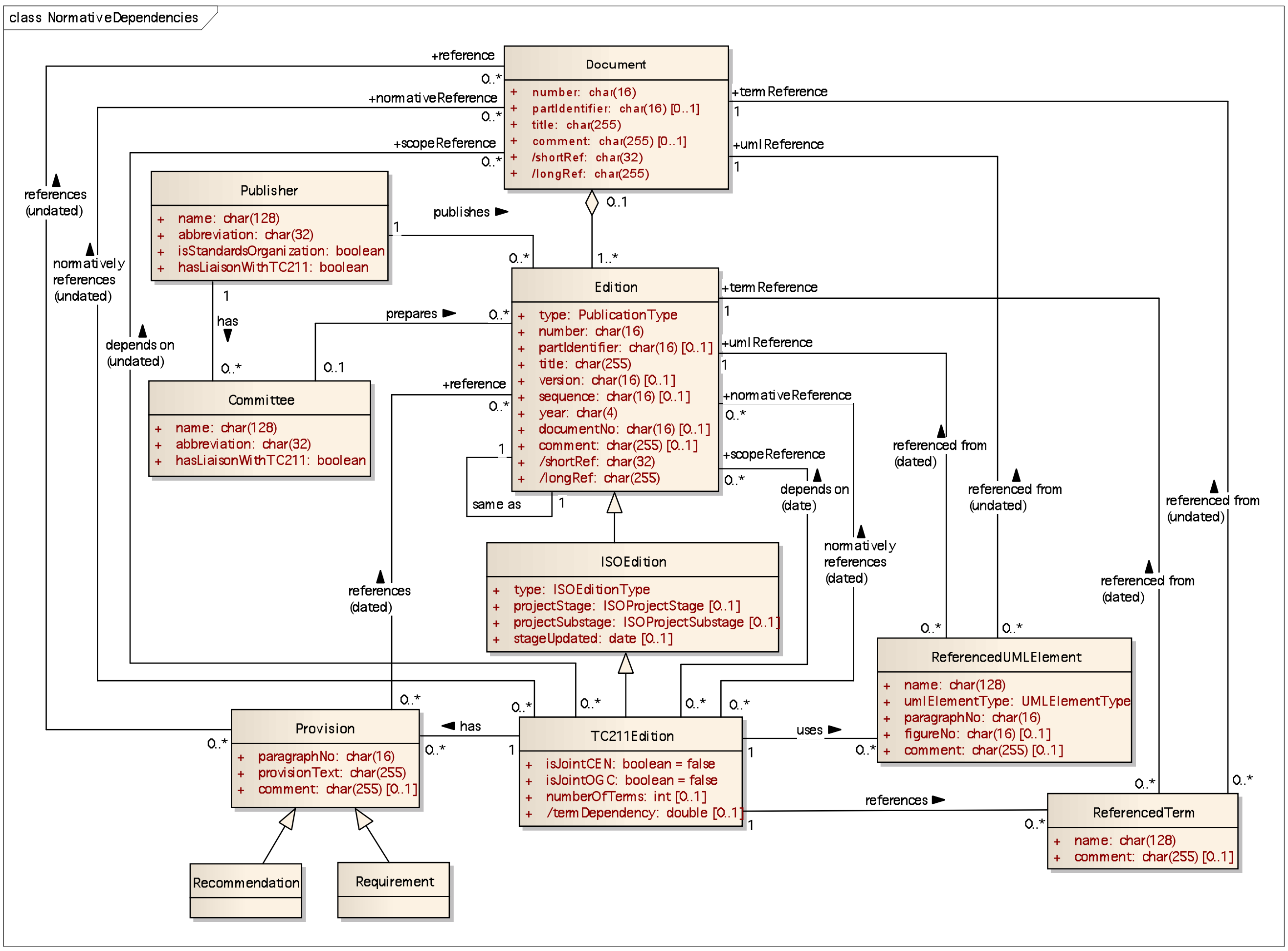

Figure 1. Conceptual model of normative dependencies in ISO/TC 211 standards

The reflexive 'same as'-association of the Edition class represents standards that are developed and published jointly with other organizations, such as ISO 19128:2005, which is equivalent to OpenGIS WMS 1.3.0. ISOEdition and TC211Edition are derived from the Edition and ISOEdition classes respectively, each defining additional ISO- and ISO/TC 211-specific attributes. A modification of an edition, such as a supplement, corrigendum or amendment, has the same associations as the TC211Edition and Edition classes, but these are not included in the diagram for the sake of simplicity.

Associations between the classes represent the normative dependencies. For example, the association labeled ‘normatively references (dated)’ between Edition and TC211Edition shows that an instance of the TC211Edition normatively references zero or more instances of the Edition class; and an instance of the Edition class is normatively referenced in zero or more instances of the TC211Edition class. Currently, the ISO/TC 211 dependency database includes normative dependencies on scope, normative references and terms. Normative dependencies on provisions and UML elements are not included because the process of identifying all provisions and references to UML elements in the text is time consuming, error prone and potentially ambiguous. Consider, for example, the paragraph from ISO 19111:2007:

‘A *coordinate tuple* is an ordered list of *n* coordinates that define the position of a single point. In this International Standard the coordinate tuple shall be composed of one, two or three spatial coordinates. The coordinates shall be mutually independent and their number shall be equal to the dimension of the coordinate space.’

The first sentence is a requirement but is not easily identifiable by ‘shall’ or a similar normative word in the sentence. The second and third sentences each contain the word ‘shall’, which indicates a normative provision. Another question is whether the paragraph constitutes a single requirement or three separate requirements (one per sentence)? It might even be part of a ‘bigger’ requirement.

Cyra et al. (2011) concur with our observation that standards are sometimes large text documents that are difficult to interpret. They propose a framework for achieving and assessing conformance to such standards. Their solution includes an analysis of the interrelationships between the individual requirements within a single standard.

While dependencies on scope, normative references and terms are important, dependencies on provisions and UML elements provide more detailed information, which is required for configuration management. The second model addresses this aspect, proposing a formalization for a standard that will alleviate not only dependency management, but also configuration management as a whole. This model is designed on the premise that a standard is a partial solution to a design problem, which overall is generally solved in multiple standards, or perhaps by use of only some provisions from a standard. The standard restricts conformant solutions in order to enhance interoperability and harmony between disparate implementations. Design issues and requirements are transformed into statements about the solution design, and are then presented in the standard as requirements of the solution, usually as requirement statements targeting the solution.

Thus, a standard presents requirements that are targeted towards implementations of solutions of the original problem, and that must be satisfied by passing the tests of the conformance suite. In our proposed model, these tests are organized into conformance classes, each of which represents a mechanism for partial satisfaction of the standard. Conformance classes give the standard a modular structure, where each requirements class is associated with a single conformance class. This design for a modular specification was developed in OGC (2009) and is now being considered in ISO/TC 211. If both organizations decide to follow a compatible modular design of their standards, this will be a huge benefit to the configuration management of the joint collection of geographic information standards.

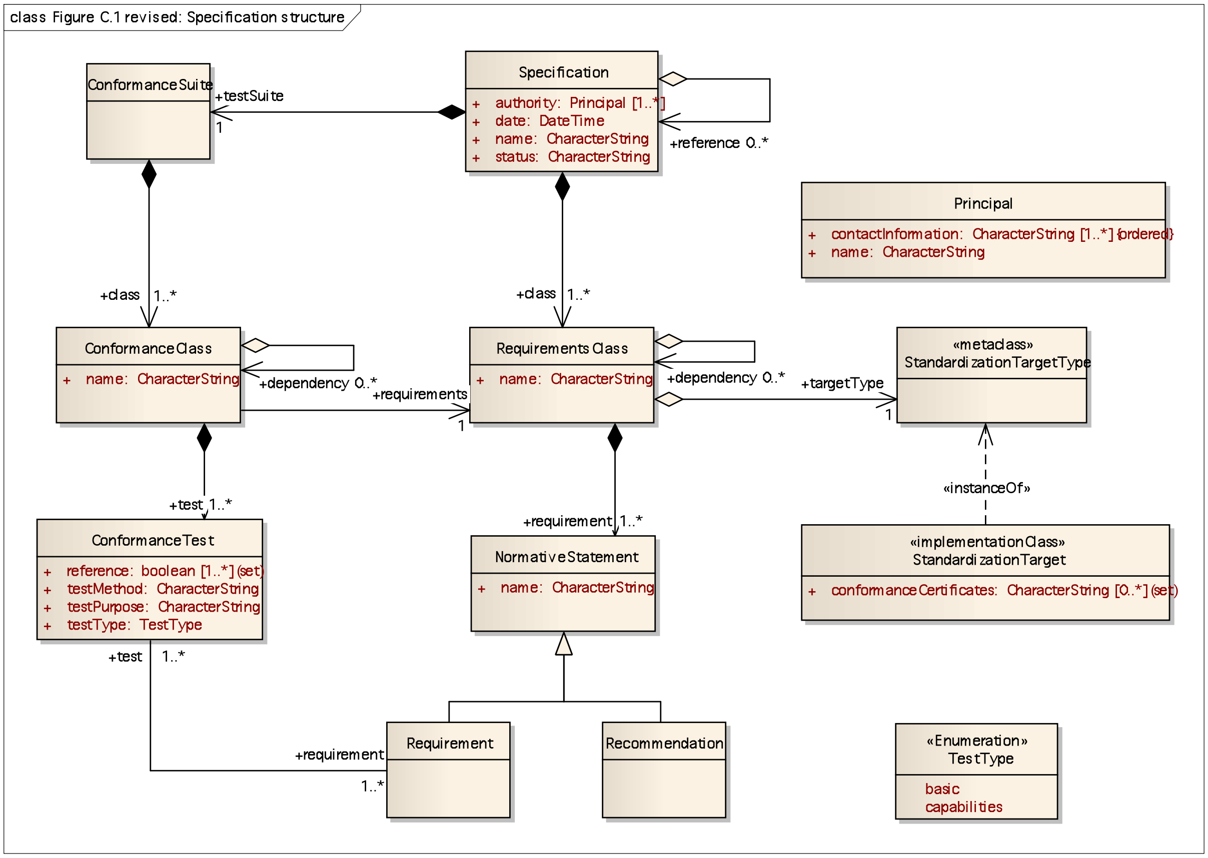

Figure 2. Conceptual model of a modular standard (from OGC, 2009)

Figure 2 shows the proposed conceptual model for the new structure of a standard:

- Provisions (NormativeStatements) in a specification are grouped into Requirements Classes.
- There can be dependencies from one Requirements Class on another, from the same or other standards.
- Conformance tests are organized into a number of Conformance Classes. If a standard does not do this, it has, by default, only one conformance class.
- There can be dependencies from one Conformance Class on another.
- There is a one-to-one association between a Conformance Class and a Requirements Class.
- All tests for conformance to the requirements of a standard are defined in the Conformance Suite. Every requirement must be tested.
- For convenience, tests may be grouped into a hierarchy of conformance test modules, as described in ISO 19105:2000.
- Each test, if conducted, will determine to some degree of certainty whether an implementation meets the requirements which the test references.
- Each Requirements Class concerns a single target type.

- No requirement in a class is optional. Every test in a conformance class must be satisfied for an implementation to be certified conformant (this is not strictly shown in the diagram).

The model requires a clear distinction between normative and informative parts of the text in a standard; and that each requirement in a standard is identifiable as a requirement. In OGC and ISO, this means use of normative language, meaning the proper use of 'shall', 'should', 'can' and 'may' or similar wording in the passive voice (ISO/IEC, 2004).

An additional benefit of the proposed model is that certificates of conformance can now be awarded by an independent testing entity based on the results of testing the conformance classes.

In a related example, Walker (2011) analyzed and compared the provisions of a number of quality and process-related standards, including ISO 9001:2008, in order to identify process activities and informational items that are included in one standard but not the other. The results of the analysis are used to improve the standards but also to assist implementers of more than one of these standards in understanding how the standards are related to each other. Individual provisions had to be identified manually and the analysis would have been a lot easier with requirements and conformance classes in the different standards.

RECOMMENDATIONS TO IMPROVE CONFIGURATION MANAGEMENT IN ISO/TC 211

In this section we propose improved configuration management activities for the ISO/TC 211 system of interdependent standards, based on the conceptual models presented in the previous section. To facilitate the discussion, we refer to the system as 'S19100'. The proposal is structured according to the SWEBOK software configuration management activities to ensure that all aspects of software configuration management are covered. Figure 3 provides an overview of the SWEBOK software configuration management activities.

Management of the SCM process

The first activity is concerned with managing the configuration by establishing and maintaining an SCM plan that fits the organizational context and constraints. Part 1 of the ISO/IEC Directives (ISO/IEC, 2009) describes the *organizational context* of S19100: the organizational structure of ISO and its technical committees, the ISO project approach to standard development, the types of standards that are published and how meetings are conducted. The S19100 configuration has to be managed within the *constraints* of these ISO processes. OGC has its own set of rules and it is the responsibility of the ISO/TC 211 JAG group to ensure smooth communication between the two organizations.

The ISO/TC 211 member bodies nominate *resources* (project leaders and experts) for participation in a project. The secretariat keeps track of who works on which project. The *responsibilities* of the project leader, editor, project experts and others in the technical committee (chair, secretariat, etc.) are clearly defined in the ISO/IEC Directives, Part 1 (ISO/IEC 2009). In addition, in ISO/TC 211 the PMG, HMMG and XMG take responsibility for maintenance of the overall program and advice on the sequencing of projects, the Harmonized Model and the XML repository, respectively. These three play an important role in the configuration management process and provide guidance to project teams on *tools* to assist with configuration management. To find the required resources (e.g. website hosting or software maintenance) for the implementation of these tools is sometimes a challenge.

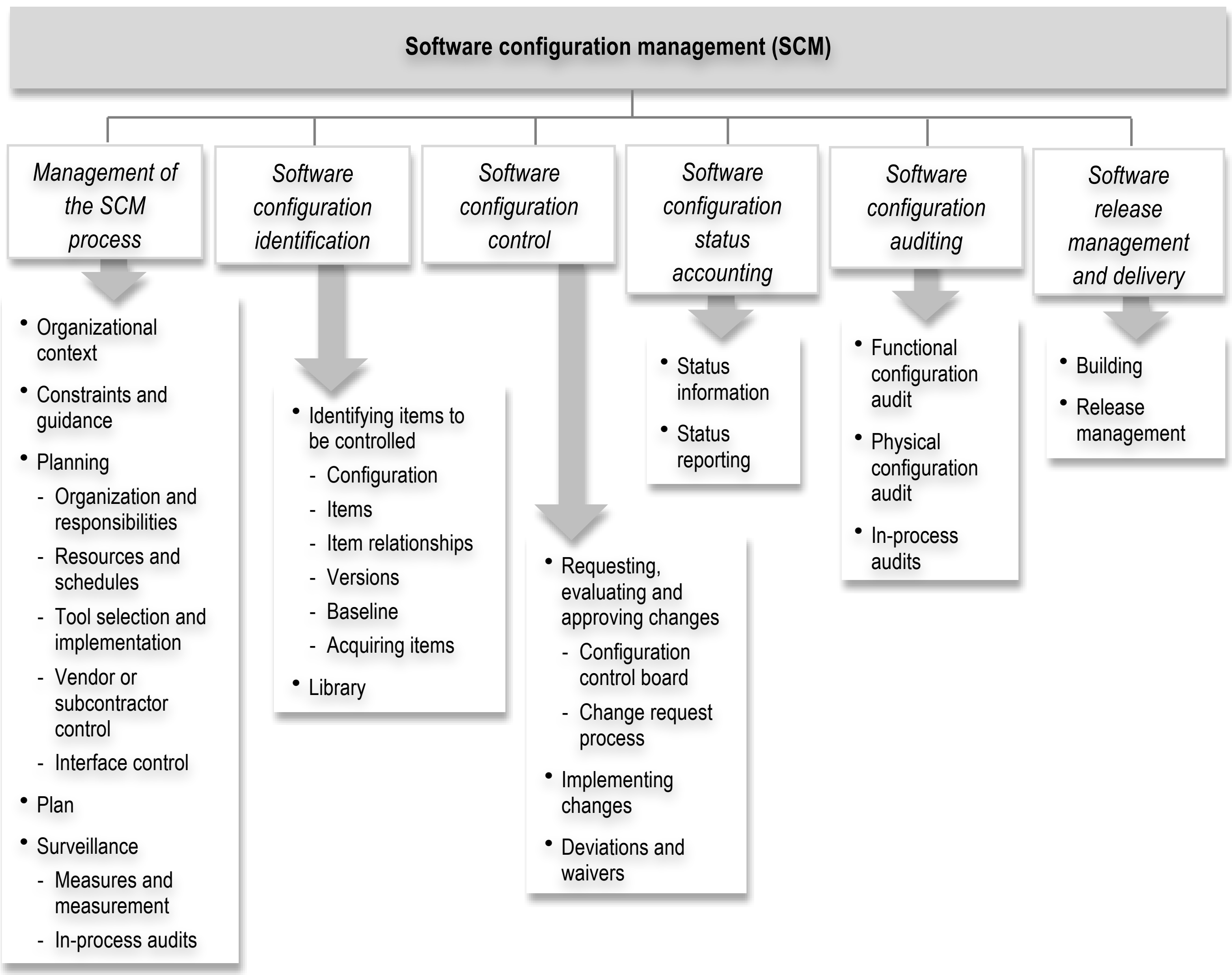

Figure 3. Software configuration management activities (adapted from Abran et al., 2004)

The copyright of all drafts and published standards belongs to ISO. However, the *vendor or subcontractor control* activity applies to material in a standard that may originate from other sources (ISO/IEC, 2009).

To date, *interface control*, i.e. identifying interfacing items and how changes to these will be communicated, has been left to individual project teams. We have already illustrated in the previous section that a typical ISO standard does not provide enough dependency information and that the requirements and conformance classes in the proposed conceptual model for a modular specification improves this situation. We further recommend that changes to the normative elements of a standard are clearly identified in the revision of a standard through all stages of its development.

The project *plan* is adjusted at a plenary meeting (typically, twice a year). See for example, the adjustment of the project schedule in ISO/TC 211 Resolution 509, *The adjustment of target dates of the ISO/TC 211 programme of work* (ISO/TC 211, 2010).

Surveillance of configuration management lies mainly with committee members, assisted by the PMG, HMMG, XMG and a fourth maintenance group, the ISO/TC 211 Terminology Maintenance Group (TMG). There is an opportunity for surveillance whenever a standard is distributed to the committee, for example, for comment or for a vote at the end of one of the ISO project stages. Such reviews by committee members serve as quality assurance on the system configuration.

Software configuration identification

This activity identifies items to be controlled, establishes identification schemes for the items and their versions, and establishes tools and techniques to be used in acquiring and managing controlled items.

An *item* is an aggregation of individual parts of a system and is treated as a single entity in the configuration management process. We propose that not only the standards, but also all the normative elements, i.e. the scope, normative references, terms, as well as conformance and requirements classes, are treated as items in the S19100 configuration management. The *relationships* between these items, i.e. the dependencies, should be recorded explicitly in a standard.

In the Harmonized Model, the configuration item is a package. A requirements class is represented by a single UML package. In XML the configuration item is an XML namespace. Similarly, a requirements class is associated with a single namespace and each XML namespace contains components from a single requirements class. To assist identification and cross referencing, each requirements class, conformance class, UML package and XML namespace should have its own unique Uniform Resource Identifier (URI).

The ISO harmonized stage codes (ISO 2011) provide proper *versioning* for standards. Revisions, amendments and corrigenda are dealt with as separate projects. A software baseline is a set of software configuration items formally designated and fixed at a specific time during the software life cycle. The S19100 *baseline* changes whenever ISO publishes a new standard. The HMMG and XMG set the rules and policies for *acquiring items* into the Harmonized Model and XML repository, respectively. This should occur as close as possible to the time of the publication of the standard. The HMMG and XMG should also develop processes that provide access to consistent states of the Harmonized Model and XML repository.

The ISO/TC 211 secretariat maintains a comprehensive documentation *library*, the so-called ‘document register’ (ISO/TC 211, 2011). Proposals, versions of the documents at different stages, resolution of comments, notices of meetings, etc. are stored there.

Software configuration control

This activity is concerned with managing changes during the software life cycle, including determining changes to be made, approving changes, support for implementing the changes, as well as formal deviations from the project requirements or scope.

In ISO *changes* to standards are proposed in a new work item proposal (NWIP) for the revision of or amendment to a standard. Any member body may submit such a proposal. The committee votes on an NWIP and thus exercises its right to *configuration control*. The consensus building process during standard development further influences configuration control. The maintenance groups PMG, HMMG, XMG and TMG do not have a vote, but may submit comments and thus influence configuration control. The PMG also supports configuration control with a database of harmonization issues that are submitted as comments to the relevant ballots. Currently, these issues are recorded in an HTML page and there is room for improvement by, for example, using a software tool to record and track the issues. The PMG also advises the committee on the sequencing of standard development and whether revisions or amendments should take place.

If at some stage during standard development it is necessary to modify the scope of the standard, the ISO process prescribes that a ballot or vote by the committee should decide about the modified scope. In this way *deviations* from the scope are accommodated.

Software configuration status accounting

This activity comprises the recording and reporting of information needed for effective management of the software configuration. The ISO and the ISO/TC 211 websites provide *status information* about standards, but additional information at a lower level of granularity (UML package, XML namespace) should be made available. For this, dependencies at a lower level of granularity (below a standard) have to be explicitly documented in a standard. This will also allow automated updates of the ISO/TC 211 dependency database, a powerful *status reporting* tool for configuration management.

Software configuration auditing

This activity independently evaluates the conformance of software products and processes to applicable regulations, standards, guidelines, plans and procedures. While there is not an independent organization to perform this evaluation for an ISO committee, comments from a large variety of member bodies could be seen as an independent evaluation.

Software release management and delivery

In this activity 'release' refers to the distribution of a specific configuration item outside the development activity. This would typically happen whenever ISO publishes an international standard, when the Harmonized Model is published, or the publicly available XML repositories are updated. The Harmonized Model and XML repositories should be updated as soon as possible after the publication of the standard.

SUMMARY OF RESULTS

In order to implement the improved configuration activities, it is recommended that ISO 19105:2009, *Geographic information – Conformance and testing* is revised to include the proposed conceptual models from this paper, and that the revision also incorporates the UML and XML configuration management requirements outlined in this paper. Consequently, it will be necessary to review the Harmonized Model and restructure its elements into packages as proposed. The same applies to the XML repository.

The modular specification approach was developed by the OGC Policy working group. From 2010 all revisions of existing OGC standards follow the policies as stated in the modular specification (OGC, 2009) outlined here. To date, ten standards published by the OGC in the last year have followed the model. Initially there was resistance to change from standards editors, mostly about the more rigorous documentation requirements, but in all cases after re-formulating the work to follow the model the assessment of those editors was that the standard was significantly improved by having followed the model. In cases where the editor had to refactor a non-modular form there was inevitably a delay, but where the modular structure had been built in from the start, it provided a more complete template than editors had previously had available. There is now general acceptance of the improved approach.

Initial indications are that well formed and well-stated requirements are having a very positive effect on 1) readability by any developer and 2) development of test suites for new OGC standards and revisions to OGC standards.

A significant limitation of the model is that although the model specifies that dependencies should be between Requirements Classes, in the common case where the dependency standard is not formalized into Requirements Classes (e.g. an existing standard in the narrative style or a standard published by a different organization), a surrogate element (usually a sub-clause) from the dependency must be used to approximate this.

Our recommendations for the improved software configuration management activities in ISO/TC 211 are equally applicable to other ISO technical committees:

- Configuration items should not be limited to whole standards, but should include other items at a lower granularity level, such as scope, terms, conformance and requirements classes, UML packages and XML namespaces.
- Dependencies between the configuration items in a standard should be recorded explicitly, e.g. in a dependency table.
- Changes to the configuration items should be clearly identified in the revision of a standard.
- Automated tools should be used for configuration status reporting and tracking.

CONCLUSION

Modular specification is a simple and widely accepted principle in software configuration management but sadly often neglected by standards editors who do not have training in software engineering. In this paper we explained the configuration management challenges for a system of interdependent standards, which are expressed as standard documents, a UML model and XML schemas. We presented models in support of understanding configuration management of this system and proved that ISO standards do not provide dependency information at a granularity required for configuration management. We proposed improved software configuration activities for ISO/TC 211 and concluded with recommendations for improved configuration management activities that are equally applicable to other ISO technical committees. Initial results show that standards developed according to the modular specification approach are easier to read and test, resulting in both tangible and intangible economic benefits. Results from more implementations of this approach are required in order to better understand the general applicability of the approach. Feedback from these implementations could recommend refinements or alterations to the modular specification approach, if required.

ACKNOWELDEGEMENTS

The authors acknowledge the contributions to the work on standard maintenance and configuration management from various members of the ISO/TC 211 committee, specifically from members of the *Programme Maintenance Group* (PMG), the *Harmonized Model Maintenance Group* (HMMG) and the *Ad hoc Group on Configuration Management*.

The authors would like to thank Carl Reed from the OGC for his comments on an early draft of this paper.

Serena Coetzee would like to thank the South African Bureau of Standards (SABS) for their travel support as a member of the South African delegation to meetings of ISO/TC 211, *Geographic information/Geomatics*.

Simon Cox's work on this topic has been supported by the European Commission Joint Research Centre, by CSIRO and by the Water Information Research and Development Alliance.

REFERENCES

Abran, A., Moore, J.W., Bourque, P. and Dupuis, R. (Eds). (2004). *Guide to the Software Engineering Body of Knowledge*, IEEE Computer Society, US.

Bershoff, E.H., (1997). Elements of Software Configuration Management. In *Software Engineering*, Dorfman M. and Thayer R.H., (Eds.), IEEE Computer Society Press.

Coetzee, S. (2011). Results from a normative dependency analysis of geographic information standards. *Computer Standards and Interfaces*, DOI: 10.1016/j.csi.2011.02.004.

Cox, S. (Ed.). (2011). *Draft report by the ISO/TC 211 Ad hoc group on configuration management and backward compatibility*. To be published by ISO/TC 211, *Geographic information/Geomatics*.

Cyra, L. and Gorski, J. (2011). SCF – A framework supporting achieving and assessing conformity with standards. *Computer Standards and Interfaces* **33**(2011), pp80–95.

Directive 2007/2 of the European Parliament and of the Council of 14 March 2007 establishing an Infrastructure for Spatial Information in the European Community (INSPIRE) (2007). Retrieved April 20, 2011, from http://inspire.jrc.ec.europa.eu/index.cfm/pageid/3.

International Organization for Standardization, (2000). ISO 19105:2000, *Geographic information – Conformance and testing*.

International Organization for Standardization, (2003). ISO 19111:2003, *Geographic information – Spatial referencing by coordinates*.

International Organization for Standardization, (2007). ISO 19111:2007, *Geographic information – Spatial referencing by coordinates*.

International Organization for Standardization, (2003). ISO 19112:2003, *Geographic information – Spatial referencing by geographic coordinates*.

International Organization for Standardization (2008). ISO 9001:2008, *Quality management systems – Requirements*.

International Organization for Standardization and International Electrotechnical Commission, (2009). *ISO/IEC Directives, Part 1, Procedures fro the technical work*. Seventh edition.

International Organization for Standardization and International Electrotechnical Commission, (2004). *ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards*. Fifth edition.

International Organization for Standardization, (2011). *Supplement – Procedures specific to ISO*. Second edition.

ISO/TC 211 (2010). *Resolutions from the 30th ISO/TC 211 plenary meeting in Southampton, United Kingdom, 2010-05-27/28*. ISO/TC 211 document number N2945.

ISO/TC 211 (2011). *Document register*. Retrieved April 20, 2011, from http://www.isotc211.org/docreg/sok?action=docreg1.

ISO/TC 211 website. Retrieved April 20, 2011, from www.isotc211.org.

ISO/TC 211 PMG website. Retrieved April 20, 2011, from www.isotc211.org/pmg.

Open Geospatial Consortium, (2006a). *Gazetteer Service, Profile of the Web Feature Service Implementation Specification*. http://www.opengis.net/doc/wfs-gaz/0.9.3

Open Geospatial Consortium, (2006b). *OpenGIS Web Map Server Implementation Specification 1.3.0*. http://www.opengis.net/doc/wms

Open Geospatial Consortium (2009). *The Specification Model – A standard for modular specifications*. Retrieved April 20, 2011, from http://www.opengis.net/doc/modular-spec

Open Geospatial Consortium, (2010). *Technical Committee Policies and Procedures*. Retrieved April 20, 2011, from http://www.opengeospatial.org/ogc/policies

Walker, A.J., (2011). *Evaluating the ISO TMB Management System Common Text proposal in terms of selection of candidate management system standards*. SPI Laboratory (Pty) Ltd, Johannesburg, South Africa, 2011.

World Wide Web Consortium (W3C), (2009). Namespaces in XML 1.0. W3C Recommendation 8 December 2009. Retrieved April 20, 2011, from http://www.w3.org/TR/xml-names/.

World Wide Web Consortium (W3C), (2004). XML Schema Part 1: Structures. W3C Recommendation 28 October 2004. Retrieved April 20, 2011, from http://www.w3.org/TR/xmlschema-1/.