Implementing E-learning Specifications with Conformance Testing: Profiling for IMS Learning Design

Implementing E-learning Specifications with Conformance Testing: Profiling for IMS Learning Design

Owen ONeill, Rob Nadolski, Rob Koper
Educational Technology Expertise Centre
Open University of the Netherlands
{owen.oneill, rob.nadolski, rob.koper}@ou.nl

Abstract

Improving interoperability between e-learning systems and content has been one of the driving forces behind the adoption of e-learning specifications over recent years. A vital step towards achieving this goal is the widespread adoption of conformant implementations of e-learning specifications. A conformant implementation is one which fully complies with the conformance requirements of the specification. However, conformance testing is time consuming and expensive. The process of localising specifications to create so-called "Application Profiles" to meet individual community needs further complicates conformance testing efforts. To solve this problem, we developed the conformance testing approach presented in this article. This approach simplifies the development of Application Profiles, and the process of conformance testing against them. Using this approach, test suites can be generated to test software applications against both e-learning specifications and their derived Application Profiles. A case study based around the IMS Learning Design specification demonstrates this process.

Key words: Conformance testing, e-Learning specifications, Application profiling, IMS Learning Design, interoperability

Introduction

The growth in popularity of e-learning specifications has been driven by a number of motivators, including the need to share and re-use content more seamlessly in different contexts. However, without standardization, e-learning content can be prohibitively expensive and time-consuming to repurpose, store and use in different contexts. A major aim of e-learning specifications is to provide standardization for both content and systems. For example, when a content developer creates content, s/he should be confident that the content will run in any system that has implemented the e-learning specifications the content was based on. Likewise, software developers need to be sure that the e-learning applications they develop will accept and run conformant e-learning content correctly. Teachers and students should also be able to expect to use high quality e-learning systems and content.

Conformance testing is a widely used tool in industries such as telecommunications (Malek & Dibuz, 1998; Hao, et al., 2004) for testing implementations against standards and specifications. For e-learning content and systems also, existing conformance testing techniques can be applied. However, unlike the telecommunications industry where standards generally must be strictly adhered to, e-learning specifications typically allow a much greater degree of freedom in how the specifications are implemented. The diversity of e-learning community needs, and the different permutations of implementations allowed by e-learning specifications contribute to make conformance testing more complex in this domain. The practice of creating "Application Profiles" or localized implementations of e-learning specifications is a common method used to cater to community-specific requirements, yet it poses a number of challenges for testing specification conformance.

In response to these issues, the Telcert (TELCERT, 2005) project developed a set of software tools for use in conjunction with established conformance testing principals. The Telcert Project is a European Union-funded 6th Framework research and technology development project involving ten partners across Europe including many European IMS partners. These tools enable the creation of e-learning specification Application Profiles. A conformance testing system has also been adapted to enable conformance testing against both the localised implementations and the base specifications themselves. This article presents an approach which uses these tools in conjunction with established conformance testing techniques to simplify e-learning specification conformance testing. The case study presented in this article illustrates this approach with profiling work based around the IMS Learning Design specification.

The next section introduces a number of existing conformance testing techniques and practices with an emphasis on their relevance for e-learning specifications testing. Section 3 introduces the set of tools which were developed to enable the creation of conformant Application Profiles and testing against base specifications and derived application profiles. To test and demonstrate the use of these tools and processes in practice, a case study is presented in Section 4. The case study focuses on specific IMS specifications, but the tools and processes are designed to be widely applicable to XML schema-based specifications. Results from this case study are provided in Section 5. The paper concludes with a discussion of the findings, and indicates future work for moving further towards interoperable e-learning implementations.

Conformance Testing

The concepts of interoperability and conformance are related but distinct. The process of conformance testing aims to determine whether an implementation conforms to the specification(s) it was based upon (Malek & Dibuz, 1998). In contrast, interoperability can be defined as the ability to exchange and use information across different systems (Kindrick, 1996). To enable seamless re-use and exchange of information, interoperability of e-learning specification implementations is needed. However to reach interoperability, specification conformance is an essential step (Hao et al., 2004) which has not yet been reached.

Tests are often divided into two types in the software testing world. White-box testing is concerned with testing the internal structure of a piece of software. This type of testing is also called structural testing. Black-box or functional testing, treats the system as a "black box" – there is no consideration of the inner workings of the system under test. Testing conformance to a specification does not imply any particular run-time behaviour of the system under test and therefore black box testing is suitable.

In order to produce effective test suites using a minimal number of test cases, we utilized a number of established conformance testing techniques. Using the principals of the Category Partition method (Ostrand, 1988), it is possible to specifically focus on elements that are more likely to produce errors. In this way, test cases can be 'partitioned' to minimize the number required. Boundary value analysis is another test data selection technique where values are chosen which lie along data extremes (Ramachandran, 2003), as these are considered more likely to return an error. Boundary values include maximum, minimum, just inside/outside boundaries, typical values, and error values. Tests that perform correctly on these higher risk values should also perform correctly for all values in between. Utilising these techniques enables the number of required test cases to be kept to a minimum, and therefore limits the amount of test content that must be produced.

Conformance Testing with e-Learning Application Profiles

As mentioned previously, the heterogeneity of the e-learning community sometimes results in specifications such as IEEE Learning Object Metadata (IEEE LOM) and IMS Content Packaging

(IMS CP) being extended and localised to accommodate the needs of disparate e-learning communities. Examples of such modifications include:

- A subset of elements within a specification may be used, with remaining elements discarded;
- Restrictions applied to certain values, such as fixed or default text values;
- Extensions may be used to add new elements not found in the base specification.

In the context of such modifications. An XML schema-based Application Profile may initially be partitioned into the following basic categories:

- XML well-formedness testing
- XML schema testing
- Non-XML schema testing

XML Well-formedness testing
In order for an XML document to be valid, there are a number of rules that it must conform to. For example, XML documents must contain a single root element. Tags in an XML document must also be correctly nested, and every open tag must have a corresponding closing tag.
Checking for well-formedness in XML documents can be done using a number of well-established XML parsers.

XML Schema testing
An XML document instance must also be checked for conformance against the XML schema(s) it was based upon. Again, established XML parsers can check such conformance. Although there are occasionally some discrepancies between parsers, such discrepancies are generally minor, and are not a focus in this article.

Non-XML Schema testing
Additional constraints may exist that are not possible to specify in an XML schema. For example, the IMS CP specification requires that every package contains a file called imsmanifest.xml. Such constraints may be derived from the base specification documentation or from an Application Profile. The additional constraints outlined in the base specification documentation may be applicable to all implementations of a specification. These types of tests must be interpreted from the specification and are added directly to the test system.

In order to express additional constraints in the Application Profile (which are not derived from the base specification documentation), the Telcert tools use the Schematron mark-up language (Lee & Chu, 2000). Schematron is an assertion-based language which utilises pattern-matching techniques. Using Schematon allows conditional processing to be added to Application Profiles in a machine readable manner.

Overview of the Telcert conformance testing process

This section contains an overview and demonstration of the Telcert project tools for conformance testing e-learning specifications. The end-goal of this process is to produce and successfully test content which can be used to test conformance to a localised implementation of an e-learning specification (Figure 1). The base specification is combined with the Application Profile to produce a set of test content which can then be seeded with deliberate errors. This test suite is then used to test against the product (represented by the black box). In this case, the implementation should also be fully conformant with the base specification. Using these tools and processes, e-learning communities have the ability to test content and develop comprehensive test suites to test software applications.

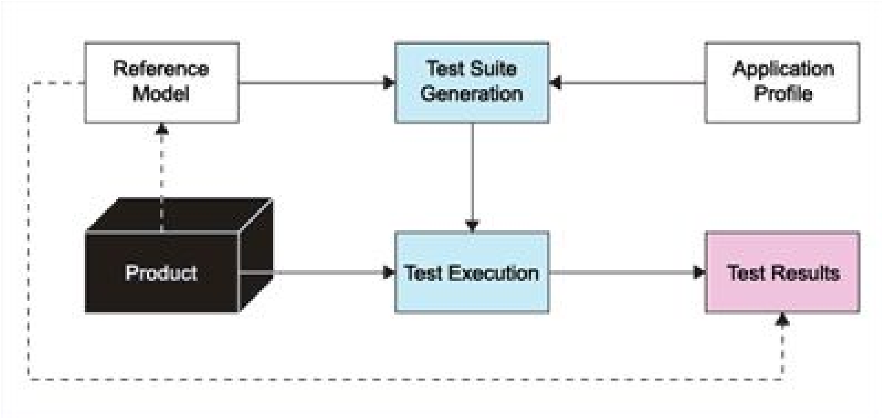

Figure 1. The general conformance testing process

The Telcert tools represent an Application Profile as an XML document detailing the modifications made against the base specification. From these modifications, a new XML schema can be derived which incorporates these modifications. The tools support the following tasks:

- Development of an Application Profile from a base specification
- Generation of a localised schema based on the Application Profile
- Creation of new content/modification of existing document instances based on localised schemas
- Testing derived test content for conformance

Development of an Application Profile from a base specification: Schemaprof

The Schemaprof tool (Schemaprof, 2005) allows the user to load a base schema and specify legal modifications and extensions without editing the schema directly. This tool allows users with a conceptual knowledge of the specification to make modifications without needing the technical knowledge to modify the schema directly. Furthermore, it reduces the incidence of human-error by limiting the possibility to create incorrect or illegal modifications. The tool also allows the addition of modifications that go beyond what can be specified in an XML schema by utilising the Schematron language. Once modifications have been made, the Schemaprof tool produces an XML file which encapsulates these modifications.

Generation of localised schema based on the Application Profile: Schema Transformation Tool

This tool works as a plug-in for Schemaprof, which enables the creation of a localised XML schema by incorporating the derived modifications and the original base specification schema. This derived schema can then form a template for the creation of content based on the application profile.

Creating new content/modifying existing document instances based on localised schemas: Content Reengineering Tool

The Content Reengineering Tool allows the creation of content based on XML schemas. The tool also allows content based on one schema to be transformed to another schema. This tool is based on the Reload editor (RELOAD, 2005) with features added to aid the creation and modification of content based on localised schema. The tool supports content based on localised schemas and also schematron rules. Functionality has also been added to support base specification extensions and to transform content from one version type to another (such as from Sharable Content Object Reference Model

(SCORM) 1.2 to SCORM 2004). The Conformance testing principals outlined previously can be used in conjunction with this tool to create test content.

Testing derived test content for conformance: Telcert Test System

In the test system, content is tested against a particular e-learning specification. For each specification, a number of tests may be derived directly from the base specification documentation. For example, the IMS CP specification contains a number of rules determining the way the package is to be compressed into an archive format (commonly a .zip file). If these tests are mandatory for any implementation, then they are added to the test system directly. Content can therefore be tested against the base specification (as represented by the specification XSD file and the specification documentation), the supplied application profile as produced by the Schemaprof tool, and the generic tests contained in the test system. The test system allows the creation of test sessions which link the test content with the required tests. Test sessions can then be run, providing detailed information on the results of each test.

Case Study: Profiling for IMS Learning Design

The test system described in the previous section was developed to work with XML Schema-based specifications in a generic way, and was tested with content based on a number of e-learning specifications including IMS Learner Information Package (IMS LIP), IMS CP and IMS Learning Resource Metadata (IMS MD). This particular case study focuses on profiling work based around IMS Learning Design (IMS LD). IMS LD is meta-language which models learning designs and supports pedagogical diversity (Koper & Olivier, 2004; Koper, 2005). Units of learning modelled in IMS LD allow for reuse of course components, in addition to learning resources. The IMS LD specification is divided into three levels; A, B and C. The core of the specification is contained in Level A, with levels B and C both integrating and extending the previous level. The profile used for this case study was based on IMS Learning Design Level A.

The IMS LD specification was conceived within the context of the broad range of specifications developed by IMS. In this case study, two specifications were chosen to be profiled for use with IMS LD. The first, IMS CP, is a specification for packaging content into an archive is the recommended method for packaging an IMS LD and its associated files. An IMS Learning Design packaged into an IMS Content Package is called a Unit of Learning. Secondly, IMS MD was selected, which is an e-learning metadata specification used here to record information about the content of a Unit of Learning. An existing metadata Application Profile (Digitale Universiteit, 2004) was used as the basis for the modifications made to the IMS MD base specification. For IMS CP, the IMS LD Best Practice and Implementation Guide specifies a number of modifications required to encapsulate IMS LD (IMS Global Learning Consortium, 2004). The IMS LD specification itself was not modified during this profiling work. The Schemaprof tool was used to create Application Profiles for IMS CP and IMS MD as described previously. Localised XML schemas were then produced with the Schema Transformation Tool plug-in for Schemaprof. In this case study, all tests were able to be expressed in XML Schema or were added to the test system as generic specification tests. As such, Schematron expressions were not required.

The test content was based on four original Units of Learning, which were firstly modified with the Content Reengineering Tool to conform exactly to the localised schemas. Following category partition principals, modified elements were focused on and known errors were manually added to the test content to check that such errors would be correctly recognised. Boundary value analysis was used to create test content that would pass and fail boundaries such as minimum and maximum allowed. For each identified test or modification made to the base specification, at least one correct and one incorrect instance of test content was created. For example, it is a requirement for every content package to contain exactly one imsmanifest.xml file.Then test content was produced with zero instances (incorrect), and one instance (correct). Identified tests fell into two groups; package tests related to the IMS CP specification, and XML file tests which tested conformance to the Application Profile schemas.

Package tests

For generic IMS Content Packages containing IMS LD, seven generic tests were identified from the documentation and added to the test system, independent of any further modifications that might be made to the base specifications. These tests check the resources and information required for processing a Unit of Learning are contained within the archive. Tests include checking that the package can be decompressed and that all resources are referenced correctly within the file. The full list of tests relevant to a Unit of Learning include:

Table 1: Package Tests

Number	Name	Description
1	Decompression	The test system is able to decompress the package file
2	Manifest presence	Exactly one imsmanifest.xml file exists in the root level of the package and is named in all lowercase letters
3	Manifest schema check	All dependent schemas are located at the root level of the decompressed package
4	Manifest validation	Check all XML tests are met – refer table 2
5	Manifest resource presence	All referenced resources can be located
6	Manifest completeness	No files are present in the package that are not listed in the manifest
7	IdentifierRef to Identifier	All *indentifierRef* attributes must have a corresponding *identifier* attribute

XML File tests

Four tests were identified to test the XML files themselves for conformance with the Application Profiles:

Table 2: XML File Tests

Number	Name	Description
1	Well formed	XML is well formed
2	Internal validation	XML validates without error against the DTD or schema
3	Non-conditional restraints	XML document successfully validates against all non-conditional rules expressed in the application profile
4	Conditional restraints	XML document successfully validations against all conditional rules (expressed using Schematron)

As no conditional restraints were used in this particular case, the fourth test was not applicable to these Application Profiles. The XML schema tests can be broken down further to test various aspects of the test content against the schema. Conformance tests include boundary testing for modified minimum and maximum values, elements made mandatory which were optional in the base specification and modifications to simple types and elements which were optional in the base specification, but are forbidden in the application profile. Again, for each modification made, a minimum of one piece of test content was created which should fail the test, and one piece that should pass.

Test Cases and Results

A total of 46 individual pieces of test content were run through the test system. Half of this content was seeded with recorded errors. Running this test content through the test system tests the content, but this case also tests the effectiveness of the test system in passing correct content and identifying and failing incorrect content.

Package tests

Fourteen pieces of content were run against the package tests (Table 3). In certain cases where content could fail in more than one way, additional test cases were required. For example, in the "Manifest Presence" test, the package must contain a file named "imsmanifest.xml" in all lowercase. In this case created erroneous content is used to test a case where the file was missing (zero instances), where two "imsmanifest.xml" files existed, and also where the file was incorrectly named.

Table 3: Package Test Content

Number	Name	Number of pieces of test content
1	Decompression	2
2	Manifest presence	6
3	Manifest schema check	2
4	Manifest validation	2
5	Manifest resource presence	2
6	Manifest completeness	2
7	IdentifierRef to Identifier	2

XML File tests

Of the three XML File tests applicable to this profiling work (Table 4), the "Non-conditional restraints" test, deserves special mention. This test checks that the content conforms to the Application Profile. Therefore, for each unique modification made to the base specification at least two pieces of content (one correct and one incorrect) were identified.

Table 4: XML File Test Content

Number	Name	Number of pieces of test content
1	Well formed	3
2	Internal validation	2
3	Non-conditional restraints	23

Test summary

The central question posed by this case-study was whether the tools and processes used were sufficient for facilitating the development of localised e-learning specifications, and whether the test system was able to effectively test for conformance. For the profiling work outlined in this article, the test system correctly identified all content containing errors and passed all content that was known to be error-free. A tool for generating fragments of test content is currently (September 2005) under development within the Telcert project which would further automate the test content creation process.

Discussion

Combined with existing conformance testing techniques, the tools and processes used in this article offer an effective way of conformance testing XML schema-based e-learning specifications. Further detailed testing based on other specifications is certainly required to ensure the system meets the needs of the widest range of e-learning specifications possible. At the time of writing, the test content used in the system has been limited to the IMS specifications of LD, MD, CP, and EDS (European Diploma Supplement - a profile of IMS LIP) with work underway to also include IMS Question & Test Interoperability (IMS QTI). A simple trial using Sharable Content Object Reference Model (SCORM) content has also been conducted, although with an existing SCORM conformance test suite already available, this has not been a major focus. Two further outcomes of this work require further mention.

Firstly, a drawback of the Telcert tools as they are described in this article is that they do not facilitate "domain profiling". A domain profile is the localisation of a *set* of specifications for use within a particular community or domain. In other words, a domain profile is a *set* of application profiles. In such cases, modifications may span multiple specifications. The test system does not currently (September 2005) have the functionality to test requirements spanning multiple specifications. The example profiled in this paper may be considered a domain profile, however there were no interdependencies profiled across the specifications in this case. However, if this example was extended to include IMS QTI, this would be a case where two of the specifications had an interdependency, as properties can be shared between IMS LD and QTI. The tools and test system are currently being modified to allow such domain profiling tests and further testing is planned.

Secondly, generic conformance rules contained in written documentation (as opposed to the specification XSD representation) must be manually identified and entered into the test system for each new specification that is required to be tested. This highlights the necessity for unambiguous documentation. Clearly it would be better to be able to specify all requirements in a machine-readable, unambiguous way. The limitations of XML Schema (Gil & Ratnakar, 2002) have led to the development of a number of alternatives to replace or supplement an XSD file. UML offers the possibility for representing e-learning specifications in an implementation-neutral manner, however this is not yet common practice.

The next step from here is to move towards comprehensive interoperability testing, which would involve runtime testing and requires testing against expected behaviour. The project partners are currently investigating the possibilities for representing base specifications and application profiles in UML and deriving conformance tests directly from the UML representations, particularly for web services. This work sets the foundations for interoperability testing because runtime behaviours could be encapsulated in such UML representations.

Conclusion

A major motivating force behind the adoption of e-learning specifications is the adoption of standardised content and systems and the eventual achievement of true interoperability. While interoperability between e-learning systems and content may be an ultimate aim, this goal cannot be reached without firstly achieving conformance to base specifications. The creation of Application Profiles is an issue that has made e-learning specification conformance testing in particular more complex because of the flexibility it allows to implementations. In response to the difficulties faced by e-learning communities with conformance testing specifications, the Telcert project tools simplify such conformance testing while preserving the flexibility of being able to localise implementations. This article combined these tools with established conformance testing techniques to provide a complete solution for testing e-learning specifications.

The results from the case study with IMS Learning Design content demonstrated the effectiveness of this approach, for localising e-learning specifications and testing content for conformance. We are currently carrying out tests with a wider range of specifications and exploring the potential for using UML to derive tests directly from the specification documentation. Wider Adoption of such conformance testing solutions in e-learning communities is necessary to ensure the development of conformant e-learning systems.

References

Binder, R. V. (1999). *Testing Object-Oriented Systems: Models, Patterns, and Tools*, Addison Wesley, Object Technology Series.

Digitale Universiteit (2004). *Werken met metadata in DU-projecten*, retrieved March 20, 2005 from http://www.du.nl/.

Gil, Y., & Ratnakar, V. (2002). A Comparison of (Semantic) Markup Languages. In Haller, S. M & Simmons, G. (Eds.) *Proceedings of the Fifteenth international Florida Artificial intelligence Research Society Conference*, Florida: AAAI Press, 413-418.

Hao, R., Lee, D., Sinha, R. & Griffeth, N. (2004). Integrated System Interoperability Testing With Applications to VoIP. IEEE/ACM Transactions on Networking, 12 (5), 823-836.

Harder, M., Mellen, J., & Ernst, M. D. 2003. Improving test suites via operational abstraction. In *Proceedings of the 25th international Conference on Software Engineering.* IEEE Computer Society, Washington, DC, 60-71.

IMS Global Learning Consortium (2003). *IMS Abstract Framework: Glossary*, retrieved April 22, 2005 from http://www.imsglobal.org/af/afv1p0/imsafglossaryv1p0.html.

IMS Global Learning Consortium (2003). *IMS Learning Design Best Practice and Implementation Guide*, retrieved August 15, 2005 from http://www.imsglobal.org/learningdesign/ldv1p0/imsld_bestv1p0.html.

IMS Global Learning Consortium (2004). *IMS Content Packaging Best Practice and Implementation Guide*, retrieved August 14, 2005 from http://www.imsglobal.org/content/packaging/cpv1p1p4/imscp_bestv1p1p4.html.

Kindrick, J., Sauter J. & Matthews R. (1996). Improving Conformance and Interoperability Testing. StandardView, 4 (1), 61-68.

Koper, R. (2005). An Introduction to Learning Design. In R. Koper & C. Tattersall (Eds.). *Learning Design: A handbook on modeling and delivering networked education and training*. Heidelberg: Springer-Verlaag, 3-20.

Koper, R. & Olivier, B. (2004). Representing the Learning Design of Units of Learning. Educational Technology & Society, 7 (3), 97-111.

Lee, D. & Chu, W. W. 2000. Comparative analysis of six XML schema languages. SIGMOD Rec. 29, 3 (Sep. 2000), 76-87.

Malek, M. & Dibuz , S. (1998). Pragmatic Method for Interoperability Test Suite Derivation, Euromicro, vol. 02, no. 2, 20838.

Ostrand, T. J. & Balcer, M. J. (1988). The Category-Partition Method for Specifying and Generating Functional Tests. Communications of the ACM, 31 (6), 676-686.

Ramachandran, M. (2003). Testing Software Components Using Boundary Value Analysis. In *29th Euromicro Conference Proceedings*. 94-98. Washington DC: IEEE Computer Society.

RELOAD (2005). *Reusable eLEarning Object Authoring & Delivery project website*, retrieved 18 August 2005 from http://www.reload.ac.uk/.

Schemaprof (2005). Schemaprof website, retrieved 15 May 2005 from http://herakles.uni-koblenz.de/schemaprof/.

TELCERT (2005). Technology Enhanced Learning Conformance: European Requirements and Testing project website, retrieved 28 July 2005 from http://www.opengroup.org/telcert/.

Xiang, X. & Shi Y., Guo, L. (2003). A Conformance Test Suite of Localized LOM Model. In *Proceedings of the 3rd IEEE International Conference on Advanced Learning Technologies*. 288-289. Washington DC: IEEE Computer Society.

Jung, Y. & Lee, J. (2003). Experiences with Generation of Conformance Test Suite for Q.2971 Network-side Testing. In *Proceedings of the 13th international Conference on information Networking*, 286-290. Washington DC: IEEE Computer Society.