

Australian Standard™

**Information technology—Multimedia
framework (MPEG-21)**

Part 8: Reference Software



This Australian Standard was prepared by Committee IT-029, Coded Representation of Picture, Audio and Multimedia/Hypermedia Information. It was approved on behalf of the Council of Standards Australia on 10 April 2006.
This Standard was published on 28 April 2006.

The following are represented on Committee IT-029:

Australian Broadcasting Authority
Australian Broadcasting Corporation
Australian Consumers' Association
Australian Communications and Media Authority
Australian Subscription Television and Radio Association
CSIRO Information and Communication Technologies Centre
Department of Defence (Australia)
Free TV Australia
Special Broadcasting Service
The University of New South Wales
University of Wollongong
Victoria University of Technology

Keeping Standards up-to-date

Standards are living documents which reflect progress in science, technology and systems. To maintain their currency, all Standards are periodically reviewed, and new editions are published. Between editions, amendments may be issued. Standards may also be withdrawn. It is important that readers assure themselves they are using a current Standard, which should include any amendments which may have been published since the Standard was purchased.

Detailed information about Standards can be found by visiting the Standards Web Shop at www.standards.com.au and looking up the relevant Standard in the on-line catalogue.

Alternatively, the printed Catalogue provides information current at 1 January each year, and the monthly magazine, *The Global Standard*, has a full listing of revisions and amendments published each month.

Australian Standards™ and other products and services developed by Standards Australia are published and distributed under contract by SAI Global, which operates the Standards Web Shop.

We also welcome suggestions for improvement in our Standards, and especially encourage readers to notify us immediately of any apparent inaccuracies or ambiguities. Contact us via email at mail@standards.org.au, or write to the Chief Executive, Standards Australia, GPO Box 476, Sydney, NSW 2001.

This Standard was issued in draft form for comment as DR 05337.

Australian Standard™

Information technology—Multimedia framework (MPEG-21)

Part 8: Reference Software

First published as AS ISO/IEC 21000.8—2006.

COPYRIGHT

© Standards Australia

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher.

Published by Standards Australia GPO Box 476, Sydney, NSW 2001, Australia

ISBN 0 7337 7391 5

PREFACE

This Standard was prepared by the Standards Australia Committee IT-029, Coded Representation of Picture, Audio and Multimedia/Hypermedia Information.

This Standard is identical with, and has been reproduced from ISO/IEC 21000-8:2006, *Information technology—Multimedia framework (MPEG-21)—Part 8: Reference Software*.

The objective of this Standard is to provide the Australian multimedia design industry with reference software for implementing and better utilizing the normative clauses of the other parts of ISO/IEC 21000.

The term ‘informative’ has been used in this Standard to define the application of the annex to which it applies. An ‘informative’ annex is only for information and guidance.

As this Standard is reproduced from an international standard, the following applies:

- (a) Its number appears on the cover and title page while the international standard number appears only on the cover.
- (b) In the source text ‘this part of ISO/IEC 21000’ should read ‘this Australian Standard’.
- (c) A full point substitutes for a comma when referring to a decimal marker.

References to International Standards should be replaced by references to Australian or Australian/New Zealand Standards, as follows:

<i>Reference to International Standard</i>		<i>Australian Standard</i>	
ISO/IEC		AS ISO/IEC	
15938	Information technology—Multimedia content description interface	15938	Information technology—Multimedia content description interface
15938-5	Part 5: Multimedia description schemes	15938.5	Part 5: Multimedia description schemes
21000	Information technology—Multimedia framework (MPEG-21) (All Parts)	21000	Information technology—Multimedia framework (MPEG-21) ¹

¹ Standards Australia is in the process of adopting most of ISO/IEC 21000 series Standards. Refer to the website for the current list.

CONTENTS

	<i>Page</i>
1 Scope	1
2 Normative references	1
3 Terms, definitions, symbols and abbreviated terms	1
3.1 Terms and definitions	1
3.2 Symbols and abbreviated terms	2
4 Overview and conventions	2
4.1 Organization of the document	2
4.2 Overview of ISO/IEC 21000 reference software	3
4.3 Overview of ISO/IEC 21000 utility software	3
4.4 Overview of integrated utility software across the individual ISO/IEC 21000 parts	4
5 Reference software for the ISO/IEC 21000 parts	4
5.1 Introduction	4
5.2 ISO/IEC 21000-2	5
5.3 ISO/IEC 21000-3	6
5.4 ISO/IEC 21000-4	7
5.5 ISO/IEC 21000-5	7
5.6 ISO/IEC 21000-6	17
5.7 ISO/IEC 21000-7	18
Annex A (informative) Utility software for the ISO/IEC 21000 parts	21
Annex B (informative) Integrated utility software across the individual ISO/IEC 21000 parts	53

INTRODUCTION

The multimedia industry is increasing at a rapid pace. For this industry, the term “content” is widely used across different segments and applied in many different ways. For this reason the term is deliberately avoided within the context of ISO/IEC 21000 International Standards, where it has been replaced by the defined terms “Digital Item”, “media resource” and “resource”. Of equal importance for the specifications of the multimedia framework is the notation of the User. A User of a system includes all members of the value chain (i.e. creator, rights holders, distributors and consumers of Digital Items).

This eighth part of ISO/IEC 21000 describes the reference software. The reference software of ISO/IEC 21000 serves three main purposes:

- validation of the written specification of the several parts of ISO/IEC 21000;
- clarification of the written specification of the several parts of ISO/IEC 21000; and
- conformance testing for checking interoperability for the various applications against the reference software which aims to be compliant with ISO/IEC 21000.

AUSTRALIAN STANDARD

Information technology — Multimedia framework (MPEG-21) —**Part 8:
Reference Software****1 Scope**

This part of ISO/IEC 21000 describes reference software implementing the normative clauses of the other parts of ISO/IEC 21000. The information provided is applicable for determining the reference software modules available for parts of ISO/IEC 21000, understanding the functionality of the available reference software modules, and utilizing the available reference software modules.

In addition to the reference software, available utility software that utilizes the reference software is also described. This utility software can assist in understanding how to utilize the reference software, as well as providing further insight into the applicable parts of ISO/IEC 21000, e.g. informative clauses of the other parts of ISO/IEC 21000.

2 Normative references

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

ISO/IEC 15938-5, *Information technology – Multimedia content description interface – Part 5: Multimedia description schemes*

ISO/IEC 21000 (all parts), *Information technology – Multimedia framework (MPEG-21)*

3 Terms, definitions, symbols and abbreviated terms**3.1 Terms and definitions****3.1.1****module**

software component implementing **reference software** or **utility software**

3.1.2**reference software**

one or more **modules** utilizing normative parts of ISO/IEC 21000 International Standards

3.1.3**utility software**

one or more **modules** utilizing informative parts of ISO/IEC 21000 International Standards and/or the usage of **reference software** within real-world applications

3.2 Symbols and abbreviated terms

For the purposes of this document, the symbols and abbreviated terms given in the following apply:

API	Application Programming Interface
DI	Digital Item
DIA	Digital Item Adaptation
DID	Digital Item Declaration
DII	Digital Item Identifier
IS	International Standard
FDIS	Final Draft International Standard
MPEG	Moving Pictures Expert Group
RDD	Rights Data Dictionary
REL	Rights Expression Language

4 Overview and conventions

4.1 Organization of the document

In the remainder of this part of ISO/IEC 21000, each reference and utility software module is described following the convention as below:

Module Name: <i>Name of the ZIP file with the following structure: /<directory>/<part>-<module_name>-<implementation>-<version>.zip</i>	
<i><directory>: directory name in which the module can be found, i.e., 21000-2_DID, 21000-3_DII, 21000-5_REL, 21000-6_RDD, 21000-7_DIA.</i>	
<i><part>: abbreviation of the ISO/IEC 21000 part, i.e., DID, DII, REL, RDD, DIA.</i>	
<i><module_name>: name of the module, e.g., Parser, Validator, etc.</i>	
<i><implementation>: letter A, B, C, etc. for different implementations.</i>	
<i><version>: version number, i.e., n_n_n n_n n</i>	
Description	
<i>Describes the functionality the module provides.</i>	
Input	
<i>Describes the input of the module.</i>	
Output	
<i>Describes the output of the module.</i>	
Programming Language(s)	

	<i>Lists the programming language(s) in which the module is written.</i>
Platform(s)	
	<i>Lists the platforms the module has been tested on and is supposed to run on.</i>
Dependencies	
	<i>Lists the required libraries and code with version information.</i>
Details	
	<i>Lists any implementation details, such as architecture diagrams and data flows.</i>

4.2 Overview of ISO/IEC 21000 reference software

This part of ISO/IEC 21000 comprises reference software modules for the following ISO/IEC 21000 parts that have advanced to FDIS or IS status respectively:

- ISO/IEC 21000-2, Information technology — Multimedia framework (MPEG-21) — Part 2: Digital Item Declaration (DID): The corresponding reference software module provides means for parsing DIDs into an internal data structure and checking the validation rules as specified within ISO/IEC 21000-2. The reference software for ISO/IEC 21000-2 is described in subclause 5.2.
- ISO/IEC 21000-3, Information technology — Multimedia framework (MPEG-21) — Part 3: Digital Item Identification (DII): The corresponding reference software module provides means for parsing DII information into an internal data structure. The reference software for ISO/IEC 21000-3 is described in subclause 5.3.
- ISO/IEC 21000-5, Information technology — Multimedia framework (MPEG-21) — Part 5: Rights Expression Language (REL): The corresponding reference software modules provide means for checking the validity of a REL license as well as the validity of an authorization request according to ISO/IEC 21000-5. Furthermore, they allow for checking if an authorization story is an authorization proof for an authorization request. The reference software for ISO/IEC 21000-5 is described in subclause 5.5.
- ISO/IEC 21000-6, Information technology — Multimedia framework (MPEG-21) — Part 6: Rights Data Dictionary (RDD): The corresponding reference software module provides means for querying the RDD database for a given input RDD term. It provides the genealogy and the IsTypeOf hierarchy of the term. The reference software for ISO/IEC 21000-6 is described in subclause 5.6.
- ISO/IEC 21000-7, Information technology — Multimedia framework (MPEG-21) — Part 7: Digital Item Adaptation (DIA): The corresponding reference software module provides means for parsing and serializing DIA descriptions as well as retrieving and modifying information within DIA descriptions. Additionally, the reference software modules for ISO/IEC 21000-7 implement the normative behavior of the processes as specified within ISO/IEC 21000-7. The reference software for ISO/IEC 21000-7 is described in subclause 5.7.

4.3 Overview of ISO/IEC 21000 utility software

This part of ISO/IEC 21000 comprises utility software modules for the following ISO/IEC 21000 parts that have advanced to FDIS or IS status respectively:

- ISO/IEC 21000-2, Information technology — Multimedia framework (MPEG-21) — Part 2: Digital Item Declaration (DID): The corresponding utility software module provides means for invoking the reference software module for ISO/IEC 21000-2 and presenting results to the user. The utility software for ISO/IEC 21000-2 is described in annex A.2.

- ISO/IEC 21000-3, Information technology — Multimedia framework (MPEG-21) — Part 3: Digital Item Identification (DII): The corresponding utility software module provides means for invoking the reference software module for ISO/IEC 21000-3 and presenting results to the user. The utility software for ISO/IEC 21000-3 is described in annex A.3.
- ISO/IEC 21000-5, Information technology — Multimedia framework (MPEG-21) — Part 5: Rights Expression Language (REL): The corresponding utility software modules provide means for interpreting and creating of licenses as specified within ISO/IEC 21000-5. The utility software for ISO/IEC 21000-5 is described in annex A.5.
- ISO/IEC 21000-6, Information technology — Multimedia framework (MPEG-21) — Part 6: Rights Data Dictionary (RDD): The corresponding utility software modules provide means for browsing terms specified within ISO/IEC 21000-6 and a multilingual registry for the ISO/IEC 21000-6 in five different languages. The utility software for ISO/IEC 21000-6 is described in annex A.6.
- ISO/IEC 21000-7, Information technology — Multimedia framework (MPEG-21) — Part 7: Digital Item Adaptation (DIA): The corresponding utility software modules provide means for demonstrating informative clauses and annexes as described in ISO/IEC 21000-7 such as resource and description adaptation engines. The utility software for ISO/IEC 21000-7 is described in annex A.7.

4.4 Overview of integrated utility software across the individual ISO/IEC 21000 parts

This part of ISO/IEC 21000 comprises integrated utility software across the individual ISO/IEC 21000 parts that have advanced to FDIS or IS status respectively:

- One integrated utility software module utilizing ISO/IEC 21000-5 and ISO/IEC 21000-6 reference and utility software modules is described in annex B.2.
- One integrated utility software module utilizing ISO/IEC 21000-2 and ISO/IEC 21000-5 reference and utility software modules is described in annex B.3.
- One integrated utility software module utilizing ISO/IEC 21000-2, ISO/IEC 21000-5, and ISO/IEC 21000-6 reference and utility software modules is described in annex B.4.
- One integrated utility software module utilizing ISO/IEC 21000-2, ISO/IEC 21000-5, and ISO/IEC 21000-7 reference and utility software modules is described in annex B.5.

5 Reference software for the ISO/IEC 21000 parts

5.1 Introduction

This clause describes the ISO/IEC 21000 reference software for the parts as listed in subclause 4.2. The ISO/IEC 21000 reference software is written in Java and follows the following package structure:

<i>Name</i>	<i>Description</i>
org	Java package name for reference software provided by organizations such as ISO/IEC, W3C, or similar.
org.iso	Java package name for reference software provided by ISO/IEC.
org.iso.mpeg	Java package name for reference software provided by ISO/IEC JTC 1/SC 29/WG 11.
org.iso.mpeg.mpeg21	Java package name for reference software provided in the course of the development of ISO/IEC 21000.

Name	Description
	Note: Subsequent packages for the individual ISO/IEC 21000 parts use the uncapitalized abbreviations as defined in subclause 3.2, e.g.,
	— <code>org.iso.mpeg.mpeg21.did</code> for ISO/IEC 21000-2 or
	— <code>org.iso.mpeg.mpeg21.dia</code> for ISO/IEC 21000-7.

5.2 ISO/IEC 21000-2

5.2.1 Introduction

This subclause describes the ISO/IEC 21000 reference software for part 2 of ISO/IEC 21000. In order to allow building applications on top of the ISO/IEC 21000-2 reference software an API is provided. In particular, the API exercises or represents all normative features, specified in the ISO/IEC 21000-2 specification.

5.2.2 DID Parser

Module Name: /21000-2 DID/DID-Parser-A-1 0 0.zip	
Description	
	The DID parser checks whether an incoming DID document is valid against the ISO/IEC 21000 DID schema; it also checks whether all validation rules that are defined within the ISO/IEC 21000 DID specification are fulfilled.
Input	
	A DIDL document.
Output	
	Valid, not valid, reasons why, according to the XML schema and the additional validation rules of 21000-2.
Programming Language(s)	
	Java
Platform(s)	
	Any platform supported by the Java J2SE and Xerces XML parser.
Dependencies	
	— Java version "1.4.1_02" or later, which can be downloaded from http://java.sun.com/j2se/1.4.2/. — Xerces2 Java Parser 2.6.2 or later, which can be downloaded from http://xml.apache.org/. — Log4j version 1.2.8 or later, which can be downloaded from http://logging.apache.org/log4j/docs/.

Details

5.3 ISO/IEC 21000-3

5.3.1 Introduction

This subclause describes the ISO/IEC 21000 reference software for part 3 of ISO/IEC 21000. In order to allow building applications on top of the ISO/IEC 21000-3 reference software an API is provided. In particular, the API exercises or represents all normative features, specified in the ISO/IEC 21000-3 specification.

5.3.2 DII Parser

Module Name: [/21000-3 DII/DII-Parser-A-1 0 0.zip](#)

Description

The DII reference software demonstrates how DII information can be included in DIDs and how this information can be used to extract relevant DID information based on a DII Identifier, DII RelatedIdentifier or DII Type.

Input

A DIDL document with DII information.

Output

Java objects in memory providing access to the information in the input DII description

Programming Language(s)

Java

Platform(s)

Any platform supported by the Java J2SE, Xerces, and kXML parser.

Dependencies

- Java version "1.4.1_02" or later, which can be downloaded from <http://java.sun.com/j2se/1.4.2/>.
- Xerces2 Java Parser 2.6.2 or later, which can be downloaded from <http://xml.apache.org/>.
- kXML 2 release 2.1.9 or later, which can be downloaded from <http://www.kxml.org/>.
- Log4j version 1.2.8 or later, which can be downloaded from <http://logging.apache.org/log4j/docs/>.

Details

5.4 ISO/IEC 21000-4

This subclause describes the ISO/IEC 21000 reference software for part 4 of ISO/IEC 21000. Currently, part 4 of ISO/IEC 21000 is not at FDIS or IS level and therefore not included in this version of ISO/IEC 21000 reference software but may be added in future amendments.

5.5 ISO/IEC 21000-5

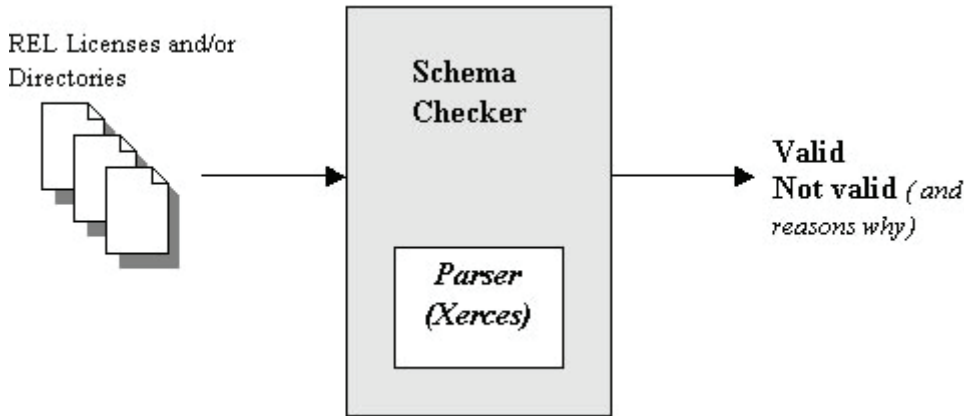
5.5.1 Introduction

This subclause describes the ISO/IEC 21000 reference software for part 5 of ISO/IEC 21000.

5.5.2 REL schema checker

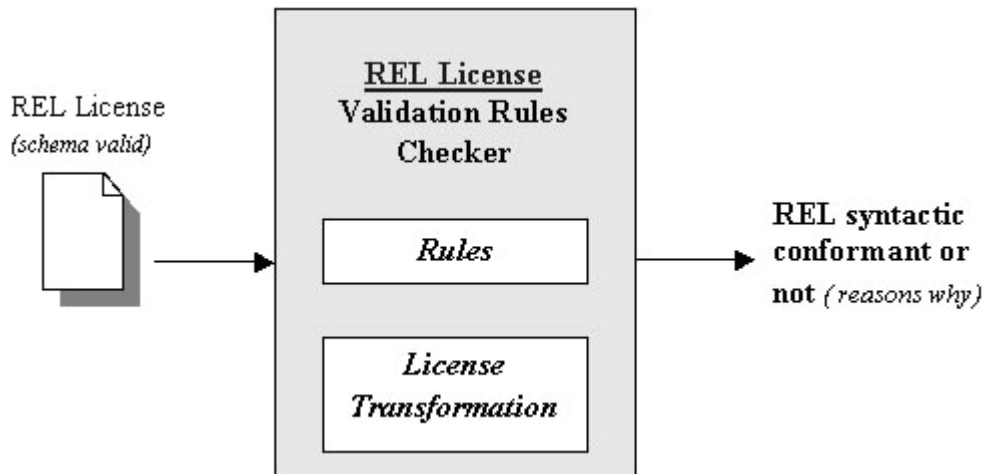
Module Name: /21000-5 REL/REL-SchemaChecker-A-1 0 0.zip	
Description	
	The schema checker module validates a set of REL Licenses against the XML Schemas specified in the Licenses. It validates syntactically the REL Licenses files and/or the REL Licenses contained in the directory specified by the user.
Input	
	A list of licenses and/or directories that contain licenses.
Output	
	Valid, not valid, reasons why, according to the XML schemas specified within the licenses.
Programming Language(s)	
	C++
Platform(s)	
	Any platform supported by Microsoft .Net framework (e.g., Windows 98, 2000, NT, ME, XP).
Dependencies	
	Microsoft .Net Framework, which can be obtained at: http://msdn.microsoft.com/netframework/downloads/howtoget.asp
Details	

Module Name: /21000-5 REL/REL-SchemaChecker-B-1 1 0.zip	
Description	
	The schema checker module validates a set of REL Licenses against the XML Schemas specified in the Licenses. It validates syntactically the REL Licenses files and/or the REL Licenses contained in the directory specified by the user.

Input	
	A list of licenses and/or directories that contain licenses.
Output	
	Valid, not-valid, and reasons why, according to the XML schemas specified within the licenses.
Programming Language(s)	
	Java
Platform(s)	
	Any platform supported by the Java J2SE and Xerces XML parser.
Dependencies	
	— The Xerces XML Java Parser 2.6.2 or later which can be downloaded at http://xml.apache.org. — The Java J2SE SDK 1.3.1 or later which can be downloaded at http://java.sun.com/j2se/.
Details	
Architecture diagram:  <pre> graph LR A[REL Licenses and/or Directories] --> B[Schema Checker] subgraph B [Schema Checker] C[Parser Xerces] end B --> D["Valid Not valid (and reasons why)"] </pre> Figure 1 — REL-SchemaChecker-B-1_1_0 architecture diagram	

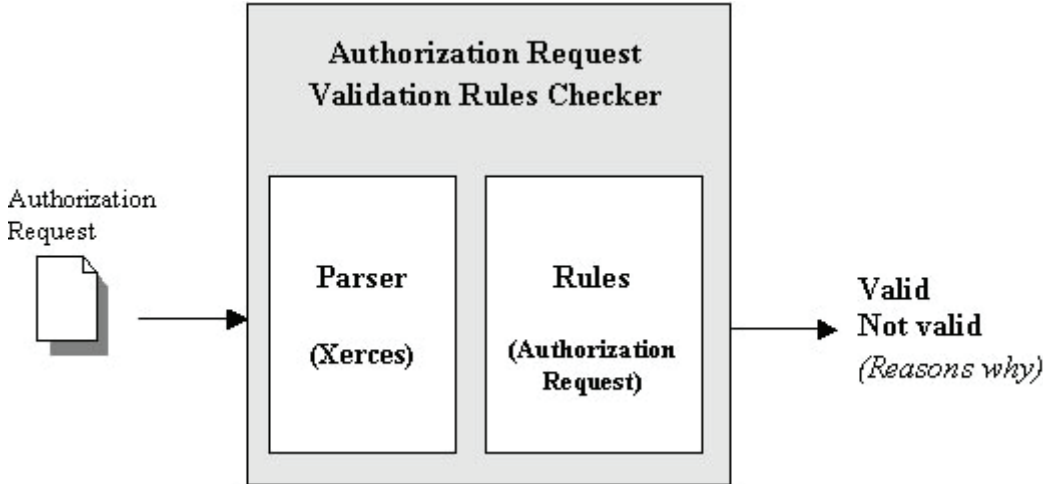
5.5.3 REL license validation rules checker

Module Name: /21000-5 REL/REL-ValidationRuleChecker-A-1 1 1.zip
Description
The validation rules checker module validates if a schema valid REL license is a valid REL License according to the MPEG-21 REL standard specification. A license will be REL syntactic conformant if it complies with all the rules specified in the REL standard specification.

Input	
	A schema valid REL license.
Output	
	REL syntactic conformant or not, reasons why.
Programming Language(s)	
	Java
Platform(s)	
	Any platform supported by the Java J2SE and Xerces XML parser.
Dependencies	
	— The Xerces XML Java Parser 2.6.2 or later which can be downloaded at http://xml.apache.org. — The Java J2SE SDK 1.3.1 or later which can be downloaded at http://java.sun.com/j2se/.
Details	
	Architecture diagram:  <pre> graph LR Input[REL License (schema valid)] --> Checker[REL License Validation Rules Checker] subgraph CheckerBox [REL License Validation Rules Checker] Rules[Rules] License[License Transformation] end Rules --> Output[REL syntactic conformant or not (reasons why)] </pre> Figure 2 — REL-ValidationRuleChecker-A-1_1_1 architecture diagram

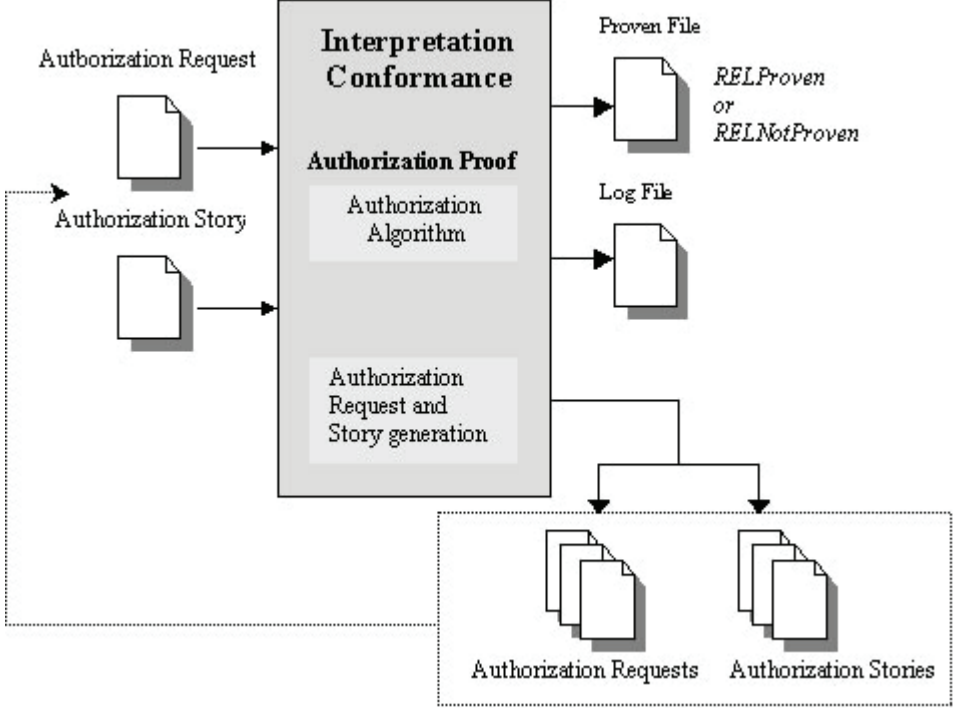
5.5.4 REL authorization request validation rules checker

Module Name: /21000-5 REL/REL-AuthorizationRequestValidationRulesChecker-1 0 0.zip
Description
The authorization request validation rules checker module checks if an authorization request is syntactically valid according to the MPEG-21 REL standard specification.

Input	
	An XML schema valid authorization request.
Output	
	REL validation rule conformant or not, reasons why.
Programming Language(s)	
	Java
Platform(s)	
	Any platform supported by the Java J2SE and Xerces XML parser.
Dependencies	
	— The Xerces XML Java Parser 2.6.2 or later which can be downloaded at http://xml.apache.org. — The Java J2SE SDK 1.3.1 or later, which can be downloaded at http://java.sun.com/j2se/.
Details	
	Architecture diagram:  Figure 3 — REL-AuthorizationRequestValidationRulesChecker-1_0_0 architecture diagram

5.5.5 REL interpretation

Module Name: /21000-5 REL/REL-Interpretation-A-1_0_0.zip	
Description	
The interpretation module checks if the Authorization Story is an Authorization Proof for the Authorization Request. This module implements the authorization proof algorithm in the MPEG-21 REL standard specification.	

Input	— Any valid authorization request. — Any valid authorization story.
Output	Authorization story IS/IS NOT authorization proof for the authorization request.
Programming Language(s)	Java
Platform(s)	Any platform supported by the Java J2SE and Xerces XML parser.
Dependencies	— The Xerces XML Java Parser 2.6.2 or later which can be downloaded at http://xml.apache.org. — The Java J2SE SDK 1.3.1 or later, which can be downloaded at http://java.sun.com/j2se/.
Details	Architecture diagram:  Figure 4 — REL-InterpretationConformance-A-1_0_0 architecture diagram

Module Name: /21000-5 REL/REL-Interpretation-B-1 0 0.zip	
Description	
The REL interpretation module implements the MPEG-21 REL standard specification. It checks if the input Authorization Request is proven within the input authorization search space (described by the other four input parameters), and returns the Authorization Result.	
Input	
— an RELAuthorizationRequest, — a set of RELAuthorizationStories, — a set of RELAuthorizers, — a set of <code>r:Grant</code> and <code>r:GrantGroup</code> RELNodes, and — a set of <code>r:ToConstraint</code> RELNodes	
Output	
an RELAuthorizationResult	
Programming Language(s)	
C#	
Platform(s)	
Any platform that supports the .NET Framework.	
Dependencies	
— The .NET Framework version 1.1. The Microsoft version of it can be downloaded at http://www.microsoft.com/.	
Details	
1. Architecture diagram	

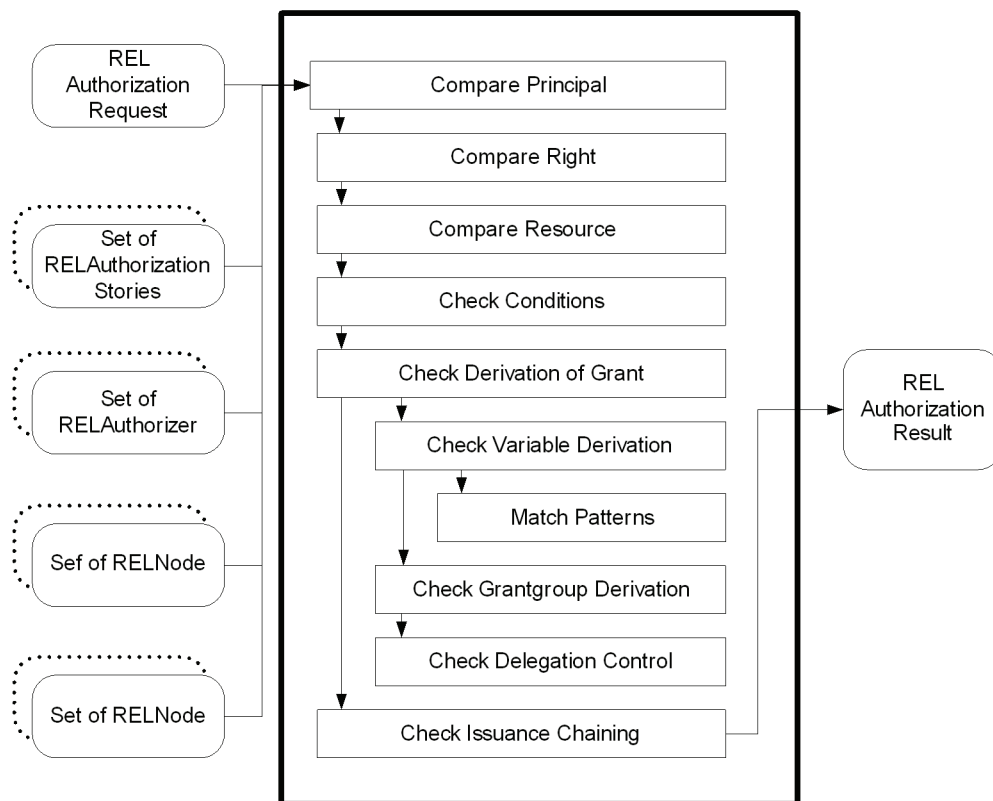


Figure 5 — REL Interpretation Reference Software Architecture Diagram

Function Performed	Corresponding API	Corresponding Source File
Compare Principal	Surpasses()	RELInterpreter.cs
Compare Right	IsEqual()	RELUtils.cs
Compare Resource	IsEqual()	RELUtils.cs
Check Conditions	IsSatisfied()	RELConditionManager.cs
Check Derivation of Grant	IsDerived()	RELInterpreter.cs
Check Variable Derivation	IsOneStepDerived6922()	RELInterpreter.cs
Match Patterns	NodeMatches()	RELInterpreter.cs
Check GrantGroup Derivation	IsOneStepDerived6923()	RELInterpreter.cs
Check Delegation Control	IsOneStepDerived6924()	RELInterpreter.cs
Check Issuance Chaining	IsAuthorizationProof()	RELInterpreter.cs

2. Data Types

The following points should be kept in mind when reading this section:

- Unless specified otherwise below, NULL values and encrypted things are not allowed.
- It is assumed that the constructors and modifiers for each data type will ensure that all of the "allowed values" constraints for each data type will be enforced.
- When it says "Any prefix:Type", it means any prefix:Type that is
 - Well formed,
 - Schema-valid, *and*
 - Valid according to any provisions in the specification applying to prefix:Type.

NOTE The reference software already exists in ISO/IEC 21000-8 to do much of this checking.

2.1 RELAuthorizationRequest (see 5.2 of ISO/IEC 21000-5)

Member	Data type	Allowed values
principal	RELNode	Any r:Principal or NULL Note: A NULL value here corresponds to the case where the Principal Member of the authorization request is absent as described in the REL.
right	RELNode	Any r:Right
resource	RELNode	Any r:Resource or NULL Note: A NULL value here corresponds to the case where the Resource Member of the authorization request is absent as described in the REL.
interval	RELInterval	Any
context	RELAuthorizationContext	Any
licenses	Set<RELNode>	Any r:License elements each with one r:issuer child
roots	Set<RELNode>	Any r:Grant elements

2.2 RELAuthorizationContext (see 5.3 of ISO/IEC 21000-5)

Member	Data type	Allowed values
properties	Map<RELPropertyName, Object>	Any

2.3 RELAuthorizationStory (see 5.4 of ISO/IEC 21000-5)

Member	Data type	Allowed values
primGrant	RELNode	Any primitive grant
grantOrGrantGroup	RELNode	Any r:Grant or r:GrantGroup
authE	RELAuthorizer	Any or NULL Note: A NULL value here corresponds to the case where the authorizer member of the authorization story is absent as described in the REL.

2.4 RELAuthorizer (see 5.5 of ISO/IEC 21000-5)

Member	Data type	Allowed values
license	RELNode	Any r:License with one r:issuer child
principal	RELNode	Any r:Principal
instant	RELDateTime	Any
context	RELAuthorizationContext	Any
story	RELAuthorizationStory	Any

2.5 RELAuthorizationResult

Member	Data type	Allowed values
result	enumeration	"RELProven" or "RELNotProven" Note: If the RELAuthorizationContext in the input does not contain all the needed properties, the RELAuthorizationResult will be RELNotProven. If there are other exceptions such as "out of memory," "can not read input file," etc., these should be handled as exceptions and shouldn't have any normative implications within the Reference Software part.