
**Information technology — MPEG
extensible middleware (MXM) —**

**Part 2:
MXM API**

Technologies de l'information — Intergiciel MPEG extensible (MXM) —

iTeh STANDARD PREVIEW
Partie 2: API MXM
(standards.iteh.ai)

[ISO/IEC 23006-2:2011](https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011)

<https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011>

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 23006-2:2011](https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011)

<https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011>

© ISO/IEC 2011

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword	v
Introduction.....	vi
1 Scope	1
2 Normative references	1
3 Terms and definitions	2
4 Abbreviated terms	2
5 Namespace conventions	3
6 Common MXM interfaces and classes	4
7 MXM Engine APIs	4
7.1 Digital Item Engine APIs	4
7.1.1 DIDEngine.....	4
7.1.2 DI Creation	5
7.1.3 DI Editing.....	5
7.1.4 DI Access.....	5
7.2 MPEG-21 File Engine APIs	5
7.2.1 MPEG21FileEngine.....	5
7.2.2 MPEG21 File Creation	5
7.2.3 MP21 File Access	5
7.3 REL Engine APIs	5
7.3.1 RELEngine	5
7.3.2 Rights Expression creation	6
7.3.3 Rights Expression access	6
7.3.4 Authorisation	6
7.4 IPMP Engine APIs.....	7
7.4.1 IPMPEngine	7
7.4.2 IPMP Information creation	7
7.4.3 IPMP Information access	7
7.5 Media Framework Engine APIs	7
7.5.1 Access::Video APIs	8
7.5.2 Access::Audio APIs	8
7.5.3 Access::Image APIs	8
7.5.4 Access::Graphics3D APIs	9
7.5.5 Creation::Video APIs	9
7.5.6 Creation::Audio APIs	9
7.5.7 Creation::Image APIs	9
7.5.8 Creation::Graphics3D APIs	9
7.6 Metadata Engine APIs	9
7.6.1 MetadataEngine	9
7.6.2 Metadata creation	9
7.6.3 Metadata Access	10
7.7 Digital Item Streaming Engine APIs	10
7.7.1 DISEngine.....	10
7.7.2 Digital Item Streaming Information creation.....	11
7.7.3 Streaming of Digital Items	11
7.8 Digital Item Adaptation Engine APIs	11
7.8.1 Introduction.....	11
7.8.2 Digital Item Adaptation Access.....	11
7.8.3 Digital Item Adaptation Creation.....	12

7.9	Event Reporting Engine APIs	12
7.9.1	EREngine	12
7.10	Content Protocol Engine APIs	12
7.10.1	ContentProtocolEngine	12
7.10.2	Content Identification Protocol	13
7.10.3	Content Authentication Protocol	13
7.10.4	Content Store Protocol	13
7.10.5	Content Access Protocol	13
7.11	License Protocol Engine APIs	13
7.11.1	LicenseProtocolEngine	13
7.11.2	Store License Protocol	14
7.11.3	Access License Protocol	14
7.11.4	Revoke License Protocol	14
7.12	IPMP Tool Protocol Engine APIs	14
7.12.1	IPMPToolProtocolEngine	14
7.12.2	IPMP Tool Access Protocol	14
7.13	Content Search Engine APIs	14
7.13.1	ContentSearchEngine	14
7.14	Security Engine APIs	15
7.14.1	SecurityEngine	15
7.14.2	CertificateManager	15
7.14.3	KeyManager	15
7.14.4	SecureDeviceManager	15
7.14.5	SecureRepositoryManager	15
7.15	MVCO Engine APIs	15
7.16	Domain Engine APIs	16
7.16.1	DomainEngine	16
7.16.2	DomainManager	16
7.17	Rendering Engine APIs	16
8	MXM Orchestrator APIs	17
8.1	Digital Item Adaptation Orchestrator Engine APIs	17
8.1.1	Introduction	17
8.1.2	Adapt	17
8.2	IPMP Orchestrator Engine APIs	17
8.3	REL Orchestrator Engine APIs	17
8.4	Player Orchestrator Engine APIs	17
Annex A (informative)	MXM Configuration	18
Bibliography	26

iTech STANDARD PREVIEW
(standards.itech.ai)

ISO/IEC 23006-2:2011
<https://standards.itech.ai/catalog/standards/siv/a790ac4f-526a-4409-977a-d71298ccc187/iso-iec-23006-2-2011>

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 23006-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 23006 consists of the following parts, under the general title *Information technology — MPEG extensible middleware (MXM)*:

- *Part 1: MXM architecture and technologies* [ISO/IEC 23006-2:2011
https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011](https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011)
- *Part 2: MXM API*
- *Part 3: MXM reference software*
- *Part 4: MXM protocols*

Introduction

ISO/IEC 23006 is a suite of standards that has been developed for the purpose of enabling the easy design and implementation of media-handling value chains whose devices interoperate because they are all based on the same set of technologies accessible from the MXM middleware.

This will enable the development of a global market of

- MXM applications that can run on MXM devices thanks to the existence of a standard MXM application API,
- MXM devices executing MXM applications thanks to the existence of a standard MXM architecture,
- MXM engines thanks to the existence of standard MXM architecture and standard APIs, and
- innovative business models because of the ease to design and implement media-handling value chains whose devices interoperate because they are all based on the same set of technologies.

iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 23006-2:2011](https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011)

<https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011>

Information technology — MPEG extensible middleware (MXM) —

Part 2: MXM API

1 Scope

This part of ISO/IEC 23006 specifies a set of Application Programming Interfaces (APIs) so that MXM Applications executing on an MXM Device can access the standard multimedia technologies contained in its Middleware as MXM Engines, as specified by ISO/IEC 23006-1.

The APIs belong to the following two classes:

- the MXM Engine APIs, i.e. the collection of the individual MXM Engine APIs providing access to a single MPEG technology (e.g. video coding) or to a group of MPEG technologies where this is convenient;
- the MXM Orchestrator API, i.e. the API of the special MXM Engine that is capable of creating chains of MXM engines to execute a high-level application call such as Play, as opposed to the typically low-level MXM Engine API calls.

[ISO/IEC 23006-2:2011](https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011)

<https://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011>

2 Normative references

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

ISO/IEC 10918-1, *Information technology — Digital compression and coding of continuous-tone still images: Requirements and guidelines*

ISO/IEC 13818-2, *Information technology — Generic coding of moving pictures and associated audio information: Video*

ISO/IEC 14496-2, *Information technology — Coding of audio-visual objects — Part 2: Visual*

ISO/IEC 14496-3, *Information technology — Coding of audio-visual objects — Part 3: Audio*

ISO/IEC 14496-10, *Information technology — Coding of audio-visual objects — Part 10: Advanced Video Coding*

ISO/IEC 14496-11, *Information technology — Coding of audio-visual objects — Part 11: Scene description and application engine*

ISO/IEC 14496-16, *Information technology — Coding of audio-visual objects — Part 16: Animation Framework eXtension (AFX)*

ISO/IEC 14496-25, *Information technology — Coding of audio-visual objects — Part 25: 3D Graphics Compression Model*

ISO/IEC 23006-2:2011(E)

ISO/IEC 15444-1, *Information technology — JPEG 2000 image coding system: Core coding system*

ISO/IEC 15948, *Information technology — Computer graphics and image processing — Portable Network Graphics (PNG): Functional specification*

ISO/IEC 21000-2, *Information technology — Multimedia framework (MPEG-21) — Part 2: Digital Item Declaration*

ISO/IEC 21000-4, *Information technology — Multimedia framework (MPEG-21) — Part 4: Intellectual Property Management and Protection Components*

ISO/IEC 21000-5, *Information technology — Multimedia framework (MPEG-21) — Part 5: Rights Expression Language*

ISO/IEC 21000-9, *Information technology — Multimedia framework (MPEG-21) — Part 9: File Format*

ISO/IEC 21000-15, *Information technology — Multimedia framework (MPEG-21) — Part 15: Event Reporting*

ISO/IEC 21000-18, *Information technology — Multimedia framework (MPEG-21) — Part 18: Digital Item Streaming*

ISO/IEC 21000-19, *Information technology — Multimedia framework (MPEG-21) — Part 19: Media Value Chain Ontology*

ISO/IEC 23006-1, *Information technology — MPEG extensible middleware (MXM) — Part 1: MXM architecture and technologies*

ISO/IEC 23006-4, *Information technology — MPEG extensible middleware (MXM) — Part 4: MXM protocols*

[ISO/IEC 23006-2:2011](http://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011)

3 Terms and definitions

<http://standards.iteh.ai/catalog/standards/sist/a796ae4f-526a-4409-977a-d7f298eec187/iso-iec-23006-2-2011>

For the purposes of this document, the terms and definitions given in ISO/IEC 23006-1 apply.

4 Abbreviated terms

BBL	Bitstream Binding Language
DIA	Digital Item Adaptation
DID	Digital Item Declaration
DIDL	Digital Item Declaration Language
DII	Digital Item Identification
DIS	Digital Item Streaming
ER	Event Report
ERR	Event Report Request
IPMP	Intellectual Property Management and Protection
MXMD	Media Streaming Device
REL	Rights Expression Language

RTP	Real Time Protocol
RTSP	Real Time Streaming Protocol
URI	Uniform Resource Identifier

5 Namespace conventions

Throughout this part of ISO/IEC 23006, Qualified Names are written with a namespace prefix followed by a colon followed by the local part of the Qualified Name.

For clarity, throughout this part of ISO/IEC 23006, consistent namespace prefixes are used. Table 1 gives these prefixes and the corresponding namespace.

Table 1 — Namespaces and prefixes

Prefix	Corresponding namespace
ipmpdidl	urn:mpeg:mpeg21:2004:01-IPMPDIDL-NS
ipmpmsg	urn:mpeg:mpeg21:2006:07-IPMPMESSAGES-NS
ipmpinfo	urn:mpeg:mpeg21:2004:01-IPMPINFO-NS
didl	urn:mpeg:mpeg21:2002:02-DIDL-NS
didmodel	urn:mpeg:mpeg21:2002:02-DIDMODEL-NS
didl-msx	urn:mpeg:maf:schema:mediastreaming:DIDLExtensions
dii	urn:mpeg:mpeg21:2002:01-DII-NS
r	urn:mpeg:mpeg21:2003:01-REL-R-NS
sx	urn:mpeg:mpeg21:2003:01-REL-SX-NS
mlx	urn:mpeg:mpeg21:2005:01-REL-MIX-NS
xsd	http://www.w3.org/2001/XMLSchema
xsi	http://www.w3.org/2001/XMLSchema-instance
dsig	http://www.w3.org/2000/09/xmldsig#
mxmacp	urn:mpeg:mpeg-m:schema:accesscontentprotocol:2009
mxmaip	urn:mpeg:mpeg-m:schema:accessipmptoolprotocol:2009
mxmalp	urn:mpeg:mpeg-m:schema:accesslicenseprotocol:2009
mxmaucp	urn:mpeg:mpeg-m:schema:authenticatecontentprotocol:2009
mxmbp	urn:mpeg:mpeg-m:schema:baseprotocol:2009
mxmd	urn:mpeg:mpeg-m:schema:domain:2009
mxmdp	urn:mpeg:mpeg-m:schema:domainprotocol:2009
mxmicp	urn:mpeg:mpeg-m:schema:identifycontentprotocol:2009
mxmlrp	urn:mpeg:mpeg-m:schema:revokelicenseprotocol:2009
mxmscp	urn:mpeg:mpeg-m:schema:storecontentprotocol:2009
mxmslp	urn:mpeg:mpeg-m:schema:storelicenseprotocol:2009

The MXM APIs are specified in the Java and C++ languages, as follows:

- a) A high-level description of the interfaces defining the MXM APIs
- b) An html format specification of the MXM APIs with a normative value provided as an attachment to this document

6 Common MXM interfaces and classes

The core part of MXM consists of a number of interfaces and classes which are common to all MXM Engines. These are:

- MXM: class enabling MXM Applications to obtain instances of MXM Engines. MXM acts as a factory instantiating those MXM engines listed in the MXM Configuration file when they are required by MXM Applications.
- MXMEngine: the interface at a highest level of an MXMEngine. Every MXMEngines extend this interface, defining a method enabling other entities to query the specific name of an MXMEngine.
- MXMObject: the basic interface for most of the MXM classes. An MXMObject is a wrapper of specific objects that can therefore be exchanged by MXM Engines. It defines the version of MXM and it provides a number of fundamental methods to know what type of object is indeed an MXMObject wrapping, what is the class name of the wrapped object, getters and setters method for getting/setting an object from/into an MXMObject, etc.
- MXMAdapter: a basic class implementing the MXMObject interface which is a wrapper of specific objects that can therefore be exchanged by MXM Engines. By means of an MXMAdapter it is possible to convert any object into an MXMObject.
- MXMEngineName: an enumeration listing all possible names (types) of MXMEngines, so that any entity can unambiguously determine the name of an MXMEngine.
- MXMException: the highest level of exception thrown by MXM. All other exceptions in MXM extends this abstract class.
- MXMEngineResponse: an enumeration used as a general-purpose return value for a number of methods.

7 MXM Engine APIs

7.1 Digital Item Engine APIs

7.1.1 DIDEngine

The *DIDEngine* interface defines for operating on ISO/IEC 21000-2 Digital Item Declaration (DID) data structures. From classes implementing the DIDEngine interface it is possible to obtain instances of classes performing the main functionalities of this MXM Engine:

- classes to create Digital Items
- classes to access data contained in a Digital Item
- classes to edit a Digital Item

7.1.2 DI Creation

Digital Item creation involves the following interfaces:

- DICreator: an interface defining the methods to create a Digital Item and adding a digital signature to it
- ItemCreator: an interface defining the methods to create a didl:Item after setting all its main properties
- ResourceCreator: an interface for creating a didl:Resource after setting all its main properties

7.1.3 DI Editing

Digital Item editing involves the following interfaces:

- DIEditor: an interface defining the methods to edit a Digital Item
- ItemEditor: an interface defining the methods to edit a didl:Item

7.1.4 DI Access

Digital Item parsing involves the following interfaces:

- DIParser: an interface defining the methods to parse a a Digital Item and retrieve the information contained in it

iTeh STANDARD PREVIEW
(standards.iteh.ai)

7.2 MPEG-21 File Engine APIs

7.2.1 MPEG21FileEngine

[ISO/IEC 23006-2:2011](https://standards.iteh.ai/catalog/standards/ist/796ae4f-536e-4409-977d-732981cc187d/iso-iec-23006-2-2011)

The MPEG21FileEngine interface defines the methods for operating over ISO/IEC 21000-9 MPEG-21 File Format files. From classes implementing the MPEG21FileEngine interface it is possible to obtain instances of:

- classes for creating an MPEG-21 file
- classes for accessing data from an MPEG-21 file

7.2.2 MPEG21 File Creation

Creating an MPEG-21 file involves the following interfaces:

- MPEG21FileCreator: an interface defining the methods to create an MPEG-21 file

7.2.3 MP21 File Access

Creating an MPEG-21 file involves the following interfaces:

- MPEG21FileCreator: an interface defining the methods to create an MPEG-21 file

7.3 REL Engine APIs

7.3.1 RELEngine

The RELEngine interface defines the methods for operating over ISO/IEC 21000-5 Rights Expression Language (REL) data structures. Classes implementing the RELEngine interface act as factories creating instances of classes performing the following functionalities:

- classes to create Rights Expressions
- classes to access data contained in Rights Expression
- classes to authorise users to exercise rights

7.3.2 Rights Expression creation

Creating a REL statement involves the following interfaces:

- LicenseCreator: an interface defining the methods to create an r:license element
- GrantCreator: an interface defining the methods to create an r:grant
- DigitalResourceCreator: an interface defining the methods to create an r:digitalResource
- ProtectedResourceCreator: an interface defining the methods to create an m1x:protectedResource
- IdentityHolderCreator: an interface defining the methods to create an m1x:identityHolder
- IssuerCreator: an interface defining the methods to create an r:issuer
- KeyHolderCreator: an interface defining the methods to create an r:keyHolder

7.3.3 Rights Expression access

Parsing a REL statement involves the following interfaces:

- LicenseParser: an interface defining the methods to parse an r:license element
- GrantParser: an interface defining the methods to parse an r:grant
- DigitalResourceParser: an interface defining the methods to parse an r:digitalResource
- ProtectedResourceParser: an interface defining the methods to parse an m1x:protectedResource
- IdentityHolderParser: an interface defining the methods to parse an m1x:identityHolder
- IssuerParser: an interface defining the methods to parse an r:issuer
- KeyHolderParser: an interface defining the methods to parse an r:keyHolder

7.3.4 Authorisation

Authorising a user to exercise a right involves the following interfaces:

- AuthorisationManager: an interface defining the methods to authorise a user to exercise a right and retrieve information from the validation operation
- AuthorisationResult: an enumeration defining possible result of an authorisation