
**Information technology — Coding
of audio-visual objects —**

**Part 5:
Reference software**

**AMENDMENT 22: Reference software for
3D Graphics Compression Model (3DGCM)**

Technologies de l'information — Codage des objets audiovisuels —

Partie 5: Logiciel de référence

*AMENDEMENT 22: Logiciel de référence pour modèle
de compression graphique 3D (3DGCM)*

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2009

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

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.

Amendment 22 to ISO/IEC 14496-5:2001 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

Information technology — Coding of audio-visual objects —

Part 5: Reference software

AMENDMENT 22: Reference software for 3D Graphics Compression Model (3DGCM)

After Clause 6, add the following new clause:

7 Reference Software for 3D Graphics Compression Model

7.1 Introduction

The MPEG-4 Part software (called MP25) contains two main modules: MP4XMLEncoder and MP4XMLDecoder. The model was implemented three XML based scene graph formats (COLLADA, X3D and XMT) and the elementary streams supported are 3DMC, BBA, IC, FAMC, JPG and JPEG 2000.

The MP4XMLEncoder is illustrated in Figure AMD22.1 and consists of three main libraries.

(1) libColladaParser, libX3DParser and libXMTParser. Each of these components deals with the appropriate file format (COLLADA, X3D and XMT respectively), parses it, fills the structures with data for mesh, animation, texture and creates a buffer with the rest of the XML file.

(2) libMP4Encoder takes these structures as input, encodes them and creates buffers with standard MPEG-4 elementary streams.

(3) libMP4Mux multiplexes the above elementary streams and creates the MP4 file.

The MP4XMLDecoder is illustrated in Figure AMD22.2 and consists of three main libraries.

(1) MP4toVBLib parses the mp4 file and outputs buffers with standard mp4 elementary streams.

(2) libMP4Decoder decodes the elementary streams in structures for mesh, animation and texture and a buffer for rest of the XML file

(3) libMP4toColladaDecoder uses the decoded structures and reconstructs the XML file.

Since the scene and the object graph contains information that requires lossless compression, this data is compressed by using GZIP. On the other hand, the nature of such data and its dimension (it only contains structures and some parameters, but not geometry animation or texture) makes inappropriate the development of specific compression schema.

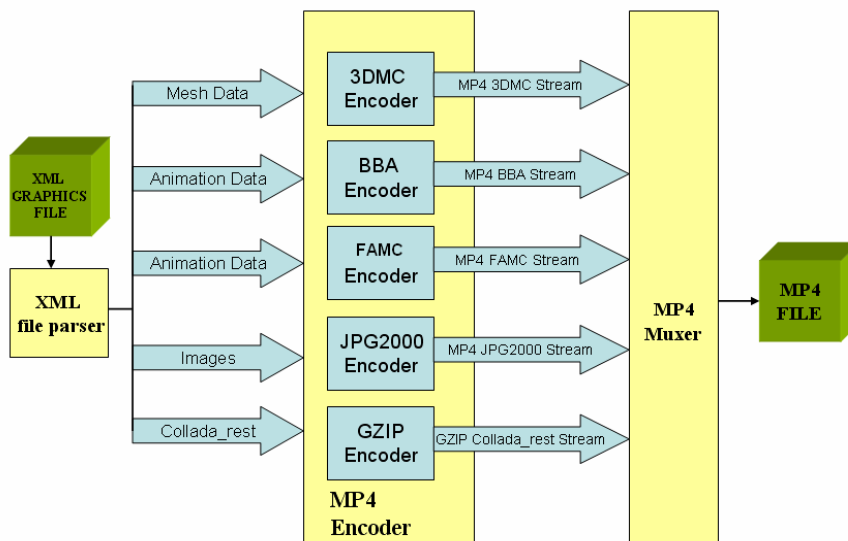


Figure AMD22.1 — MP4XMLEncoder structure

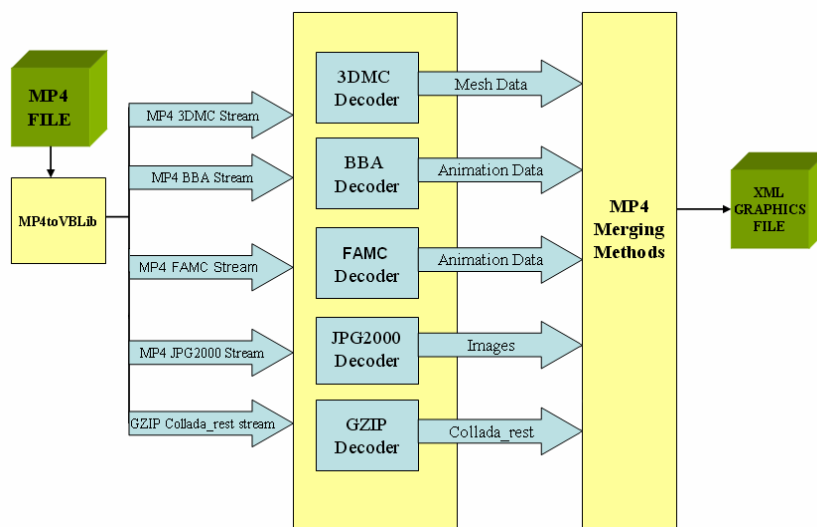


Figure AMD22.2 — MP4XMLDecoder structure

7.2 Folder structure

MP4Collada\Libraries\ libColladaParser	COLLADA file format parser
MP4Collada\Libraries\ MP4ClassesLib	Classes defining data structures
MP4Collada\Libraries\ libMP4Encoder	Set of encoders for mp4 elementary streams
MP4Collada\Libraries\ libMP4Decoder	Set of decoders from mp4 elementary streams
MP4Collada\Libraries\ MP4toVBLib	MP4 file format parser
MP4Collada\Libraries\ libMP4Mux	MP4 file creator
MP4Collada\Libraries\libMP4toColladaDEncoder	MP4 to COLLADA file decoder and COLLADA file creator

7.3 Definition of classes

Class	Files	Description
ColladaParser	ColladaParser.h, ColladaParser.cpp	Load COLLADA file, parse and give at output structures with mesh, animation, texture and rest of collada file
MP43DMC	MP43DMC.h, MP43DMC.cpp	Defines internal structure for mesh data
MP4BoneData	MP4Animation.h, MP4Animation.cpp	Defines internal structure for one bone
MP4Frames	MP4Animation.h, MP4Animation.cpp	Defines internal structure for one frame
MP4AnimationBuff	MP4Animation.h, MP4Animation.cpp	Define internal structure for animation data
EncodeMP4Streams	EncodeMP4Streams.h, EncodeMP4Streams.cpp	Encoder from internal data structure for mesh, animation and texture in 3DMC,BBA and JPG2000 streams. Encoder from xml to gzip COLLADA_rest
MPG_IfsQ_C_Ifs : Public MPG_IfsQ_C	MPG_IfsQ_C_Ifs.h, MPG_IfsQ_C_Ifs.cpp	Class used to put data in the 3DMC encoder
MPG_IfsQ_E_Ifs : Public MPG_IfsQ_E	MPG_IfsQ_E_Ifs.h, MPG_IfsQ_E_Ifs.cpp	Class used to get the data from the 3DMC encoder
DecodeMP4Streams	DecodeMP4Streams.h, DecodeMP4Streams.cpp	decoder from in 3DMC,BBA and JPG2000 streams to internal data structure for mesh, animation and texture decoder from gzip for rest of COLLADA
Matrix4	matrix4.h, matrix4.cpp	Define matrix 4x4 and manipulation with matrices
Ostream	Ostream.h, ostream.cpp	Define method for read/write in stream
MP4toVBLib	MP4toVBLib.h, MP4toVBLib.cpp	Load mp4 file and give as output separate tracks and meta data
MakeOD	MakeOD.h, MakeOD.cpp	Class that create Object descriptor for mp4 file
SampleData	MP4Create.h	Class that define data structure for one sample in mp4 file
TrackData	MP4Create.h	Class that define data structure for one track in mp4 file
MP4Mux	MP4Mux.h, Mux.cpp	Create mp4 file from standard mp4 streams
MP4toColladaDecode	MP4toColladaDecode.h, MP4toColladaDecode.cpp	Create COLLADA file from mp4 file

