

ETSI TS 103 973 V1.1.1 (2024-10)



Coded Multisource Media Format (CMMF) for Content Distribution and Delivery

(<https://standards.iteh.ai>)

Document Preview

[ETSI TS 103 973 V1.1.1 \(2024-10\)](https://standards.iteh.ai/catalog/standards/etsi/e2cef66a-def0-40ea-ba03-a903fc002f4d/etsi-ts-103-973-v1-1-1-2024-10)

<https://standards.iteh.ai/catalog/standards/etsi/e2cef66a-def0-40ea-ba03-a903fc002f4d/etsi-ts-103-973-v1-1-1-2024-10>

EBU

Reference

DTS/JTC-112

Keywords

broadband, broadcast, CDN, container,
distribution, multimedia, multi-path, multi-source,
network coding, robustness

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from the
ETSI [Search & Browse Standards](#) application.

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format on [ETSI deliver](#).

Users should be aware that the present document may be revised or have its status changed,
this information is available in the [Milestones listing](#).

If you find errors in the present document, please send your comments to
the relevant service listed under [Committee Support Staff](#).

If you find a security vulnerability in the present document, please report it through our
[Coordinated Vulnerability Disclosure \(CVD\)](#) program.

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2024.

© European Broadcasting Union 2024.

All rights reserved.

Contents

Intellectual Property Rights	15
Foreword.....	15
Modal verbs terminology.....	15
Introduction	16
1 Scope	17
2 References	17
2.1 Normative references	17
2.2 Informative references.....	18
3 Definition of terms, symbols and abbreviations.....	18
3.1 Terms.....	18
3.2 Symbols.....	20
3.3 Abbreviations	20
4 Overview	21
4.0 Introduction	21
4.1 Source data	23
4.2 CMMF bitstream creation	23
4.2.1 Encoding CMMF	23
4.2.2 Symbol groups in CMMF	25
4.2.3 Decoding CMMF	25
4.2.4 Mapping to/from CMMF	27
4.3 Media delivery using CMMF	28
4.3.1 Overview	28
4.3.2 CMMF transport objects and transport sessions	29
4.3.3 CMMF delivery architecture reference points	29
4.3.4 CMMF delivery procedure	32
4.3.5 CMMF Configuration Information.....	35
4.3.6 CMMF as a Content Delivery Protocol	35
4.4 Overview of the Specification	35
5 Bitstream syntax	36
5.0 Bitstream organization	36
5.1 Semantics of syntax specification.....	37
5.1.1 Pseudocode syntax.....	37
5.1.2 Bitstream variable syntax.....	37
5.1.3 Bitstream structure syntax	37
5.1.4 Iteration and conditional operators	38
5.1.5 Boolean operations	38
5.1.6 Labels and comments.....	39
5.1.7 Operational variables not in the bitstream	39
5.1.8 Arrays	39
5.1.9 Bit field encoding	39
5.2 Syntax specification	40
5.2.1 cmmf_bitstream()	40
5.2.2 subatom()	40
5.2.3 sync()	41
5.2.4 bitstream_header()	42
5.2.5 block_header()	42
5.2.6 addl_cce_parameters()	44
5.2.7 prng_parameters().....	45
5.2.8 packet()	45
5.2.9 packet_header().....	45
5.2.10 encoder_content_info()	46
5.2.11 media_segment_info()	47
5.2.12 cmmf_time()	49

5.2.13	chunked_subatom()	50
5.2.14	block_group_directory()	50
5.2.15	fb_integrity()	51
5.2.16	packet_integrity()	51
5.2.17	coefficient_vector()	51
5.2.18	extension()	52
5.2.19	packet_header_only()	52
5.2.20	rfc5052_information()	52
5.2.21	packet_group()	53
5.2.22	packet_group_header()	53
5.2.23	num_bits_code()	54
5.2.24	block_index_or_count_value()	55
5.2.25	multi_block_packet_group()	55
5.2.26	mbpg_header()	55
6	Bitstream description	56
6.0	Introduction	56
6.1	Description of bitstream elements	56
6.1.1	cmmf_bitstream()	56
6.1.2	subatom()	57
6.1.2.0	Introduction	57
6.1.2.1	subatom_id, subatom_id_ext	57
6.1.2.2	b_bitstream_id_present	57
6.1.2.3	sas_bits	58
6.1.2.4	bitstream_id	58
6.1.2.5	subatom_size	58
6.1.3	sync()	58
6.1.3.0	Introduction	58
6.1.3.1	syncword	58
6.1.3.2	version	58
6.1.3.3	b_content_encode_uuid	58
6.1.3.4	content_encode_uuid	58
6.1.4	bitstream_header()	59
6.1.4.0	Introduction	59
6.1.4.1	content_source_size	59
6.1.4.2	content_source_type	59
6.1.4.3	b_content_source_split	59
6.1.4.4	content_soure_split_start, content_source_split_end	59
6.1.4.5	code_type, code_type_ext	59
6.1.4.6	b_rfc5052, rfc5052_information(), b_addl_rfc5052_information_present	60
6.1.4.7	block_count_minus1, block_count	60
6.1.4.8	b_content_block_separate_sources	61
6.1.4.9	num_content_block_sources_minus1	61
6.1.4.10	b_profile_information_present	61
6.1.4.11	profile_type_size, profile_type	61
6.1.4.12	profile_description	61
6.1.4.13	b_block_cc_encrypted	61
6.1.4.14	bitstream_encryption_key_id_size_exp	61
6.1.4.15	bitstream_encryption_key_id	61
6.1.5	block_header()	62
6.1.5.0	Introduction	62
6.1.5.1	block_index	62
6.1.5.2	block_size	62
6.1.5.3	block_symbol_size	62
6.1.5.4	bns_bits	62
6.1.5.5	block_num_symbols	62
6.1.5.6	b_block_max_symbol_index_present	62
6.1.5.7	bmsi_bits	62
6.1.5.8	block_max_symbol_index	63
6.1.5.9	b_block_content_source_index_present	63
6.1.5.10	block_content_source_index	63
6.1.5.11	b_block_composite_sources	63

6.1.5.12	block_num_composite_sources_minus1	63
6.1.5.13	bcss_bits	63
6.1.5.14	block_composite_source_size	63
6.1.5.15	b_addl_block_coding_info_present	63
6.1.5.16	addl_block_coding_mask	64
6.1.5.17	b_addl_window_info_present	64
6.1.5.18	b_reserved_block_coding_params_present	64
6.1.5.19	block_mask	64
6.1.5.20	b_sufficient_symbols_present	65
6.1.5.21	bsp_bits	65
6.1.5.22	block_symbols_present	65
6.1.5.23	block_field_size_exp	65
6.1.5.24	Encrypted Coefficients	66
6.1.5.24.0	Introduction	66
6.1.5.24.1	block_cc_encryption_info_size_bits_code	66
6.1.5.24.2	byte_align	66
6.1.5.24.3	block_cc_encryption_info_size	66
6.1.5.24.4	block_cc_encryption_algorithm	67
6.1.5.24.5	block_cc_encryption_mode	67
6.1.5.24.6	block_cce_key_size_exp, block_cce_key	67
6.1.5.24.7	b_addl_block_cce_params_present	67
6.1.5.24.8	addl_cce_parameters()	67
6.1.5.25	Pseudorandom Noise Generator (PRNG)	68
6.1.5.25.0	Introduction	68
6.1.5.25.1	prng_type	69
6.1.5.25.2	prng_seed_bits_code	69
6.1.5.25.3	prng_seed	69
6.1.5.25.4	prng_density_percentage	69
6.1.6	packet()	69
6.1.6.0	Introduction	69
6.1.6.1	packet_block_index	70
6.1.6.2	coded_symbol	70
6.1.7	packet_header()	70
6.1.7.0	Introduction	70
6.1.7.1	b_systematic_symbol	70
6.1.7.2	packet_mask	71
6.1.7.3	psi_bits	72
6.1.7.4	packet_symbol_index	72
6.1.7.5	Encryption Parameters	72
6.1.7.5.0	Introduction	72
6.1.7.5.1	b_systematic_symbol_encrypted	72
6.1.7.5.2	b_addl_packet_cce_params_present	72
6.1.7.6	window_start_index, window_stop_index	73
6.1.7.7	byte_align	73
6.1.8	encoder_content_info()	73
6.1.8.0	Introduction	73
6.1.8.1	b_encoder_id_present	73
6.1.8.2	encoder_uuid	73
6.1.8.3	Content Identification	73
6.1.8.4	b_content_id_present	73
6.1.8.4.0	Introduction	73
6.1.8.4.1	content_id_type	74
6.1.8.4.2	content_id_size_minus1	74
6.1.8.4.3	content_id	74
6.1.8.5	b_content_location_present	74
6.1.8.6	content_location_size, content_location	74
6.1.8.7	b_content_type_present	74
6.1.8.8	content_type_size, content_type	74
6.1.8.9	b_content_header_present	75
6.1.8.10	content_header_size, content_header	75
6.1.8.11	b_file_integrity_present	75
6.1.8.12	b_media_preso_dur_present	75

6.1.9	media_segment_info()	75
6.1.9.0	Introduction	75
6.1.9.1	media_segment_block_index, media_segment_block_index_ext	75
6.1.9.2	media_segment_index, media_segment_index_ext	76
6.1.9.3	b_composite_source_index_present	76
6.1.9.4	media_segment_composite_source_index	76
6.1.9.5	b_asset_name_present	76
6.1.9.6	asset_name_size, asset_name	76
6.1.9.7	segment_tag_mask	76
6.1.9.8	segidx_bits, segcnt_bits	76
6.1.9.9	segment_index	76
6.1.9.10	segment_count	76
6.1.9.11	b_media_mime_type_present	77
6.1.9.12	media_mime_type_size, media_mime_type	77
6.1.9.13	b_media_codec_present	77
6.1.9.14	media_codec_size, media_codec	77
6.1.9.15	b_bit_rate_present	77
6.1.9.16	bit_rate_bits_code	77
6.1.9.17	bit_rate	77
6.1.9.18	b_ms_content_type_present	77
6.1.9.19	ms_content_type	78
6.1.9.20	b_ms_content_type_info_present	78
6.1.9.21	b_aspect_ratio_present	78
6.1.9.22	sample_aspect_ratio	78
6.1.9.23	sar_width, sar_height	78
6.1.9.24	b_dynamic_resolution_video	78
6.1.9.25	b_resolution_present	78
6.1.9.26	resolution_width, resolution_height	78
6.1.9.27	b_frame_rate_present	78
6.1.9.28	frame_rate	79
6.1.9.29	b_hdr_info_present	79
6.1.9.30	hdr_compatibility_mask	79
6.1.9.31	b_addl_hdr_info_present	79
6.1.9.32	hdr_compat_mask_index	80
6.1.9.33	hdr_profile	80
6.1.9.34	hdr_level	80
6.1.9.35	hdr_compatibility_id	80
6.1.9.36	b_addl_video_info_present	80
6.1.9.37	b_sampling_freq_present	80
6.1.9.38	b_sampling_freq_is_48k	80
6.1.9.39	sampling_frequency	80
6.1.9.40	b_audio_config_present	80
6.1.9.41	audio_channel_config	81
6.1.9.42	b_audio_props_present	82
6.1.9.43	b_virtualized_bin	82
6.1.9.44	b_object_audio	82
6.1.9.45	b_complexity_index_present	82
6.1.9.46	complexity_index	82
6.1.9.47	b_addl_audio_info_present	82
6.1.9.48	b_addl_ms_content_type_info_present	82
6.1.9.49	accessibility_mask	82
6.1.9.50	language_size, language	83
6.1.10	cmmf_time()	83
6.1.10.0	Introduction	83
6.1.10.1	b_ddhhmmss	83
6.1.10.2	DD:HH:MM:SS format	83
6.1.10.3	int_seconds_bits_code	83
6.1.10.4	int_seconds	84
6.1.10.5	b_fract_seconds_present	84
6.1.10.6	fract_seconds_bits_code	84
6.1.10.7	fract_seconds	84
6.1.11	chunked_subatom()	84

6.1.11.0	Introduction.....	84
6.1.11.1	chunk_segment_id	85
6.1.11.2	chunk_segment_index.....	85
6.1.11.3	num_chunk_segments.....	85
6.1.11.4	original_subatom_id, original_subatom_id_ext.....	85
6.1.11.5	oss_bits.....	85
6.1.11.6	original_subatom_size.....	85
6.1.11.7	byte_align.....	85
6.1.11.8	chunked_subatom_segment_data.....	85
6.1.12	block_group_directory()	86
6.1.12.0	Introduction.....	86
6.1.12.1	block_group_dir_mask.....	86
6.1.12.2	block_header_subatom_offset[block]	86
6.1.12.3	num_packet_groups[block]	86
6.1.12.4	packet_group_index[block][pg]	86
6.1.12.5	packet_group_subatom_offset[block][pg]	86
6.1.12.6	num_multi_block_packet_groups	87
6.1.12.7	multi_block_packet_group_subatom_offset[mbpg].....	87
6.1.13	fb_integrity()	87
6.1.13.0	Introduction.....	87
6.1.13.1	fb_hash_type	87
6.1.13.2	fb_hash_algorithm	87
6.1.13.3	fb_hash_size.....	88
6.1.13.4	b_fb_integrity_ext.....	88
6.1.13.5	fb_hash.....	88
6.1.14	packet_integrity()	88
6.1.14.0	Introduction.....	88
6.1.14.1	packet_hash_algorithm	89
6.1.14.2	packet_hash_size.....	89
6.1.14.3	b_packet_integrity_ext.....	89
6.1.14.4	packet_hash.....	89
6.1.15	coefficient_vector().....	90
6.1.15.0	Introduction.....	90
6.1.15.1	coded_symbol_coeff[index].....	90
6.1.16	extension()	90
6.1.16.0	Introduction.....	90
6.1.16.1	extension_byte_size	90
6.1.17	packet_header_only().....	90
6.1.18	packet_group()	90
6.1.18.0	Introduction.....	90
6.1.18.1	packet_group_block_index	91
6.1.18.2	packet_group_index	91
6.1.18.3	pgns_bits	91
6.1.18.4	packet_group_num_symbols.....	91
6.1.18.5	packet_group_type	91
6.1.18.6	coded_symbol	91
6.1.19	packet_group_header()	91
6.1.19.0	Introduction.....	91
6.1.19.1	packet_group_symbol_arrangement	92
6.1.19.2	packet_group_mask.....	92
6.1.19.3	pgsi_bits	93
6.1.19.4	packet_group_symbol_index	93
6.1.19.5	pgfsi_bits.....	93
6.1.19.6	packet_group_first_symbol_index.....	93
6.1.19.7	packet_group_index_difference	93
6.1.19.8	pgfsii_bits.....	94
6.1.19.9	Symbol Arrangements in a Packet Group	94
6.1.19.10	Encryption Parameters	94
6.1.19.10.0	Introduction	94
6.1.19.10.1	b_addl_packet_group_cce_params_present	95
6.1.20	num_bits_code()	95
6.1.20.0	Introduction.....	95

6.1.20.1	bits_code	95
6.1.21	block_index_or_count_value()	95
6.1.21.0	Introduction.....	95
6.1.21.1	block_index_or_count, block_index_or_count_ext	95
6.1.22	multi_block_packet_group().....	95
6.1.22.0	Introduction.....	95
6.1.22.1	mbpg_index.....	96
6.1.22.2	mbpg_start_block_index	96
6.1.22.3	mbpg_num_blocks	96
6.1.22.4	mbpg_num_symbols	96
6.1.22.5	coded_symbol	96
6.1.23	mbpg_header().....	97
6.1.23.0	Introduction.....	97
6.1.23.1	mbpg_symbol_arrangement.....	97
6.1.23.2	mbpgsi_bits	97
6.1.23.3	mbpg_source_block_index, mbpg_symbol_index	98
6.1.23.4	mbpgfsi_bits.....	98
6.1.23.5	mbpg_first_symbol_index, b_mbp_g_is_symbol_group_subset, mbpg_symbol_group_subset_index.....	98
6.1.23.6	mbpg_index_difference.....	98
6.1.23.7	mbpgsai_bits_code.....	98
6.1.23.8	Symbol Arrangements in a Multiple Block Packet Group	99
6.1.23.9	b_mbp_g_integrity_present.....	103
6.1.23.10	b_mbp_g_header_ext_present.....	103
7	Design considerations.....	103
7.0	Introduction	103
7.1	Coding coefficients.....	103
7.1.0	Generating coding coefficients using a PRNG	103
7.1.1	Coefficient density control.....	105
7.1.2	Mersenne twister PRNG type	105
7.2	Handling variable source symbol size	105
7.3	Encrypting coding coefficient information.....	106
7.3.0	Introduction.....	106
7.3.1	Using a bitstream/session key and symmetric keys	107
Annex A (normative):	xCD-1	109
A.0	Introduction	109
A.1	Encoding.....	109
A.2	Decoding	112
Annex B (informative):	Media service architecture Examples	114
B.0	Introduction	114
B.1	MPEG-DASH HTTP adaptive streaming service example.....	114
Annex C (informative):	Example bitstreams	119
C.0	Introduction	119
C.1	Multisource Video-on-Demand.....	119
C.1.0	Multisource Video-on-Demand example using code_type xCD-1.....	119
C.1.1	Bitstream construction.....	119
C.1.2	Sync construction	120
C.1.3	Subatom construction	121
C.1.3.1	Bitstream header subatom construction	121
C.1.3.2	Block header subatom construction	123
C.1.3.3	Encoder content information subatom construction	126
C.1.3.4	Media segment information subatom construction.....	128
C.1.3.5	Packet subatom construction - systematic symbol.....	132
C.1.3.6	Packet subatom construction - coded symbol	134

C.2	Encrypted coding coefficient information example using CMMF	136
Annex D (normative):	Content delivery protocol-based instantiations.....	138
D.1	CMMF content delivery protocol principles	138
D.1.1	Introduction	138
D.1.2	FEC Building Block principles	138
D.1.3	FEC Schemes and related information	138
D.1.4	FEC Scheme information in CMMF	139
D.1.5	Configuration Information parameters	141
D.1.6	Example Instantiations	142
D.2	FLUTE-based CMMF CDP Instantiation	142
D.2.1	Introduction	142
D.2.2	Procedures for FLUTE-based CMMF CDP Instantiation	143
D.2.3	Extended File Delivery Table.....	144
D.2.3.1	Semantics	144
D.2.3.2	Extended FDT Schema for CMMF.....	145
D.2.3.3	Extended FDT Description for CMMF.....	146
D.2.3.4	IANA registration for Extended FDT Description.....	146
D.2.4	Transport object formats	147
D.2.4.1	General.....	147
D.2.4.2	Source objects.....	147
D.2.4.3	Coded/repair objects	148
D.2.4.3.1	General	148
D.2.4.3.2	Mapping of FEC Payload ID information to transport blocks	150
D.2.4.3.2.1	General	150
D.2.4.3.2.2	Symbol group arrangement 3: encoding symbol interleaving	151
D.2.4.3.2.3	Symbol group arrangement 2: source symbol interleaving	153
D.2.5	Examples	154
D.2.5.1	Single file - source and partial encoding object.....	154
D.2.5.2	Multiple files - self-contained objects including source symbols	155
D.2.6	Potential receiver operation.....	155
History		156

ETSI TS 103 973 V1.1.1 (2024-10)

<https://standards.iteh.ai/catalog/standards/etsi/e2cef66a-def0-40ea-ba03-a903fc002f4d/etsi-ts-103-973-v1-1-1-2024-10>

List of figures

Figure 1: CMMF layer within multimedia transport stack.....	22
Figure 2: CMMF encode example.....	24
Figure 3: Example symbol groups.....	25
Figure 4: CMMF decode example.....	26
Figure 5: CMMF bitstream encoding/packaging.....	27
Figure 6: CMMF bitstream decoding.....	27
Figure 7: Generic Example CMMF delivery/transport sessions	28
Figure 8: Delivery session model for CMMF	29
Figure 9: CMMF delivery architecture reference points	30
Figure 10: HTTP-Based Adaptive Streaming CMMF Delivery Procedure Example	33
Figure 11: High-level CMMF bitstream organization.....	36
Figure 12: Example symbol arrangement for mbpg_symbol_arrangement=0010b (Source-Symbol Interleaved Arrangement)	101
Figure 13: Example symbol arrangement for mbpg_symbol_arrangement=0011b (Encoded-Symbol Interleaved Arrangement)	103
Figure 14: Block PRNG	104
Figure 15: Packet PRNG	104
Figure 16: Handling variable size data.....	106
Figure 17: Bitstream, block, key management.....	107
Figure 18: Client process for decryption.....	108
Figure 19: Bitstream and key download process.....	108
Figure A.1: xCD-1 segment to symbols	110
Figure A.2: xCD-1 symbol vector and coefficient matrix.....	111
Figure A.3: xCD-1 coefficient matrix	111
Figure A.4: xCD-1 coded symbol vector and decoded coefficient matrix	112
Figure A.5: xCD-1 symbol reconstruction	113
Figure A.6: xCD-1 data extraction	113
Figure B.1: MPEG-DASH with CMMF Delivery System Example.....	114
Figure B.2: CMMF reference architecture in relation to MPEG-DASH HTTP adaptive streaming example	115
Figure B.3: Example MPEG-DASH master manifest	116
Figure B.4: CMMF bitstreams generated to deliver the MPEG-DASH packaged content	116
Figure B.5: CMMF request and content delivery example for MPEG-DASH.....	117
Figure C.1: Multi-source CDN-client communication.....	119
Figure C.2: CMMF multisource example bitstream arrangement.....	120

Figure C.3: Bitstream with example encrypted coding coefficient information.....	137
Figure D.1: High Level Procedure for a FLUTE-based CMMF delivery	143
Figure D.2: Formation of transport objects from source objects	151
Figure D.3: Symbol arrangement for arrangement 3.....	152
Figure D.4: Symbol arrangement for arrangement 2.....	154

iTeh Standards
(<https://standards.iteh.ai>)
Document Preview

[ETSI TS 103 973 V1.1.1 \(2024-10\)](https://standards.iteh.ai/catalog/standards/etsi/e2cef66a-def0-40ea-ba03-a903fc002f4d/etsi-ts-103-973-v1-1-1-2024-10)

<https://standards.iteh.ai/catalog/standards/etsi/e2cef66a-def0-40ea-ba03-a903fc002f4d/etsi-ts-103-973-v1-1-1-2024-10>

List of tables

Table 1: Example mapping.....	28
Table 2: Example of bitstream variable.....	37
Table 3: Example of bitstream structure	38
Table 4: Example of shorthand If syntax format.....	38
Table 5: Example of expanded If syntax format	38
Table 6: Example of Boolean operation syntax format	38
Table 7: Example of bitstream label.....	39
Table 8: Example of bitstream comment.....	39
Table 9: Example of operational variable format	39
Table 10: Bit field descriptors	40
Table 11: Syntax of cmmf_bitstream().....	40
Table 12: Syntax of subatom()	40
Table 13: Syntax of sync()	41
Table 14: Syntax of bitstream_header()	42
Table 15: Syntax of block_header()	42
Table 16: Syntax of addl_cce_parameters()	44
Table 17: Syntax of prng_parameters()	45
Table 18: Syntax of packet()	45
Table 19: Syntax of packet_header().....	45
Table 20: Syntax of encoder_content_info()	46
Table 21: Syntax of media_segment_info()	47
Table 22: Syntax of cmmf_time()	49
Table 23: Syntax of chunked_subatom().....	50
Table 24: Syntax of block_group_directory()	50
Table 25: Syntax of fb_integrity()	51
Table 26: Syntax of packet_integrity()	51
Table 27: Syntax of coefficient_vector().....	51
Table 28: Syntax of extension()	52
Table 29: Syntax of packet_header_only().....	52
Table 30: Syntax of rfc5052_information().....	52
Table 31: Syntax of packet_group()	53
Table 32: Syntax of packet_group_header()	53
Table 33: Syntax of num_bits_code()	54

Table 34: Syntax of block_index_or_count_value()	55
Table 35: Syntax of multi_block_packet_group()	55
Table 36: Syntax of mbpg_header()	55
Table 37: subatom_id meaning	57
Table 38: Subatom dependencies	57
Table 39: content_source_type meaning	59
Table 40: code_type meaning	59
Table 41: Object Transmission Information Fields	60
Table 42: addl_block_coding_mask meaning	64
Table 43: block_mask meaning	64
Table 44: block_field_size_exp information	65
Table 45: block_cc_encryption_info_size_bits_code meaning	66
Table 46: block_cc_encryption_algorithm meaning	67
Table 47: block_cc_encryption_mode meaning	67
Table 48: cce_parameter_type meaning	68
Table 49: block_prng_type meaning	69
Table 50: block_prng_seed_bits_code meaning	69
Table 51: packet_mask meaning	71
Table 52: content_id_type description	74
Table 53: segment_tag_mask meaning	76
Table 54: bit_rate_bits_code meaning	77
Table 55: ms_content_type description	78
Table 56: frame_rate values	79
Table 57: hdr_compatibility_mask values	79
Table 58: sampling_frequency values	80
Table 59: audio_channel_config values	81
Table 60: accessibility_mask meaning	82
Table 61: DD:HH:MM:SS format fields and meaning	83
Table 62: int_seconds_bits_code meaning	83
Table 63: fract_seconds_bits_code meaning	84
Table 64: fract_seconds range and divisor	84
Table 65: block_group_dir_mask meaning	86
Table 66: fb_hash_type meaning	87
Table 67: fb_hash_algorithm meaning	87
Table 68: fb_hash_size meaning	88

Table 69: Allowed fb_hash_size values based on fb_hash_algorithm	88
Table 70: packet_hash_algorithm meaning	89
Table 71: packet_hash_size meaning	89
Table 72: Allowed packet_hash_size values based on packet_hash_algorithm	89
Table 73: packet_group_type meaning	91
Table 74: packet_group_symbol_arrangement Meaning	92
Table 75: packet_group_mask meaning	92
Table 76: Arithmetic coded symbol arrangement in a packet group pseudocode	94
Table 77: bits_code meaning	95
Table 78: mbpg_symbol_arrangement Meaning	97
Table 79: mbpgsai_bits_code meaning	98
Table 80: Separate systematic and interleaved coded symbol arithmetic sequences symbol arrangement in a multi-block group pseudocode	99
Table 81: Interleaved by block coded symbol arithmetic sequences symbol arrangement in a multi-block group pseudocode	102
Table 82: Density controller pseudocode	105
Table 83: Mersenne twister pseudocode	105
Table C.1: sync() structure construction	120
Table C.2: Bitstream header subatom construction	121
Table C.3: Block header subatom construction	123
Table C.4: Encoder content info subatom construction	126
Table C.5: Media segment info subatom construction	128
Table C.6: Packet subatom - systematic packet construction	132
Table C.7: Packet subatom - coded packet construction	134
Table D.1: Parameters and Coding of FEC schemes used in CMMF	139
Table D.2: FEC scheme parameters to CMMF mapping	140
Table D.3: Example Configuration Information parameters	141