
**Information technology — Generic digital
audio-visual systems —**

**Part 6:
Information representation**

*Technologies de l'information — Systèmes audiovisuels numériques
génériques —*

Partie 6: Représentation des informations

Reference number
ISO/IEC 16500-6:1999(E)



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.

© ISO/IEC 1999

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 22 734 10 79
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

Contents	Page
Foreword	vii
Introduction	viii
1. Scope.....	1
2. Normative references.....	2
3. Definitions	4
4. Acronyms and abbreviations	6
5. Conventions.....	8
6. Monomedia Components	9
6.1. Character Information	9
6.2. Text Information	10
6.2.1 Decoding requirements on HTML support	10
6.2.2 HTML encoding.....	11
6.2.3 HTML mapping to MHEG elements.....	11
6.2.4 Informative list of supported HTML Tags.....	11
6.3. Outline Font Format	13
6.3.1 Requirements.....	13
6.3.2 Font Format Specification.....	14
6.3.3 Font rendering.....	14
6.4. Language Information	16
6.5. Service Information	16
6.6. Telephone Numbers	16
6.7. Compressed Audio	17
6.7.1 Compressed audio coding using MPEG-1 Audio	17
6.7.2 Compressed audio coding using ATSC A/52 Audio	17
6.8. Scaleable Audio	18
6.9. Linear Audio.....	18
6.10. Compressed Video	19
6.10.1 Coding constraints for video with a resolution up to ITU-R 601.....	19
6.10.2 Coding constraints for video with a resolution beyond ITU-R 601	22
6.10.3 Decoding tool requirements	24
6.11. Still Pictures	25
6.11.1 Normal resolution still pictures.....	25
6.11.2 Higher resolution still pictures	25
6.12. Compressed Graphics	26
6.12.1 Requirements.....	26
6.12.2 Format for Compressed Graphics	26
6.13. Compressed Character Data	28

6.14.	Network Graphics.....	28
7.	Monomedia Streams.....	29
7.1.	Types of Monomedia Components	29
7.2.	Real-time and Stored Monomedia Streams	30
7.3.	Carriage of Monomedia Streams in PES Packets	31
7.3.1	Packetization of MPEG- and ATSC-defined Components	31
7.3.2	Packetization of DVB-defined Components	31
7.3.3	Packetization of DAVIC-defined Components	31
7.4.	Packetization of Compressed Character Data.....	31
8.	Transport of Monomedia Streams and Components	32
8.1.	Transport of Real Time Streams.....	32
8.2.	Transport of Stored Streams.....	32
8.3.	Transport of Stand-alone Monomedia Components.....	32
9.	Application Format	33
9.1.	Application Interchange Format	33
9.2.	MHEG-5 Profile for the DAVIC Application Domain.....	34
9.2.1	Object Interchange Format.....	34
9.2.2	Set of Classes	34
9.2.3	Set of Features.....	34
9.2.4	Content Data Encoding	34
9.2.5	Attribute Encoding	35
9.2.6	UserInput Registers.....	36
9.2.7	Constraints on the Use of Variables.....	37
9.2.8	Semantic Constraints on the MHEG-5 Applications	37
9.2.9	EngineEvent.....	37
9.2.10	GetEngineSupport	37
9.2.11	TransitionEffect Parameter of the TransitionTo Elementary Action	38
9.2.12	MHEG-5 Resident Programs	40
9.2.13	Protocol Mapping and External Interaction	45
9.3.	Mapping of MHEG-5 Elements to DSM-CC U-U	47
9.3.1	Stream Events and Normal Play Time Mapping.....	47
9.3.2	Namespace Mapping.....	47
9.3.3	MHEG-5 Object References	48
9.3.4	MHEG-5 Content References	49
9.3.5	Mapping of MHEG-5 ComponentTag to DSM-CC and DVB-SI.....	49
9.3.6	Java class File Names.....	49
9.4.	Core set of Java APIs	51
9.4.1	java.lang	51
9.4.2	java.util.....	51
9.4.3	java.io	51
9.4.4	iso.mheg5	51
9.4.5	The DSM-CC User to User API.....	52
9.4.6	The Service Information (SI) API.....	52
9.4.7	The MPEG-2 Section Filter API.....	52
9.4.8	The Resource Notification API.....	52
9.4.9	The MPEG Component API	52

9.5.	URL Format for Access to Broadcast Services.....	53
9.5.1	Introduction	53
9.5.2	Numerical format	53
9.6.	Run-time execution environment.....	54
9.6.1	Application execution	54
9.6.2	User Input Events.....	54
9.6.3	IDL definition for RTE run remote call	54
9.6.4	Mapping of High-Level API Actions on DSM-CC Primitives	55
10.	DAVIC Reference Model for Contents Decoding	56
10.1.	Scope	56
10.2.	Reference Decoder Model	56
10.3.	DAVIC Application Resource Descriptor	59
10.4.	Minimum ISO/IEC 16500 STU requirements	61
10.5.	Support for Graphics in STUs.....	61
10.6.	Persistent Memory.....	61
11.	Content Packaging and Metadata.....	62
11.1.	Content package structure.....	62
11.1.1	Content Item Elements	62
11.1.2	Content Item.....	62
11.1.3	Content package	63
11.2.	Content Metadata	64
11.2.1	Types.....	64
11.2.2	Semantics of the Loading Commands.....	65
11.2.3	Metadata Mapping	66
11.3.	Content Packaging Format.....	72
11.3.1	Bento	72
11.3.2	Bento definition of DAVIC Objects	72
11.4.	Content Loading Toolset.....	79
11.4.1	Scope.....	79
11.4.2	Toolset requirements.....	79
11.4.3	Toolset features	80
11.4.4	Toolset Specifications	80
11.4.5	Content Loading by CMSL	82
Annex A (normative)	Coding of Outline Fonts	84
Annex B (normative)	Coding of Linear Audio	112
Annex C (normative)	Default CLUT for single bitmaps with compressed graphics	116
Annex D (normative)	PES Packetization of DAVIC defined Monomedia Components.....	120
Annex E (normative)	MPEG-2 Section Filter API.....	122
Annex F (normative)	Resource Notification API	152
Annex G (normative)	MPEG Component API	156

ISO/IEC 16500-6:1999(E)

Annex H (informative) Carriage of Private Data 166

Annex I (informative) Video Input Formats..... 168

Annex J (not used) 170

Annex K (informative) STU Video Display Capabilities..... 172

Annex L (informative) Coding and Carriage of A/52 Audio in ATSC Systems 177

Annex M (informative) Transition Effects for Still Picture Compositions..... 179

Annex N (informative) Example of an OSI NSAP Address Format..... 185

Annex O (informative) Content Metadata Specification Language 187

Annex P (informative) Example of Simple Movie Content Item 193

Bibliography.....195

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. 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 part of ISO/IEC 16500 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 16500-6 was prepared by DAVIC (Digital Audio-Visual Council) and was adopted, under the PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by national bodies of ISO and IEC.

ISO/IEC 16500 consists of the following parts, under the general title *Information technology — Generic digital audio-visual systems*:

- *Part 1: System reference models and scenarios*
- *Part 2: System dynamics, scenarios and protocol requirements*
- *Part 3: Contours: Technology domain*
- *Part 4: Lower-layer protocols and physical interfaces*
- *Part 5: High and mid-layer protocols*
- *Part 6: Information representation*
- *Part 7: Basic security tools*
- *Part 8: Management architecture and protocols*
- *Part 9: Usage information protocols*

Annexes A to G form a normative part of this part of ISO/IEC 16500. Annexes H to P are for information only.

Introduction

ISO/IEC 16500 defines the minimum tools and dynamic behavior required by digital audio-visual systems for end-to-end interoperability across countries, applications and services. To achieve this interoperability, it defines the technologies and information flows to be used within and between the major components of generic digital audio-visual systems. Interoperability between these components and between individual sub-systems is assured through specification of tools and specification of dynamic systems behavior at defined reference points. A reference point can comprise one or more logical (non-physical) information-transfer interfaces, and one or more physical signal-transfer interfaces. A logical interface is defined by a set of information flows and associated protocol stacks. A physical interface is an external interface and is fully defined by its physical and electrical characteristics. Accessible reference points are used to determine and demonstrate compliance of a digital audio-visual subsystem with this international standard.

A summary of each part follows.

ISO/IEC 16500-1 (DAVIC 1.3.1a Part 2) defines the normative digital audio-visual systems technical framework. It provides a vocabulary and a Systems Reference Model, which identifies specific functional blocks and information flows, interfaces and reference points.

ISO/IEC 16500-2 (DAVIC 1.3.1a Part 12) defines system dynamic behavior and physical scenarios. It details the locations of the control functional entities along with the normative protocols needed to support the systems behavior. It is structured as a set of protocol walk-throughs, or “*Application Notes*”, that rehearse both the steady state and dynamic operation of the system at relevant reference points using specified protocols. Detailed dynamics are given for the following scenarios: video on demand, switched video broadcast, interactive broadcast, and internet access.

ISO/IEC 16500-3 (DAVIC 1.3.1a Part 14) provides the normative definition of DAVIC Technology Contours. These are strict sets of Applications, Functionalities and Technologies which allow compliance and conformance criteria to be easily specified and assessed. This part of ISO/IEC 16500 contains the full details of two contours. These are the Enhanced Digital Broadcast (EDB) and Interactive Digital Broadcast (IDB). ISO/IEC 16500-3 specifies required technologies and is a mandatory compliance document for contour implementations.

ISO/IEC 16500-4 (DAVIC 1.3.1a Part 8) defines the toolbox of technologies used for lower layer protocols and physical interfaces. The tools specified are those required to digitize signals and information in the Core Network and in the Access Network. Each tool is applicable at one or more of the reference points specified within the Delivery System. In addition a detailed specification is provided of the physical interfaces between the Network Interface Unit and the Set Top Unit and of the physical interfaces used to connect Set Top Boxes to various peripheral devices (digital video recorder, PC, printer). The physical Delivery System mechanisms included are copper pairs, coaxial cable, fiber, HFC, MMDS, LMDS, satellite and terrestrial broadcasting.

ISO/IEC 16500-5 (DAVIC 1.3.1a Part 7) defines the technologies used for high and mid-layer protocols for ISO/IEC 16500 digital audio-visual systems. In particular, this part defines the specific protocol stacks and requirements on protocols at specific interfaces for the content, control and management information flows.

ISO/IEC 16500-6 (DAVIC 1.3.1a Part 9) defines what the user will eventually see and hear and with what quality. It specifies the way in which monomedia and multimedia information types are coded and exchanged. This includes the definition of a virtual machine and a set of APIs to support interoperable exchange of program code. Interoperability of applications is achieved, without specifying the internal design of a set top unit, by a normative Reference Decoder Model which defines specific memory and behavior constraints for content decoding. Separate profiles are defined for different sets of multimedia components.

ISO/IEC 16500-7 (DAVIC 1.3.1a Part 10) defines the interfaces and the security tools required for an ISO/IEC 16500 system implementing security profiles. These tools include security protocols which operate across one or both of the defined conditional access interfaces CA0 and CA1. The interface CA0 is to all security and conditional access functions, including the high speed descrambling functions. The interface CA1 is to a tamper resistant device used for low speed cryptographic processing. This cryptographic processing function is implemented in a smart card.

ISO/IEC 16500-8 (DAVIC 1.3.1a Part 6) specifies the information model used for managing ISO/IEC 16500 systems. In particular, this part defines the managed object classes and their associated characteristics for managing the access network and service-related data in the Delivery System. Where these definitions are taken from existing standards, full reference to the required standards is provided. Otherwise a full description is integrated in the text of this part. Usage-related information model is defined in ISO/IEC 16500-9.

ISO/IEC 16500-9 (DAVIC 1.3.1a Part 11) specifies the interface requirements and defines the formats for the collection of usage data used for billing, and other business-related operations such as customer profile maintenance. It also specifies the protocols for the transfer of Usage Information into and out of the ISO/IEC 16500 digital audio-visual system. In summary, flows of audio, video and audio-visual works are monitored at defined usage data collection elements (e.g., servers, elements of the Delivery System, set-top boxes). Information concerning these flows is then collected, processed and passed to external systems such as billing or a rights administration society via a standardised usage data transfer interface.

Additional Information

ISO/IEC TR 16501 is an accompanying Technical Report. Further architectural and conformance information is provided in other non-normative parts of DAVIC 1.3.1a (1999). A summary of these documents is included here for information.

ISO/IEC TR 16501 (DAVIC 1.3.1a Part 1) provides a detailed listing of the functionalities required by users and providers of digital audio-visual applications and systems. It introduces the concept of a contour and defines the IDB (Interactive Digital Broadcast) and EDB (Enhanced Digital Broadcast) functionality requirements which are used to define the normative contour technology toolsets provided in ISO/IEC 16500-3.

DAVIC 1.3.1a Parts 3, 4 and 5 are DAVIC technical reports. They provide additional architectural and other information for the server, the delivery-system, and the Service Consumer systems respectively. Part 3 defines how to load an application, once created, onto a server and gives information and guidance on the protocols transmitted from the set-top user to the server, and those used to control the set-up and execution of a selected application. Part 4 provides an overview of Delivery Systems and describes instances of specific DAVIC networked service architectures. These include physical and wireless networks. Non-networked delivery (e.g., local storage physical media like discs, tapes and CD-ROMs) are not specified. Part 5 provides a Service Consumer systems architecture and a description of the DAVIC Set Top reference points defined elsewhere in the normative parts of the specification.

DAVIC 1.3.1a Part 13 is a DAVIC technical report, which provides guidelines on how to validate the systems, technology tools and protocols through conformance and / or interoperability testing.

.....

Information technology — Generic digital audio-visual systems — Part 6: Information representation

1. Scope

This part of ISO/IEC 16500 takes a practical approach to the specification of Information Representation. Just the information types that cannot be dispensed with in producing the set of DAVIC applications (viz. broadcast, movies on demand, home shopping, etc.) are specified. The approach taken in this part of ISO/IEC 16500 starts by defining the various monomedia information types. They include character, text, fonts, service information, audio, video, and graphics. Consistent with DAVIC principles, one tool is selected for the encoding of each information type. Multimedia components comprise one or more monomedia components. This part of ISO/IEC 16500 defines the way in which multimedia information is coded and exchanged. This includes the definition of a virtual machine and a set of APIs to support interoperable exchange of program code. Finally, this part of ISO/IEC 16500 defines a Reference Decoder Model for contents decoding which provides constraints on content. The major problem addressed by the model is to ensure interoperability of applications by specifying memory and behaviour constraints for contents decoding by a hypothetical STU, without specifying the internal design of an STU. An application built according to the reference decoder model will be an "ISO/IEC 16500 conforming application" and will successfully execute on a STU that is compliant to ISO/IEC 16500.

For each monomedia and multimedia component the coding format is specified, as well as applicable constraints for coding of the components. Three types of monomedia components are distinguished. Monomedia components which are included within other monomedia components, such as characters within text, are of type implied. Non-implied monomedia components that do not require synchronization with a time base at play back, are of type stand-alone. Finally, non-implied monomedia components of which the presentation may require synchronization with a time base are of type stream. This part of ISO/IEC 16500 defines which type each DAVIC defined monomedia component may take, and specifies that the coded representation of monomedia components of type stream are packetized in PES packets (for definition of PES packets refer to ISO/IEC 13818-1). PES packets permit (1) to include time stamps to support mutual synchronisation of multiple monomedia components in reference to a common time base and (2) to define timing and buffer behaviour in a common reference model for contents decoding. While there are various ways to deliver the monomedia and multimedia components to the STU, This part of ISO/IEC 16500 defines how the components are carried in an MPEG-2 Transport Stream.

DAVIC specifies a number of different profiles. In a specific profile there may be support of a subset of the monomedia components. Each STU that complies to a specific profile of DAVIC shall be capable of decoding and presenting each monomedia and multimedia component permitted within that profile.

This part of ISO/IEC 16500 also specifies methods for packaging of contents and metadata. The way in which content is packaged for delivery is independent of the way in which content data is delivered to the SPS (it may be delivered to a Service Provider either on physical media or over a transmission system). All programming content is represented in the DAVIC system as multimedia components. Multimedia components comprise one or more monomedia components coupled with the logical relationships between the monomedia components. The multimedia components will be created by content providers for input to the servers.

2. Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 16500. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of ISO/IEC 16500 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau (TSB) maintains a list of currently valid ITU-T Recommendations.

2.1. ISO, ISO/IEC and ITU Normative References

1. ISO 639, *Codes for the representation of names of languages*.
2. ISO 3166, *Codes for the representation of names of countries*.
3. ISO/IEC 8859-1:1987, *Information technology - 8-bit single-byte coded graphic character sets - Part 1: Latin alphabet No. 1*.
4. ISO/IEC 10646-1, *Information technology - Universal Multiple-Octet Coded Character Set (UCS) - Part 1: Architecture and Basic Multilingual Plane* (also known as Unicode).
5. ISO/IEC 11172-2:1993, *Information technology—Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s—Part 2: Video* (Note: known as MPEG-1 Video).
6. ISO/IEC 11172-3:1993, *Information technology—Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s—Part 3: Audio* (Note: known as MPEG-1 Audio).
7. ISO/IEC 13522-5:1997, *Information technology—Coding of multimedia and hypermedia information—Part 5: Support for base-level interactive applications* (Note: known as MHEG-5).
8. ISO/IEC 13522-6, *Information technology - Coding of multimedia and hypermedia information - Part 6: Support for enhanced interactive applications*.
9. ISO/IEC 13818-3:1998, *Information technology – Generic coding of moving pictures and associated audio information – Part 3: Audio* (Note: known as MPEG-2 Audio).
10. ISO/IEC 13818-6, *Information technology—Generic coding of moving pictures and associated audio information—Part 6: Extensions for DSM-CC*.
11. ITU-T (CCITT) Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1)* | ISO/IEC 8824: 1990, *Information Technology—Open Systems Interconnection—Specification of Abstract Syntax Notation One (ASN.1)*.
12. ITU-T (CCITT) Recommendation X.209 (1988) *Specification of Basic Encoding rules for abstract syntax notation one (ASN.1)* | ISO/IEC 8825: 1990, *Information technology—Open Systems Interconnection—ASN.1 encoding rules—Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*.
13. ITU-T Recommendation H.222.0 (1995) | ISO/IEC 13818-1: 1996, *Information technology—Generic coding of moving pictures and associated audio information: Systems* (Note: known as MPEG-2 Systems).
ISO/IEC 13818-1/Amendment 1: 1997, *Registration procedure for “copyright identifier”*.
ISO/IEC 13818-1/Amendment 2: 1997, *Registration procedure for “format identifier”*.
ISO/IEC 13818-1/Amendment 3: 1998, *Private data identifier*.
14. ITU-T Recommendation H.262 | ISO/IEC 13818-2, *Information technology—Generic coding of moving pictures and associated audio information: Video* (Note: known as MPEG-2 Video).
ISO/IEC 13818-2 /Amendment 1: *Registration procedure for “copyright identifier”*.

2.2. Other Normative References

2.2.1 ATSC (Advanced Television Systems Committee)

1. ATSC A/52: *Digital audio compression standard (AC-3)*.
available at <ftp://ftp.atsc.org/pub/Standards/A52>.
2. ATSC A/53: *Digital television standard for HDTV transmission*.
available at <ftp://ftp.atsc.org/pub/Standards/A53>.

2.2.2 ANSI (American National Standards Institute)

1. ANSI SMPTE 274M-1995, *Television - 1920x1080 Scanning and interface*.
2. ANSI SMPTE 296M-1997, *Television - 1280x720 Scanning, analog and digital representation and analog interface*.

2.2.3 Apple Corporation Inc.

1. *AIFF-C Audio Interchange File Format, version C, allowing for Compression*.
2. *Bento Specification, Revision 1.0d5, July 15, 1993*.

2.2.4 ETSI (European Telecommunications Standards Institute)

1. ETR 162 (October 1995): *Digital broadcasting systems for television, sound and data services: Allocation of Service Information (SI) codes for Digital Broadcasting (DVB) systems*.
2. ETR 211: *Digital broadcasting systems for television, sound, and data services; Guidelines for the usage of Service Information (SI) in Digital Video Broadcasting (DVB) systems*.
3. ETS 300 468 (January 1997): *Specification for Service Information (SI) in DVB Systems Informative Annex C: Conversion Between Time and Date Conventions*.
4. ETS 300 472, *Digital broadcasting systems for television, sound, and data services; Specification for conveying ITU-R System B Teletext in Digital Video Broadcasting (DVB) bitstreams*.
5. ETS 300 743, *Digital Video Broadcasting (DVB), DVB subtitling*.
6. ETS 300 777-2, *Use of Digital Storage Media Command and Control (DSM-CC) for basic multimedia applications*.
7. ETSI DI / MTA-01074, *Multimedia Terminals and Applications, Application Programming Interface (API) for DAVIC Service Information*.

2.2.5 SCTE (Society of Cable Telecommunications Engineers, Inc)

1. SCTE DVS/026 - *Digital Video : Subtitling methods for Broadcast Cable*.

2.2.6 SMPTE (Society of Motion Picture and Television Engineers)

See ANSI (American National Standards Institute)

2.2.7 W3C

1. CSS-1, *Cascading Style Sheets, level 1*; by Håkon Wium Lie and Bert Bos, 17-December-96. available at <http://www.w3.org/TR/REC-CSS1-961217>.
2. HTML 3.2, *HyperText Mark-up Language reference specification*, by Dave Raggett, 14-Jan-1997. available at <http://www.w3.org/TR/REC-html32.html>.
3. PNG *Portable Network Graphics version 1*, 01-October-1996. available at <http://www.w3.org/TR/REC-png.html>.

3. Definitions

This clause defines new terms, and the intended meaning of certain common terms, used in this part of ISO/IEC 16500. Annex A of ISO/IEC 16500-1 defines additional terms and, in some cases, alternative interpretations that are appropriate in other contexts. For convenience, the normative definitions below are included in the annex.

- 3.1. access control:** Provides means to access services and protection against the unauthorized interception of the services.
- 3.2. anchor:** one of two ends of a hyperlink
- 3.3. application:** a set of objects that provides an environment for processing Application Service Layer information flows.
- 3.4. Application Programming Interface (API):** set of inter-layer service request and service response messages, message formats, and the rules for message exchange between hierarchical clients and servers. API messages may be executed locally by the server, or the server may rely on remote resources to provide a response to the client.
- 3.5. assets:** Things that a user sees or hears, e.g., bitmap, audio, text.
- 3.6. character:** an atom of textual information, for example a letter or a digit
- 3.7. conditional access:** A means of allowing system users to access only those services that are authorized to them.
- 3.8. Content Item:** A collection of content items / Content Item Elements that will form a complete application or a complete programme.
- 3.9. Content Item Element:** the smallest (and indivisible) content component.
- 3.10. Content package:** A set of content Item Elements and/or content items for transfer across an A10 interface between Content Provider and Service Provider Systems.
- 3.11. Content Provider:** one who owns or is licensed to sell content.
- 3.12. Control Word:** the secret key used for a scrambling algorithm.
- 3.13. Delivery System (DS):** The portion of the DAVIC System that enables the transfer of information between DS-users.
- 3.14. element:** a component of the hierarchical structure defined by a document type definition; it is identified in a document instance by descriptive markup, usually a start-tag and end-tag.
- 3.15. end-tag:** descriptive markup that identifies the end of an element
- 3.16. encryption:** a mathematical technique used to ensure the confidentiality of security management information.
- 3.17. Entitlement Control Message (ECM):** conditional access messages carrying an encrypted form of the control words or a means to recover the control words, together with access parameters, i.e., an identification of the service and of the conditions required for accessing this service.
- 3.18. Entitlement Management Message (EMM):** conditional access messages used to convey entitlements or keys to users, or to invalidate or delete entitlements or keys.
- 3.19. key management:** The generation, storage, distribution archiving, deletion, revocation, registration, and deregistration of cryptographic keys.
- 3.20. hyperlink:** a relationship between two anchors
- 3.21. joint stereo:** a coding option in MPEG-1 audio that exploits the redundancy between the left and right audio channels
- 3.22. logical interface:** an interface where the semantic, syntactic, and symbolic attributes of information flows is defined. Logical interfaces do not define the physical properties of signals used to represent the information. A logical interface can be an internal or external interface. It is defined by a set of information flows and associated protocol stacks.
- 3.23. monomedia component:** a collection of data representing a single type of audiovisual information.

- 3.24. monospace format:** a presentation format of characters in which each character utilizes a character matrix of the same size, independent of the width and height of the character.
- 3.25. multimedia component:** a collection of data comprising one or more multimedia components
- 3.26. navigation:** the process of reaching a service objective by means of making successive choices; the term may be applied to the selection of a service category, a service provider or an offer within a particular service.
- 3.27. protocol:** set of message formats (semantic, syntactic, and symbolic rules) and the rules for message exchange between peer layer entities (which messages are valid when).
- 3.28. real-time stream:** an MPEG-2 transport stream containing monomedia components of which the timing of the decoding and presentation in an STU is controlled by the characteristics of the stream during the delivery of the stream to the STU.
- 3.29. rendering:** the process in the STU to combine one or more monomedia components such as characters, text, and graphical objects into one presentation on a screen.
- 3.30. scrambling:** The process of making a signal unintelligible at the transmission point in order that it can only be received if an appropriate descrambling system is in place at the point of reception. Scrambling can be applied to audio, video or data signals
- 3.31. server:** any service providing system.
- 3.32. Service Information (SI):** Digital data describing the delivery system, content and scheduling/timing of MPEG-2 Transport Streams. It includes MPEG-2 PSI together with independently defined extensions.
- 3.33. Service Provider:** an entity that provides a service to a client.
- 3.34. session:** an interval during which a logical, mutually agreed correspondence between two objects exists for the transfer of related information. A session defines a relationship between the participating users in a service instance.
- 3.35. Set Top Box (STB):** a module that comprises both Set Top Unit (STU) and Network Interface Unit (NIU) functional elements. The STB may be either “integrated” or “modular”. An integrated STB is designed for connection to a single DAVIC A1 or equivalent interface. A modular STB may be equipped with a DAVIC A0 or equivalent interface to enable connection of a range of NIUs.
- 3.36. Set Top Unit (STU):** a module that contains the “network independent” functionalities of a Set Top Box (STB). The following functionalities are contained in a typical STU:- Processing & Memory Functions; MPEG2 Demux & AV Decoders; Graphics Display; Modulator Output for TV; Peripheral Interfaces.
- 3.37. start-tag:** descriptive markup that identifies the start of an element
- 3.38. tag:** markup that delimits an element
- 3.39. virtual machine (VM):** An abstract specification of a micro-processor and its behaviour

NOTE: A VM may be implemented on different hardware processors. A VM therefore implements the mechanism for all these processors to execute the same instruction set. It is also possible for a micro-processor to be designed so that its instruction set is identical to that of a VM. VM code can be used to make software portable. In the context of DAVIC, the VM is used to extend interoperability by allowing program code produced once to be delivered to and executed on any compliant STU.

4. Acronyms and abbreviations

This clause defines the acronyms and abbreviations used in this part of ISO/IEC 16500. Annex B of ISO/IEC 16500-1 defines acronyms and abbreviations used within ISO/IEC 16500.

AC-3	ATSC A52 Audio
AIFF	Audio Interchange File Format
API	Application Programming Interface
ASCII	American Standard Code for Information Exchange
ASN.1	Abstract Syntax Notation 1
ATSC	Advanced Television Systems Committee
BNF	Backus-Naur Form
bslbf	Bit string left bit first
CA	Conditional Access
CIE	Content Item Element
CI	Content Item
CLUT	Color LookUp Table
CMSL	Content Metadata Specification Language
CORBA	Common Object Request Broker Architecture
CPS	Content Provider System
CRC	Cyclic Redundancy Check
CW	Control Word
DIS	Draft International Standard
DSM-CC	Digital Storage Media - Command and Control
DSM-CC U-N	DSM-CC User to Network
DSM-CC U-U	DSM-CC User to User
DTS	Decoding Time Stamp
DVB	Digital Video Broadcasting
DVB-SI	DVB - Service Information
ECM	Entitlement Control Message
EMM	Entitlement Management Message
EPG	Electronic Program Guide
ESC	End Service Consumer
ESCS	End-Service Consumer System
ESP	End Service Provider
ESPS	End-Service Provider System
ETS	European Telecommunications Standard
ETSI	European Telecommunications Standards Institute
ETR	European Telecommunications Recommendation
FIFO	First In First Out
fpvsbf	Floating point value sign bit first

HDTV	High Definition Television
HTML	HyperText Markup Language
ID	Identification
IDL	Interface Definition Language
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPR	Intellectual Property Rights
ISO	International Organization for Standardization
ITU	International Telecommunications Union
Mbps	Megabits per second
LSB	Least Significant Bit
MHEG	Multimedia and Hypermedia information coding Experts Group
MPEG	Moving Picture Experts Group
MPEG-TS	MPEG-2 Transport Stream
MSB	Most Significant Bit
MUX	Multiplex
NIU	Network Interface Unit
NPT	Normal Play Time
NTSC	National Television Systems Committee
OMG	Object Management Group
OMG-UNO	Object Management Group - Universal Networked Object
OS	Operating System
OSI	Open Systems Interconnection (Reference Model)
PC	Personal Computer
PCR	Program Clock Reference
PES	Packetized Elementary Stream
PID	Packet Identifier, or Program Identification
PMT	Program Map Table
PN	Program Number (MPEG-2)
PNG	Portable Network Graphics (specified by W3C)
PSI	Program Specific Information (MPEG-2)
PTS	Presentation Time Stamp
QoS	Quality of Service
RDM	Reference Decoder Model
RGB	Red Green Blue
RPC	Remote Procedure Call
RTE	RunTime Engine
SCS	Service Consumer System
SCTE	Society of Cable Telecommunications Engineers, Inc

ISO/IEC 16500-6:1999(E)

SGML	Standard Generalized Markup Language
SI	Service Information
SPS	Service Provider System
SPV	Service Provider
SRC	Service Related Control
STU	Set-Top Unit
TBD	To be defined
TCP	Transmission Control Protocol
tcimbsf	Two's complement integer, msb (sign) bit first
TS	Transport Stream
T-STD	Transport System Target Decoder
TV	Television
UDP	User Datagram Protocol
uimbsf	Unsigned integer most significant bit first
UNO	Universal Networked Object
UTC	Universal Coordinated Time
VCR	Video Cassette Recorder
VM	Virtual Machine
W3C	World Wide Web Consortium

5. Conventions

The style of this Part of ISO/IEC 16500 follows the *Guide for ITU-T and ISO/IEC JTC 1 cooperation. Appendix II: Rules for presentation of ITU-T | ISO/IEC common text (March 1993)*.

6. Monomedia Components

The basic monomedia components in ISO/IEC 16500 include the following elements with their associated coding formats. Various options within the coding standards and other file and packaging formats are allowed as detailed in the following clauses. If the content is encrypted the following descriptions pertain to the decrypted data.

Table 6-1. — Coding options of monomedia components

<i>Monomedia component</i>	<i>Coding Options</i>
Characters	ISO 10646-1
Text	HTML 3.2
Outline Fonts	Defined in this specification
Language Information	ISO 639, part 2
Service Information	ETS 300-468
Telephone Numbers	ETS 300 468
Compressed Audio	MPEG-1 Audio, AC-3 Audio
Scaleable Audio	MPEG-2 BC Audio
Linear Audio	AIFF-C
Compressed Video	MPEG-2 Video (note that this includes constrained MPEG-1 video)
Still Picture	MPEG-2 Intra Frame and MPEG-2 Systems
Compressed graphics	ETS 300 743
Compressed Character Data	SCTE DVS/026
Network graphics	PNG (Portable Network Graphics)
<i>Note : Network Graphics replaces the monomedia component 'uncompressed graphics' from DAVIC 1.0 up to DAVIC 1.2.</i>	

6.1. Character Information

In ISO/IEC 16500 coding of characters is based on Unicode, ISO 10646-1, to support multilingual text. Rather than extending the mandatory Latin character set in each DAVIC compliant STU, as defined in DAVIC 1.0, ISO/IEC 16500 provides a mechanism to download character images that are not residently available in a STU. To this effect character images can be downloaded. ISO/IEC 16500 does not specify a coding format for character images. Future versions of DAVIC specifications are expected to adopt the result of the ongoing work in the World Wide Web Consortium on an outline format for character coding.

Character images for the Latin-1 characters as defined in ISO 8859-1 are mandatorily supported in ISO/IEC 16500 STUs. The STU shall at least be capable to display 24 lines per screen with 40 monospaced characters per line with respect to ITU-R 601.

6.2. Text Information

The following requirements on coding of text information have been identified :

- the syntax provides a tag for hyperlinks consistent with MHEG-5 Hypertext class anchor
- the syntax provides tags for rendering control: new line (left, right, center, justify), italics, underline, bold, emphasis, strong emphasis, font selection, colour selection, size selection
- the syntax and semantics are suitable for verification in the Reference Decoder Model
- the syntax is extensible in order to accommodate the expected evolution of text coding requirements
- the syntax is efficient with respect to coding and parsing
- the syntax allows for error resilience
- the character encoding encompasses the full Latin character set (ISO 8859-1)
- the character encoding is multilingual and allow expansion (ISO 10646-1)
- the syntax allows control over the starting corner, text flow direction and rotation
- the syntax provides additional tags for sub-script, super-script, embedding bitmap objects and language specifiers

ISO/IEC 16500 is based on the HTML 3.2 Reference Specification (W3C Recommendation 14-Jan-1997), including the support of RFC 2070 Internationalization of the Hypertext Markup Language, as well as the support of Cascading Style Sheets (CSS). The specification supports the marking of fragments of text as being in a certain language or writing direction (left to right or right to left), so that proper formatting can be applied to them.

Language specifiers are included in the HTML 3.2 specification. The language is identified with attributes that are formed according to RFC 1766, which in turn is based on ISO 639 and ISO 3166. Also superscripts and subscripts are supported in HTML 3.2, as well as alignment attributes on most elements and selection of colour and size using the FONT tag. Increased presentational control and greater separation of structural and presentation information is provided by (Cascading) Style Sheets. Style Sheets also allow targeting for different types of display, for example, there might be one stylesheet for typical PC/workstation display, another for display on an ISO/IEC 16500 compliant STU, another for handheld mobile devices, and a fourth for a laser printer.

ISO/IEC 16500 text is coded as either a stand-alone <BODY> element or a full <HTML> element, both as defined in HTML 3.2. ISO 10646-1 as specified in HTML 3.2 is used both as the document character set as well as the character encoding scheme.

6.2.1 Decoding requirements on HTML support

DAVIC compliant decoders shall act upon all HTML 3.2 tags, except the following elements which may be optionally acted upon:

1. all HTML 3.2 deprecated elements; these include: LISTING, XMP.
2. all table-based elements; these include: TABLE, TD, TH, TR.
3. all form-based elements; these include: ISINDEX, FORM, INPUT, SELECT, TEXTAREA.
4. all applet or script-based elements (such as JavaScript), which should be parsed, but may not be displayed (except any alternate text ALT); these include: APPLET, SCRIPT.
5. all client-side image-map elements; these include MAP.
6. all URL modification tags; these include: BASE.

Tags specified in RFC2070 shall also be parsed, and it is recommended that these should be acted upon (in addition to the HTML 3.2 reference specification); these include: LANG, DIR, ALIGN, Q, SPAN plus additional character entities.

Tags specified in Cascading Style Sheets, level 1 (W3C Recommendation 17 Dec 1996) shall also be parsed, and it is recommended that these should be acted upon (in addition to the HTML 3.2 reference specification); these include: STYLE, STYLESHEET, CLASS, ID.

Additional tags shall be parsed and may either be discarded if no alternative text (ALT) is available, or display the alternative text if it is available, or may act upon the tag if it is known.

6.2.2 HTML encoding

ISO/IEC 16500 specifies that HTML 3.2 documents are encoded using a variant of UTF-8 (Amendment 1 to ISO/IEC 10646-1) so that Unicode characters can be supported directly within HyperText objects. This variant of UTF-8 specifies the following :

- the null byte, (byte) 0, is encoded using the two-byte format, rather than the one-byte format — this ensures that there are no embedded null's in a UTF-8 string.
- Only one-, two- and three-byte formats are used; longer variants are not recognised.

This variant of UTF-8 encoding is also used in the Sun Java virtual machine.

The UTF-8 encoding ensures that the ASCII subset of Unicode is transmitted in a single byte, whilst other Unicode characters are transmitted in two or three bytes. This approach offers a significant advantage when most characters lie within the ASCII portion of Unicode.

See Clause 6.2.4 for an informative list of DAVIC supported HTML tags.

6.2.3 HTML mapping to MHEG elements

Various elements within an HTML document require a mapping to MHEG elements; these include: font-based elements and colour-based elements.

- Fonts: HTML supports a relative sizing model for fonts, ranging from 1–7. A mapping between HTML font sizes and MHEG fonts is to be provided.
- Colours: HTML specifies colours as a 24-bit RGB hex triple, e.g. #FF0000. Such colours may be mapped to MHEG absolute colours by considering the most significant 4 bits of each of the HTML RGB components to be mapped directly to each of the RGB components of an MHEG RGB α 16 absolute colour. By default, the α component of the MHEG RGB α 16 absolute colour may be considered to be opaque, i.e. 15.

6.2.4 Informative list of supported HTML Tags

In addition to standard SGML entities, such as document type and comments, the support of tags from HTML3.2 by ISO/IEC 16500 includes :

Table 6-2. — Examples of HTML 3.2 tags supported by ISO/IEC 16500

<i>HTML Tag</i>	<i>Meaning</i>
A	Anchor: create a hyperlink
ADDRESS	Address: the enclosed text is an address
B	Bold: format the enclosed text as bold
BASEFONT	Base font: Specify the font size for subsequent text
BIG	Big: format the enclosed text using a bigger font
BLOCKQUOTE	Block quotation: the enclosed text is a block quotation
BODY	Body: delimit the body of the document text
BR	Break: break the current text flow
CAPTION	Caption: specify a caption for a table
CENTER	Centre: center the enclosed text
CITE	Citation: the enclosed text is a citation
CODE	Code: the enclosed text is a code sample
DD	Definition: define the definition portion of an element in a definition list
DFN	Definition: the enclosed text is a definition of a term

DIR	Directory: create a “directory” list
DIV	Division: create a division within a document
DL	Definition list: create a list of definitions and their terms
DT	Definition term: define the term portion of an element in a definition list
EM	Emphasis: the enclosed text should be emphasised
FONT	Font: set the size / colour of the font
H(1-6)	Heading: the enclosed text is a level 1–6 heading
HEAD	Head: delimit the head of a document
HR	Horizontal rule: draw a horizontal rule
HTML	HTML: delimit the extent of the whole HTML document
I	Italic: format the enclosed text as italic
IMG	Image: insert an image into the current text flow
KBD	Keyboard: the enclosed text is keyboard-like input
LI	List item: an item in an ordered or unordered list
LINK	Link: define a link between the current document and another document specified in the head — this can be used for CSS
MENU	Menu: define a menu list
META	Meta information: provide additional information about a document
OL	Ordered list: delimit an ordered list
P	Paragraph: define the start (and end) of a paragraph
PLAINTEXT	Plain text: format the remainder of the document as pre-formatted plain text
PRE	Pre-formatted text: format the enclosed text in its original pre-formatted version
SAMP	Sample: the enclosed text is a sample
SMALL	Small: format the enclosed text using a smaller font
STRIKE	Struck through: format the enclosed text as struck through with a horizontal line
STRONG	Strong: strongly emphasize the enclosed text
STYLE	Style: used for cascading style sheets (CSS)
SUB	Subscript: format the enclosed text as a subscript
SUP	Superscript: format the enclosed text as a superscript
TITLE	Title: specify the title of an HTML document
TT	Typewriter text: format the enclosed text in a typewrite (mono-spaced) font
U	Underlined text: format the enclosed text as underlined
UL	Unordered list: delimit an unordered list
VAR	Variable: the enclosed text is a variable

6.3. Outline Font Format

6.3.1 Requirements

General requirements for a font format to support DAVIC are listed below. The DAVIC font format may be implemented in a static or dynamic process. For example, a set of fonts may exist in ROM of a set-top unit, and are rendered on demand by applications executing within the set-top.

The DAVIC font format can be used by content providers in several ways. First, the content provider may render or rasterize the content and produce a MPEG, or PNG data that is then broadcast down to the set-top unit. Secondly for down loaded applications that require a variety of fonts, servers can dynamically create small DAVIC font objects that contain only the character data required to image the content, and they are downloaded to the set-top. Once the application has no further need, the small font objects can be deleted.

Specific details about dynamic generation of this format and rendering this format are outside the scope and purpose of this document. Those details are available from DAVIC member, Bitstream Inc. The DAVIC font format is intended to be used by members of DAVIC or developers of DAVIC technology. The requirements for the font format which extends the current bitmap DAVIC technology is as follows.

- **Platform Independent font format:** DAVIC requires a byte stream format that can be transferred to any platform within a DAVIC network. Network nodes may be running a variety of operating systems on a variety of processors so it is necessary the format is reduced to a byte stream of known format. Implementation of the format across processors and operating systems must not favor or require a particular processor or operating system.
- **Scaleable outline font format:** The request for a scaleable cubic outline format is required to resolve the differing resolutions or aspect ratios of the target devices. The format must be capable of supporting multiple fonts (logical fonts) with a variable number of characters. The format must be capable of supporting any font style or treatment for the purpose of imaging within a DAVIC device. In the event that static output devices such as printers are connected to a DAVIC device, the format must be suitably rendered in the resolution of the output device.
- **Character Encoding:** Digital content that references characters within DAVIC environments are based on UNICODE and the font format must fully support UNICODE.
- **Default Character Set for Resident Font(s) within DAVIC devices:** DAVIC devices may contain the ISO-Latin 8859-1 character set in a Bitmap or Outline format. It is desirable to extend the current text and UNICODE character access mechanism to support outline font technology. Augmenting the character set dynamically is highly desirable so that multi-lingual digital content can be imaged and displayed. Manufactures may implement any extra character sets to support target markets within their devices statically or by dynamic downloading. The default ISO-Latin 8859-1 character set is always assumed to be within the DAVIC device.
- **Dynamic Merging of character sets:** Character access via UNICODE implies multi-lingual support and can be accomplished by downloading either bitmap or outline font fragments. The selected font format must be easy to combine font fragments so that font fragments appear as a single font with a single unified character set.
- **Memory, Font Format Size and Compression:** Size of the outline font format must not burden the DAVIC devices with unnecessary data (ROM or RAM) or require specialized decompression logic. The selected format must be dense or tightly coded, and not require the commercial licensing of data compression logic.
- **Floating Point Processing:** DAVIC compliant network nodes, may not contain a FPP so the selected font format must not require floating point processing.
- **Font Effects and Rendering Considerations:** The selected scaleable cubic outline font format must be able to be rendered as follows:
 - Scaled independently in X and Y to accommodate asymmetric resolutions. Scaling is also required to produce bitmaps for devices which have higher resolutions, such as printers which may be connected to the set-top boxes in the future.
 - Scale to any arbitrary device context (in case static printouts are desired in the future)
 - Rotation of characters or text to any angle. This is important for down-loadable applications that wish to provide these effects. Orthogonal rotations as well as any increment in 10ths of a degree are required.

- Slanting or Obliqueing or shearing by using a transform matrix rather than requiring an Italic font must be optionally callable from any application. Usually a shear of 12 degrees is used to obtain an Italic effect. Shearing must be specified to any angle in 10ths of degree increments.
- Anti-aliased to arbitrary levels
- Filtered, gamma corrected, colored or blended
- Be made transparent or opaque
- **Format and technology availability:** DAVIC requires that the selected format be available to DAVIC members at no charge. DAVIC also requires commercially available technology to dynamically generate the format and render the format.
- **Industry synergy:** DAVIC devices may be connected to the Internet in the future, and therefore are expected to be capable of viewing Internet content. Synergy with the Internet industry and other standards is desirable.

6.3.2 Font Format Specification

The DAVIC specification of the format for outline fonts is contained in Annex A of this specification.

6.3.3 Font rendering

To ensure consistent text composition of rendering engines, the use of the following parameters in the rendering process are recommended :

- A default anti-alias bit depth of four.
- A default sub-pixel character placement of one quarter pixel accuracy.
- The use of a method for calculation of character string and algorithm for rounding to the nearest sub-pixel boundary with the same result as from the method given in the following example :

Consider the word "DAVIC" to be written at a specified position (x0, y0) in device coordinates. Device coordinates should be stored in fractional pixels. DAVIC recommends to use a 16.16 representation of each coordinate in a 32-bit word. In other words, to describe x and y coordinates each in units of 1/65536 pixels.

The first character 'D' is rendered with its character origin at the nearest sub-pixel boundary relative to (x0, y0). With a sub-pixel accuracy of the recommended 1/4 pixel, the rounded character origin (x, y) is calculated by:

$$x = (x0 + 0x00002000) \& 0xFFFFC000;$$

$$y = (y0 + 0x00002000) \& 0xFFFFC000;$$

The set width of the 'D' is stored in the PFR in metricsResolution units. This is transformed into device coordinates by multiplying it by the current transformation matrix (CTM). The CTM is the product of the outputMatrix and the fontMatrix. The fontMatrix defines the size of the font in pixels per user unit. The outputMatrix defines the transformation from user units to pixels. Both matrices are expressed in 16.16 units. Multiplication of one 16.16 unit by another preserves the full precision except, of course, for the ultimate rounding to the nearest 1/65536 pixel.

The formal expression for the transformation of the setWidth is:

$$\begin{vmatrix} \text{setWidth.x} \\ \text{setWidth.y} \end{vmatrix} = \begin{vmatrix} \text{outputMatrix} \\ \text{fontMatrix} \end{vmatrix} * \begin{vmatrix} \text{charWidth} \\ 0 \end{vmatrix}$$

where the value of charWidth is equal to the PFR-defined set width of the 'D' in metrics resolution units, multiplied by 65536 and then divided by its metrics resolution.

The current position is then updated by adding (setWidth.x, setWidth.y) to it. Note that the rounding required to render the 'D' is not included in the update to the current position. This avoids any accumulation of rounding errors across a line and ensures that each character is positioned with the best possible accuracy.

The updated current position is then used to position the letter 'A'. Like the first character, it is rendered at the nearest sub-pixel boundary to its current position.

If pair kerning is enabled, the character pair 'D' and 'A' are looked up in the list of kerning pairs for the currently selected font. If a matching kerning pair is found, the adjustment is transformed into device coordinates by multiplying it with the CTM and the result added to the current position prior to rendering the letter 'A'.

If track kerning is enabled, the track kerning adjustment associated with the currently selected font and the current nominal point size is also transformed into device coordinates by multiplying it with the CTM and the result added to the current position prior to rendering the letter 'A'.

If both pair kerning and track kerning apply to an inter-character space, the track and pair kerning adjustments are added together before transforming the composite adjustment into device coordinates. This improves performance by saving a transformation operation. It also improves precision by eliminating a rounding error.

The process of applying intercharacter kerning adjustments and rendering characters continues until the 'C' has been rendered. No kerning adjustment is applied after the last character. If additional characters are appended to the string, the appropriate kerning adjustment should be applied prior to the first character. This is unlike the first character in a new string.

6.4. Language Information

In ISO/IEC 16500 coding of language information shall be based on ISO 639-2. If an elementary stream in a program represents a specific language, then the ISO 639 descriptor shall be used in the Program Map Table to identify that language. In the case of a compressed audio stream with two independent channels, each representing a different language, both languages shall be identified by first including the ISO 639 descriptor for channel 1, immediately followed by the same descriptor for channel 2.

6.5. Service Information

This clause describes the service information (SI) data which forms a part of compliant bitstreams, in order to provide the user with information to assist in the selection of services and/or events within the bitstream. In addition it provides physical transmission information to enable the set-top unit to access a service.

In ISO/IEC 16500 the Service Information format shall comply to the ETS 300 468 specification.

In addition, the use of service information in ISO/IEC 16500 shall adhere to the SI implementation guidelines, specified in ETR 211.

6.6. Telephone Numbers

In ISO/IEC 16500 the coding of telephone numbers shall be according to the telephone number segmentation format specified for the ETSI SI telephony descriptor in ETSI ETS 300-468. This permits a common telephone number segmentation to be referenced by application objects in the STU. This would allow, for example the telephone number provided as part of a televote service to be automatically modified according to the users wishes before the number was dialed.

6.7. Compressed Audio

ISO/IEC 16500 compressed audio includes two methods of compression that can be used to realise two sets of functionalities. These methods are described in the following subsections.

6.7.1 Compressed audio coding using MPEG-1 Audio

MPEG-1 Audio (ISO/IEC 11172-3) shall be coded with the following constraints:

- Compressed audio shall use MPEG-1 Layers I and II coding (layer = '11' or '10')
- Compressed audio shall be in single channel, dual channel, joint stereo or stereo.
- Compressed audio shall use a sampling rate of 32 kHz, 44.1 kHz, or 48 kHz.
- For Layer I, compressed audio may have each bit rate permitted in the range between 32 and 448 kbits/sec (32, 64, 96, 128, 160, 192, 224, 256, 288, 320, 352, 384, 416 and 448), with the exception of the free format bit rate.
- For Layer II, compressed audio may have each bit rate permitted in the range between 32 and 384 kbits/sec (32, 48, 56, 64, 80, 96, 112, 128, 160, 192, 224, 256, 320 and 384), with the exception of the free format bit rate.
- Compressed audio shall not apply the free format bit rate (bitrate_index = '0000' is forbidden).
- Compressed audio shall have no emphasis (emphasis = '00').
- Compressed audio shall include the parity word check (crc_check) in each audio frame.

Note that the bit rate may be switched in compressed audio streams on audio frame boundaries.

6.7.2 Compressed audio coding using ATSC A/52 Audio

ATSC A/52 Audio, including support for multichannel surround sound, shall be coded with the following constraints:

- ATSC A/52 compressed audio shall be constrained to a maximum bit rate of 448 kb/s.

Note that the bit rate may be switched in compressed audio streams on audio frame boundaries.

Further information on coding and carriage of ATSC A/52 Audio in ATSC systems is described in Annex L.

6.8. Scaleable Audio

This subclause describes the information representation of audio content to be used in environments where scalability is needed (i.e. a level of performance beyond the capability of the compressed audio tools in subclause 6.7 ‘Compressed Audio’), such as delivery of audio over bandwidth limited media or over media with contention (e.g. the internet or intranets with packet loss and variable delay, or over mobile channels), and non-guaranteed rates.

For scaleable audio, ISO/IEC 13818-3 (Second Edition, 1997) shall be used with the following constraints:

- Scaleable audio shall use MPEG-2 Layer II coding (layer = ‘10’)
- Scaleable audio shall not apply the free format bit rate (bitrate_index = ‘0000’ is forbidden).
- Scaleable audio shall have no emphasis (emphasis = ‘00’).
- Scaleable audio shall include the parity word check (crc_check) in each audio base_frame.
- Scaleable audio shall not apply prediction (mc_prediction_on = ‘0’)
- Scaleable audio shall not apply multilingual (no_of_multi_lingual_ch = ‘000’)
- Scaleable audio decoders shall support a switch in base bit rate and extension bit rate on audio frame boundaries.

6.9. Linear Audio

Linear audio shall be coded using AIFF-C. The specification of AIFF-C is contained in Annex B of this specification. AIFF-C describes a very versatile audio coding format. In the AIFF-C specification, the audio sample is broken into ‘chunks’, four of which must be present in all audio samples: the Form chunk, the Format Version chunk, the Extended Common chunk, and the Sound Data chunk. The Form chunk must be present at the beginning of the audio sample. All other chunks including user defined chunks, referred to herein as Private chunks, are allowed to exist in any order after the Form chunk. Multiple instantiations of all chunks except the Format Version chunk and the Sound Data chunk are allowed within a sample.

Note : Support of Linear Audio as a real-time stream in future versions of the DAVIC specifications may require concatenation of objects containing linear audio with a play back duration of up to 0.7 second each.

6.10. Compressed Video

Video information shall be coded using MPEG-1 Video (ISO/IEC 11172-2) or MPEG-2 Video (ISO/IEC 13818-2). For MPEG-1 Video, the data must follow the constrained parameter set, i.e. in the MPEG-1 Video data the `constrained_parameter_flag` shall be set to '1'. For MPEG-2 Video, the data shall conform to Main Profile syntax. Next to coding of video up to the standard video resolution as specified in ITU-R 601, DAVIC specifies coding of video at higher resolutions. Informative Annex L provides examples of video input formats that may be used.

Constraints for the coding of video with a resolution up to ITU-R 601 are specified in Clause 6.10.1. Clause 6.10.2 specifies constraints for the coding of video with resolutions higher than ITU-R 601.

6.10.1 Coding constraints for video with a resolution up to ITU-R 601

6.10.1.1 Main Profile

For the coding of video with a resolution up to ITU-R 601, the constraints defined by MPEG for the Main Profile at Main Level (MP@ML) shall apply.

6.10.1.2 Aspect Ratio of 4:3 and Pan Vectors

Compressed video shall have a source aspect ratio of 1:1, 4:3 or 16:9. Specifically, the `aspect_ratio_information` shall have the value '0001', '0010' or '0011'. It is recommended that pan vectors for a 4:3 window are included in the video bitstream when the source aspect ratio is 16:9. The vertical component of each pan vector shall be zero. If pan vectors are included, then the `sequence_display_extension` shall be present in the bitstream and the `aspect_ratio_information` shall be set to '0010' (4:3 display). The `display_vertical_size` shall be equal to the `vertical_size_value`. The `display_horizontal_size` shall contain the resolution of the target 4:3 display. The value of the `display_horizontal_size` field may be calculated by the following formula :

$$\text{display_horizontal_size} = (3/4) * (\text{horizontal_size_value})$$

The table below gives some typical examples.

Table 6-3. — Some typical values for `display_horizontal_size` for 4:3 window in 16:9 picture

<i>horizontal_size_value</i>	<i>display_horizontal_size</i>
720	540
704	528
544	408
528	396
480	360
352	264

6.10.1.3 Full Screen

If no `sequence_display_extension` is present, then the coded picture size shall be any of the values from Table 6-4, with the required upsampling ratios for full screen display on 4:3 and 16:9 monitors with 720 pixels per line. Note that DAVIC does not constrain the horizontal display resolution of the STU to 720 pixels; the applied display resolution is fully at the discretion of the implementation.

Table 6-4. — Horizontal Upsampling Ratios for Full Screen Picture Sizes

coded_picture_size ⁶⁾	source_aspect_ratio	horizontal upsampling ratios to 720 active pixels per line	
		4:3 monitor	16:9 monitor
720 x 576	4:3	x 1	x (3/4) ¹⁾
	16:9	x (4/3) ⁷⁾	x 1
704 x 576	4:3	x 1 ³⁾	x (3/4) ¹⁾
	16:9	x (4/3) ^{3) 7)}	x 1
544 x 576	4:3	x (4/3)	x 1 ¹⁾
	16:9	x (16/9) ⁷⁾	x (4/3)
528 x 576	4:3	x (4/3)	x 1 ¹⁾
	16:9	x (16/9) ⁷⁾	x (4/3)
480 x 576	4:3	x (3/2)	x (9/8) ¹⁾
	16:9	x 2 ⁷⁾	x (3/2)
352 x 576	4:3	x 2	x (3/2) ¹⁾
	16:9	x (8/3) ⁷⁾	x 2
352 x 288	4:3	x 2 ²⁾	x (3/2) ^{1) 2)}
	16:9	x (8/3) ^{2) 7)}	x 2 ²⁾
720 x 480	4:3	x 1	x (3/4) ¹⁾
	16:9	x (4/3) ⁷⁾	x 1
704 x 480	4:3	x 1	x (3/4) ¹⁾
	16:9	x (4/3) ⁷⁾	x 1
640 x 480	4:3 ⁴⁾	x (9/8) ⁵⁾	x (27/32) ¹⁾
544 x 480	4:3	x (4/3)	x 1 ¹⁾
	16:9	x (16/9) ⁷⁾	x (4/3)
528 x 480	4:3	x (4/3)	x 1 ¹⁾
	16:9	x (16/9) ⁷⁾	x (4/3)
480 x 480	4:3	x (3/2)	x (9/8) ¹⁾
	16:9	x 2 ⁷⁾	x (3/2)
352 x 480	4:3	x 2	x (3/2) ¹⁾
	16:9	x (8/3) ⁷⁾	x 2
352 x 240	4:3	x 2 ²⁾	x (3/2) ^{1) 2)}
	16:9	x (8/3) ^{2) 7)}	x 2 ²⁾

note 1 : this upsampling may be optional as 16:9 monitors can (in general) be switched to operate in 4:3 mode.
note 2 : also vertical upsampling x 2
note 3: upsampling to a window with a width of 704 pixels within an active area with a width of 720 pixels
note 4 : in this case the aspect_ratio_information field may be coded with '0001', indicating an aspect ratio of 1
note 5 : x (11/10) in case of 704 active pixels per line
note 6 : the coded_picture_size is defined to be equal to the horizontal_size_value by the vertical_size_value
note 7 : this upsampling is only applied to the (3/4)x(horizontal_size_value) pixels from the 16:9 picture to be displayed on the 4:3 display

6.10.1.4 Non-Full Screen Pictures

If the sequence_display_extension is present, and the coded picture size is smaller than or equal to the display picture size, then the display picture size shall be any of the values from Table 6-5, with the required upsampling ratios for display on 4:3 and 16:9 monitors with 720 active pixels per line. For the purpose of this clause, the coded picture size is smaller than or equal to the display picture size if both the horizontal size and the vertical size of the coded picture are smaller than or equal to the horizontal size and the vertical size of the displayed picture respectively.

Table 6-5. — Horizontal Upsampling Ratios For Non-Full Screen Pictures

<i>displayed_picture_size</i> ⁶⁾	<i>permitted coded_picture_size</i>	<i>source_aspect_ratio</i>	<i>horizontal upsampling ratios</i> 4:3 monitor 16:9 monitor	
720 x 576	<=720 x <=576	4:3	x 1	x (3/4) ¹⁾
		16:9	x (4/3) ⁷⁾	x 1
704 x 576	<=704 x <=576	4:3	x 1 ³⁾	x (3/4) ¹⁾
		16:9	x (4/3) ⁷⁾	x 1
544 x 576	<=544 x <=576	4:3	x (4/3)	x 1 ¹⁾
		16:9	x (16/9) ⁷⁾	x (4/3)
528 x 576	<=528 x <=576	4:3	x (4/3)	x 1 ¹⁾
		16:9	x (16/9) ⁷⁾	x (4/3)
480 x 576	<=480 x <=576	4:3	x (3/2)	x (9/8) ¹⁾
		16:9	x 2 ⁷⁾	x (3/2)
352 x 576	<=352 x <=576	4:3	x 2	x (3/2) ¹⁾
		16:9	x (8/3) ⁷⁾	x 2
352 x 288	<=352 x <=288	4:3	x 2 ²⁾	x (3/2) ^{1) 2)}
		16:9	x (8/3) ^{2) 7)}	x 2 ²⁾
720 x 480	<=720 x <=480	4:3	x 1	x (3/4) ¹⁾
		16:9	x (4/3) ⁷⁾	x 1
704 x 480	<=704 x <=480	4:3	x 1	x (3/4) ¹⁾
		16:9	x (4/3) ⁷⁾	x 1
640 x 480	<=640 x <=480	4:3 ⁴⁾	x (9/8) ⁵⁾	x (27/32) ¹⁾
544 x 480	<=544 x <=480	4:3	x (4/3)	x 1 ¹⁾
		16:9	x (16/9) ⁷⁾	x (4/3)
528 x 480	<=528 x <=480	4:3	x (4/3)	x 1 ¹⁾
		16:9	x (16/9) ⁷⁾	x (4/3)
480 x 480	<=480 x <=480	4:3	x (3/2)	x (9/8) ¹⁾
		16:9	x 2 ⁷⁾	x (3/2)
352 x 480	<=352 x <=480	4:3	x 2	x (3/2) ¹⁾
		16:9	x (8/3) ⁷⁾	x 2
352 x 240	<=352 x <=240	4:3	x 2 ²⁾	x (3/2) ^{1) 2)}
		16:9	x (8/3) ^{2) 7)}	x 2 ²⁾
note 1 : this upsampling is optional as 16:9 monitors can (in general) be switched to operate in 4:3 mode. note 2 : also vertical upsampling x 2 note 3: upsampling to a window with a width of 704 pixels within an active area with a width of 720 pixels note 4 : in this case the aspect_ratio_information field may be coded with '0001', indicating an aspect ratio of 1 note 5 : x (11/10) in case of 704 active pixels per line note 6 : displayed_picture_size is defined to be equal to the display_horizontal_size by the display_vertical_size note 7 : this upsampling is only applied to the (3/4)x(horizontal_size_value) pixels from the 16:9 picture to be displayed on the 4:3 display				

In the absence of control by the high-level API over the position of non-full screen video, the position of the coded picture video on the display (as defined by the displayed picture size), is defined by the pan vectors, when included. If no pan vectors are included, a default position shall be assumed in the center of the displayed picture. If the high-level API takes control over the position, the pan vectors, when included, may be disregarded.

DAVIC supports trick mode for compressed video by indicating the trick mode in the PES packet header in which the MPEG video data is contained, fully compliant to ISO/IEC 13818-1.

6.10.2 Coding constraints for video with a resolution beyond ITU-R 601

6.10.2.1 High profile

The allowable parameters shall be bounded by the upper limits specified for the Main Profile at High Level.¹ Additionally, the MPEG-2 video data shall meet the constraints and specifications described in the following sections².

6.10.2.2 Syntactical Constraints

The following tables list the allowed values for each of the ISO/IEC 13818-2 syntactic elements which are restricted beyond the limits imposed by MP@HL. In these tables conventional numbers denote decimal values, numbers preceded by **0x** are to be interpreted as hexadecimal values and numbers within single quotes (e.g., '10010100') are to be interpreted as a string of binary digits. The following table identifies parameters in the sequence header of a bit stream that shall be constrained and lists the allowed values for each.

Table 6-6. — Sequence Header Constraints

<i>Sequence header syntactic element</i>	<i>Allowed value</i>
horizontal_size_value	see Table 6-7
vertical_size_value	see Table 6-7
aspect_ratio_information	see Table 6-7
frame_rate_code	see Table 6-7
bit_rate_value (≤ 38.8 Mbps)	≤ 97000
vbv_buffer_size_value	≤ 488

6.10.2.3 Compression Format Constraints

The following table lists the allowed compression formats.

Table 6-7. — Compression Format Constraints³

<i>vertical_size_value</i>	<i>horizontal_size_value</i>	<i>aspect_ratio_information</i>	<i>frame_rate_code</i>	<i>progressive_sequence</i>
1080 ⁴	1920	1, 3	1,2,3,4,5	1
			3,4,5	0
1035 ⁴	1920	3	4,5	0
720	1280	1,3	1,2,3,4,5,6,7,8	1
576	720	2, 3	6	1
			3	1
			3	0
	704	2, 3	6	1
			3	1

¹ See ISO/IEC 13818-2, Section 8 for more information regarding profiles and levels.

² The additional constraints and specifications are based on ATSC A/53 and, in part, on ITU-R BT.1208.

³ Shaded areas describe formats up to ITU-R resolution. This information is provided for convenience and does not constrain in any manner the functionality for coding of video up to ITU-R 601 resolution.

⁴ Note that 1088 lines are actually coded in order to satisfy the MPEG-2 requirement that the coded vertical size be a multiple of 16 (progressive scan) or 32 (interlaced scan) and to provide a common coding format for 1035 and 1080.

<i>vertical_size_ value</i>	<i>horizontal_size_ value</i>	<i>aspect_ratio_ information</i>	<i>frame_rate_ code</i>	<i>progressive_ sequence</i>
	544	2, 3	3	0
			3	1
			3	0
			3	1
	528	2, 3	3	0
			3	1
			3	0
			3	1
480	720	2, 3	2,5,7,8	1
			1, 4	1
			5	0
			4	0
	704	2, 3	2,5,7,8	1
			1,4	1
			5	0
			4	0
	640	1, 2, 3	2,5,7,8	1
			1,4	1
			5	0
			4	0
	544	2, 3	1,4	1
			4	0
	528	2, 3	1,4	1
			4	0
	480	2, 3	1,4	1
			4	0
	352	2, 3	1,4	1
			4	0
288	352	2, 3	3	1
240	352	2, 3	1,4	1

<i>Legend for MPEG-2 coded values</i>					
<i>aspect_ratio_information</i>	1 = square samples	2 = 4:3 display aspect ratio	3 = 16:9 display aspect ratio		
<i>frame_rate_code</i>	1 = 23.976 Hz	2 = 24 Hz	3 = 25 Hz	4 = 29.97 Hz	5 = 30 Hz
	7 = 59.94 Hz	8 = 60 Hz			
<i>progressive_sequence</i>	0 = interlaced scan	1 = progressive scan			

6.10.2.4 Sequence Extension Constraints

The following table identifies parameters in the sequence extension part of a bit stream that shall be constrained by the video subsystem and lists the allowed values for each. A `sequence_extension` structure is required to be present after every `sequence_header` structure.

Table 6-8. — Sequence Extension Constraints

<i>Sequence extension syntactic element</i>	<i>Allowed values</i>
<code>progressive_sequence</code>	see Table 6-7
<code>profile_and_level_indication</code>	see Note
<code>chroma_format</code>	'01'

<i>Sequence extension syntactic element</i>	<i>Allowed values</i>
horizontal_size_extension	'00'
vertical_size_extension	'00'
bit_rate_extension	'0000 0000 0000'
vbv_buffer_size_extension	'0000 0000'
frame_rate_extension_n	'00'
frame_rate_extension_d	'0000 0'

Note: The profile_and_level_indication field shall indicate the lowest profile and level defined in ISO/IEC 13818-2, Subclause 8, that is consistent with the parameters of the video elementary stream.

6.10.2.5 Sequence Display Extension Constraints

The following table identifies parameters in the sequence display extension part of a bit stream that shall be constrained by the video subsystem and lists the allowed values for each.

Table 6-9. — Sequence Display Extension Constraints

<i>Sequence display extension syntactic element</i>	<i>Allowed values</i>
video_format	'000'

The preferred and default values for color_primaries, transfer_characteristics, and matrix_coefficients are defined to be SMPTE 274M⁵ (value 0x01 in all three cases). While all values described by MPEG-2 are allowed in the transmitted bit stream, it is noted that SMPTE 170M values (0x06 in all three cases) will be the most likely alternate in common use.

6.10.2.6 Picture Header Constraints

In all cases other than when vbv_delay has the value 0xFFFF, the value of vbv_delay shall be constrained as follows:

$$\text{vbv_delay} \bullet 45000$$

6.10.3 Decoding tool requirements

6.10.3.1 Decoding tools for video with a resolution up to ITU-R 601

DAVIC tools for decoding video with resolutions up to ITU-R 601 shall decode bitstreams of all of the specified resolutions into the output for which the specific STU was designed. When video encoded with 525 lines at a framerate of 29.97 Hz is delivered to the STU, the STU shall produce a baseband NTSC output signal or some other video output signal for which the STU was designed. Similarly, when video encoded with 625 lines at a framerate of 25 Hz is delivered to the STU, the STU shall produce a baseband PAL output signal or some other video output signal for which the STU was designed.

6.10.3.2 Decoding tools for video with a resolution beyond ITU-R 601

DAVIC tools for decoding video with resolutions higher than ITU-R 601 shall decode bitstreams of all of the specified resolutions into the output for which the specific STU was designed. The informative Annex K explains some implementation examples for providing compatibility with each video format such that each decoder can decode and display each DAVIC input bitstream resolution.

⁵ At some point in the future, the color gamut may be extended by allowing negative values of RGB and defining the transfer characteristics for negative RGB values.

6.11. Still Pictures

Each still picture shall be encoded as an MPEG-2 Intra Frame; see ISO/IEC 13818-2. In the case of a series of still pictures with a presentation schedule associated to a time base (e.g. a series representing a slide show), then the still picture feature of MPEG-2 Systems (ISO/IEC 13818-1) shall be used. For the definition of still pictures in terms of MPEG Systems refer to Clause 2.1.52 of ISO/IEC 13818-1. In the case of a single still picture (bitmap), only ISO/IEC 13818-2 applies.

6.11.1 Normal resolution still pictures

For still pictures with coded picture size smaller than or equal to 720 pixels by 576 lines, the picture size constraints specified in Clause 6.10.1 apply.

6.11.2 Higher resolution still pictures

For still pictures with coded picture size larger than 720 pixels by 576 lines, the picture size constraints specified in Clause 6.10.2 apply.

6.12. Compressed Graphics

6.12.1 Requirements

The following prioritized list of criteria and requirements for compression of graphics have been identified. A larger value indicates a lower priority.

1. Ease and speed of software implementation in a STU
2. Minimum use of memory in the STU
3. Visual appearance while decoding is in progress in the STU
4. Lossless
5. Efficiency of compression
6. Technology available without IPR royalty

6.12.2 Format for Compressed Graphics

In ISO/IEC 16500 compressed graphics shall be coded using ETS 300 743, also known as the DVB subtitling system.

6.12.2.1 Real-Time Compressed Graphic streams

If the presentation of the graphics is associated to the decoder system time clock (STC as defined in ISO/IEC 13818-1) then ETS 300 743 is fully applicable.

6.12.2.2 Single bitmaps with Compressed Graphics

In the case of single bitmaps with compressed graphics of which the presentation is not associated to a time base then each bitmap is represented by a “Region Composition Segment” followed by each “Object Data Segment” referred to; for the definition of these segments refer to Clauses 1.8.2 and 1.8.4 of ETS 300 743. For decoding a single bitmap with compressed graphics a separate colour palette may be available that fully complies to the “CLUT definition segment” as specified in Clause 1.8.3 of ETS 300 743. For single bitmaps with compressed graphics the encoded value of the following fields of Region Composition Segments, Object Data Segments and CLUT Definition Segments may be disregarded at decoding:

- page_id
- region_id
- region_version_number
- object_version_number
- CLUT-version_number

6.12.2.3 Default CLUT for single bitmaps with Compressed Graphics

ISO/IEC 16500 specifies a default CLUT for single bitmaps with compressed graphics. The default CLUT is applicable when the use of no other CLUT is specified in this bitmap. A Visible Object, when referring to a Palette Object, will interpret specified IndexColours as entries in the default CLUT, except for Bitmap objects that use any specific Palette Object.

6.12.2.3.1 Requirements

The default CLUT for use by single bitmaps with compressed graphics is to meet the following requirements :

- the default CLUT shall define a CLUT with 256 entries
- the default CLUT shall support transparency values of 0 %, 25 %, 50 %, 75 % and 100 %
- the colours defined by default CLUT shall have a perceptually uniform distribution over the colour space

- the default CLUT shall leave some colours available for private use
- the colours defined by the default CLUT, when used in an STU implementation as the basic colours for the presentation of graphic images, are suitable for rendering any graphic image
- the R, G and B components of the colours defined by the default CLUT are orthogonal, i.e. quantization of a random colour towards a colour defined by the default CLUT is possible for each R, G or B component independently of the value and quantization of the other components.

6.12.2.3.2 Default CLUT characteristics

To meet the above requirements, a default CLUT is defined with properties as given in the following table, where for each transparency level the total number of CLUT entries is given, as well as the quantization levels for the Red, Green and Blue component of the colour associated to the entry. Each possible combination of component quantization levels is allowed for. As a consequence the total number of entries per transparency level is found by multiplying the number of quantization levels of the Red, Green and Blue components for that transparency level. The default CLUT reserves 12 entries for private use.

Table 6-9. — Default CLUT characteristics

<i>Transparency level</i>	<i>Total number of CLUT entries</i>	<i>Quantization levels for Red</i>	<i>Quantization levels for Green</i>	<i>Quantization levels for Blue</i>
0 % (fully opaque)	135	5 levels : 0, 63, 127, 191, 255	9 levels : 0, 31, 63, 95, 127, 159, 191, 223, 255	3 levels : 0, 127, 255
25 %	56	4 levels : 0, 85, 170, 255	7 levels : 0, 42, 85, 127, 170, 212, 255	2 levels : 0, 255
50 %	36	3 levels : 0, 127, 255	6 levels : 0, 51, 102, 153, 204, 255	2 levels : 0, 255
75 %	16	2 levels : 0, 255	4 levels : 0, 85, 170, 255	2 levels : 0, 255
100 % (fully transparent)	1	-	-	-
privately defined	12	privately defined	privately defined	privately defined

6.12.2.3.3 Default CLUT specification

The colours of the default CLUT are defined in Annex C of this specification.

6.13. Compressed Character Data

Compressed character data shall be coded using SCTE DVS/026, defined by the Society of Cable Telecommunications Engineers (SCTE).

6.14. Network Graphics

As format for Network Graphics DAVIC supports the PNG format as defined by the World Wide Web Consortium (W3C Recommendation 01 October 1996, available at <http://www.w3.org/TR/REC-png-multi.html>).

7. Monomedia Streams

7.1. Types of Monomedia Components

DAVIC distinguishes three types of monomedia components. The first type are implied monomedia components; implied components are either included within other monomedia components (e.g. characters within text) or are implied by some other mechanism defined by DAVIC, such as the use of Service Information within an MPEG-2 Transport stream. The monomedia components which are not of the implied type are one of the two following types :

- Monomedia streams. The monomedia components of which the presentation requires synchronization with a time base are represented as monomedia streams. For such synchronization the mechanisms provided by MPEG Systems, ISO/IEC 13818-1, are used. To enable the use of time stamps as defined in ISO/IEC 13818-1, each monomedia stream is packetized in PES packets.
- Stand-alone monomedia components. Monomedia components that do not require synchronization with a time base at play back, are referred to as stand-alone monomedia components. Stand-alone monomedia components are not packetized in PES packets.

Table 7-1 defines which type each DAVIC defined monomedia component may take. Some monomedia components can be of more than one type, specifically they can be a monomedia stream and a stand-alone component. Specifically a graphical object may either be a single bitmap (such as a button for user control) or a graphical representation of text used within a graphical stream for subtitling. Furthermore a still picture may be a single bitmap, e.g. representing a background for a downloaded application, or may be part of a series of still pictures representing a slide show.

Table 7-1. — Types of monomedia components

<i>Monomedia component</i>	<i>Representation</i>
Characters	implied component, e.g. in Text and Service Information
Text	stand-alone component
Outline Fonts	stand-alone component
Language Information	implied component, e.g. in Text and Service Information
Service Information	implied component in MPEG-2 Transport Stream
Telephone Numbers	implied component, e.g. in downloaded Application
Compressed Audio	stream
Scaleable Audio	stream
Linear Audio	stream
Compressed Video	stream
Still Picture series	stream
Still Picture bitmap	stand-alone component
Compressed graphic real-time stream	stream
Compressed graphic bitmap	stand-alone component
Compressed character data stream	stream
Network graphic bitmap	stand-alone component

7.2. Real-time and Stored Monomedia Streams

For monomedia streams ISO/IEC 16500 distinguishes real-time streams and stored streams, which differ in the process which drives their play back.

- The playback of real-time streams is real time driven; real-time streams are played back in reference to a time base which is controlled by the PCR fields from the MPEG-2 Transport Stream that contains the real-time stream. With real-time streams the timing of the decoding and presentation in an STU is controlled by the characteristics of the stream during the delivery of the stream to the STU.
- The timing of the playback of stored streams is controlled by an application. The stream is stored in STU memory, providing the functionality of a local server that can be controlled from an application that also resides in the STU. As an example, in a certain application it may be possible for the user to play back a stored stream by pushing a specific button on a Remote Control.

Also transmission requirements differ for real-time streams and stored streams. In the case of real-time streams, constraints for real-time delivery apply to avoid problems such as buffer underflow and overflow. For synchronization of real-time streams and stored streams during playback, the mechanisms provided by ISO/IEC 13818-1 are applied by ISO/IEC 16500. Therefore monomedia streams are stored in PES packets.

While the data structure used to carry real-time streams and stored streams in PES packets is the same, there are important consequences for the encoding and interpretation of PTSs and DTSs. In the case of real-time streams the time base of the program is used, as specified in ISO/IEC 13818-1 for MPEG audio and MPEG video. For real-time AC-3 audio streams the time base of the program specified in ISO/IEC 13818-1 is also used. In the case of stored streams an independent time base may be used, but the phase of the time base for playback of the stored streams is determined by the application. Each time a stored stream is played back, the time base (system time clock) starts with a value of zero.

Table 7-2 summarizes for each ISO/IEC 16500 non-implied monomedia component whether it can be defined as a real-time stream, as a stored stream object and whether it is contained in a PES packet.

Table 7-2. — Real-time stream, stored stream and use of PES packets per monomedia component

<i>Monomedia component (non-implied)</i>	<i>Real-time stream</i>	<i>Stored stream</i>	<i>contained in PES packets</i>
Text	no	no	no
Outline Fonts	no	no	no
Compressed Audio	yes	yes	yes
Scaleable Audio	yes	yes	yes
Linear Audio	no	yes	yes
Compressed Video	yes	yes	yes
Still Picture series	yes	yes	yes
Still Picture bitmap	no	no	no
Compressed graphic real-time stream	yes	yes	yes
Compressed graphic bitmap	no	no	no
Compressed character data stream	yes	no	no
Network graphic bitmap	no	no	no

7.3. Carriage of Monomedia Streams in PES Packets

Monomedia streams are carried in PES packets, with the only exception of Compressed Character Data. One of the advantages of applying PES packets is that they provide a general structure with the following major features :

- identification of monomedia components.
- association of Time Stamps with monomedia streams. Time Stamps (PTSs and DTSSs) are essential to ensure synchronisation and to ensure correct buffer behavior in the T-STD defined in ISO/IEC 13818-1 and in the Reference Decoder Model for contents decoding defined in ISO/IEC 16500.
- association of trick mode operation with monomedia streams, specifically compressed video.
- association of copyright information with monomedia streams.

7.3.1 Packetization of MPEG- and ATSC-defined Components

The packetization of MPEG- and ATSC-defined monomedia streams in PES packets shall fully comply to ISO13818-1. In addition the following constraints shall apply :

- In the case of a stored stream containing compressed audio, the first byte of the first PES packet with data from the stored stream shall be the first byte of an MPEG or AC-3 audio frame, i.e. the first byte of a sync word. In addition the last byte of the last PES packet containing data from the stored stream shall be the last byte from an audio frame. Consequently, each stored audio stream must have an integer number of audio frames.
- In the case of a stored stream containing compressed video or a series of still pictures, the first byte of the first PES packet with data from the stored stream shall be the first byte of a sequence_start_code. In addition, the PES_packet_length field shall not be encoded with the value of zero. The sequence header associated with this start code shall be followed immediately by an I-picture, optionally preceded by a GOP header. The last byte of the last PES packet containing data from the stored stream shall be the last byte of a sequence_end_code. Consequently, each stored video or still picture stream must have an integer number of video frames.

7.3.2 Packetization of DVB-defined Components

The packetization of DVB subtitling in PES packets shall fully comply to ETS 300 743.

7.3.3 Packetization of DAVIC-defined Components

The packetization in PES packets of monomedia streams with a coding format defined by ISO/IEC 16500 is defined in Annex D of this specification.

7.4. Packetization of Compressed Character Data

Compressed character data are carried in MPEG private sections, defined in ISO/IEC 13818-1. The packetization of Compressed Character Data within private sections shall fully comply to SCTE DVS/026.

8. Transport of Monomedia Streams and Components

8.1. Transport of Real Time Streams

Real time streams are mapped directly to the MPEG-2 Transport Stream, without using DSM-CC. For real-time streams containing compressed audio, compressed video, or a series of still pictures, the mapping shall comply to ISO/IEC 13818-1. In the case of ISO/IEC 16500 defined compressed graphics, the mapping shall comply to ETS 300 743. In the case of Compressed character data the mapping shall comply to SCTE DVS/026.

8.2. Transport of Stored Streams

In ISO/IEC 16500 the transport method of stored streams depends on whether upstream information flow is used by the delivery system. For applications which utilize upstream information flow, the stored streams may be transmitted in reply to the DSM-CC File Read function (refer to clause 7.3.8.1; file access) with the UNO remote procedure call, data representation and transport mechanism (refer to clause 7.2.2, user-to-user interaction; 7.3.1 DSM-CC option choices summary; and 7.3.2, remote procedure call) as specified in ISO/IEC 16500-5. Alternatively, the stored streams are transmitted in “User-to-User-Object Carousels” within MPEG-2 Transport Streams, using private MPEG-2 sections, as specified by DSM-CC.

8.3. Transport of Stand-alone Monomedia Components

To transport stand-alone monomedia components the same mechanisms are used as for stored streams; see Clause 8.2.

9. Application Format

9.1. Application Interchange Format

To deliver multimedia information to STUs in an interoperable way, applications shall use the MHEG-5 final form interchange format, as defined by ISO/IEC 13522-5. The ASN.1 notation and encoding, as defined by Annex A of ISO/IEC 13522-5, shall be used to interchange MHEG-5 objects. This format defines the semantics and the encoding of the multimedia and hypermedia objects. Subclause 9.2 below specifies the MHEG-5 options whose support by the DAVIC platform is mandatory or optional, as well as the semantic extensions to MHEG-5 related to the DAVIC application domain.

To deliver program code to STUs in an interoperable way, applications shall use the MHEG-5 InterchangedProgram class to encapsulate Java⁶ VM code, according to the semantics and encoding defined by ISO/IEC 13522-6. Java VM classes are called from MHEG-5 objects using the MHEG-5 Call and Fork actions.

The Java VM code interchange unit is a Java VM class. Java VM classes shall be encoded as defined by the *Class File Format* section of the *Java Virtual machine specification*. A Java class encapsulates data and methods that consist of sequences of instructions. The instruction set is defined by the *Java Virtual machine instruction set* section of the *Java Virtual machine specification*.

⁶ Java is a trademark or a registered trademark of Sun Microsystems, Inc.

9.2. MHEG-5 Profile for the DAVIC Application Domain

This subclause specifies the features of ISO/IEC 13522-5 that shall be supported by the DAVIC application domain, according to the application domain definition principles set by Annex D of ISO/IEC 13522-5.

9.2.1 Object Interchange Format

The ASN.1 notation defined in Annex A of ISO/IEC 13522-5 shall be used as the application interchange format of ISO/IEC 13522-5, as well as the Distinguished Encoding Rules (DER) for the encoding of the interchanged objects.

9.2.2 Set of Classes

The following set of MHEG-5 classes shall be mandatory:

Action, Application, Audio, Bitmap, BooleanVariable, ContentRefVariable, DynamicLineArt, EntryField, HotSpot, HyperText, IntegerVariable, InterchangedProgram, Link, ListGroup, ObjectRefVariable, OctetStringVariable, Palette, PushButton, Rectangle, RemoteProgram, ResidentProgram, RTGraphics, Scene, Slider, Stream, SwitchButton, Text, TokenGroup, Video.

9.2.3 Set of Features

The set of mandatory and optional features shall be as defined in Table 9-1.

Table 9-1. — Feature requirements

<i>Feature</i>	<i>Requirement</i>
Caching	Optional
Cloning	Mandatory
Free moving cursor	Optional
Scaling (Video and Bitmap)	Optional
Stacking of Applications	Optional
Trick modes	Optional
Ancillary connections	Optional

9.2.4 Content Data Encoding

DAVIC specifies the content data encoding as defined in Table 9-2.

Table 9-2. — Content Encoding

<i>Type of content</i>	<i>Specification (Data Types)</i>	<i>Hook values</i>	<i>Clause</i>
Font encoding format	ISO/IEC 16500 defined Outline Font Format	1	6.3
Palette encoding format	CLUT Definition Segment (ETS 300 743) ¹⁾	1	6.12.2.2
Bitmap encoding format	reserved	1	6.11 6.12.2.2 6.14
	MPEG-2 Intra frame	2	
	Region Definition Segment (ETS 300 743)	3	
	PNG bitmap	4	
Text encoding format	Subset of HTML 3.2	1	6.2

EntryField encoding format	Characters encoded according to ISO 10646-1 (Unicode) or according to ISO 8859-1, depending on the value of the CharacterSet Attribute of the Entryfield	1	6.1
HyperText encoding format	Subset of HTML 3.2	1	6.2
Stream encoding format	video: MPEG-1 Video (ISO/IEC 11172-2) MPEG-2 Video (ISO/IEC 13818-2) audio: MPEG-1 Audio (ISO/IEC 11172-3) rtgraphics: DVB Subtitling (ETS 300 743)	1 ²⁾	6.10 6.10 6.7.1 6.12.2.1
	video: MPEG-2 Still (ISO/IEC 13818-2) audio: MPEG-1 Audio (ISO/IEC 11172-3) rtgraphics: DVB Subtitling (ETS 300 743)	2 ²⁾	6.10 6.7.1 6.12.2.1
	video: - audio: DAVIC Linear Audio (AIFF-C) rtgraphics: -	3 ³⁾	6.9
	video: MPEG-1 Video (ISO/IEC 11172-2) MPEG-2 Video (ISO/IEC 13818-2) audio: AC-3 Audio (ATSC A52) rtgraphics: DVB Subtitling (ETS 300 743)	4 ²⁾	6.10 6.10 6.7.2 6.12.2.1
	video: MPEG-2 Still (ISO/IEC 13818-2) audio: AC-3 Audio (ATSC A52) rtgraphics: DVB Subtitling (ETS 300 743)	5 ²⁾	6.10 6.7.2 6.12.2.1
	video: MPEG-1 Video (ISO/IEC 11172-2) MPEG-2 Video (ISO/IEC 13818-2) scaleable audio: MPEG-2 Audio (ISO/IEC 13818-3, layer II) rtgraphics: DVB Subtitling (ETS 300 743)	6 ²⁾	6.10 6.10 6.8 6.12.2.1
	video: MPEG-2 Still (ISO/IEC 13818-2) scaleable audio: MPEG-2 Audio (ISO/IEC 13818-3, layer II) rtgraphics: DVB Subtitling (ETS 300 743)	7 ²⁾	6.10 6.8 6.12.2.1
LineArt encoding format	(None specified)		
CursorShape encoding format	(None specified)		
InterchangedProgram encoding format	MHEG-6 (ISO/IEC 13522-6)	1	9.1
<i>Note 1 : Only for use by bitmaps coded in the Region Definition Segment format</i> <i>Note 2 : For Stream objects with ContentHook 1, 2, 4, 5, 6 and 7, the value of the attribute Storage shall be “stream” or “memory”.</i> <i>Note 3 : For Stream objects with the ContentHook 3, the value of the attribute Storage shall be “memory”.</i>			

9.2.5 Attribute Encoding

DAVIC specifies the attribute encoding as defined in Table 9-3.

Table 9-3. — Attribute Encoding

<i>Attribute</i>	<i>Specification (Data Types)</i>
Permissible FontAttributes	“<style>.<size>“ with <style> being “plain”, “bold”, “italic”, “bold-italic”, “emphasis” or “strong”, and <size> being an integer. e.g. “bold-italic.20”
Permissible FontNames	“fixed” specifying a font with a fixed spacing “proportional” specifying a font with proportional spacing
TransitionEffects	See Clause 9.2.11
CharacterSet	1: ISO 8859-1 (Latin) 2: ISO 10646-1 (Unicode)
AbsoluteColour	RGBα32, that is coding of graphics in 32 bits per pixel, allocating 8 bits to the Red, Green and Blue components, as well as 8 bits for the translucency component. The first three bytes are unsigned integers, providing the value of the red, green and blue components of the pixel respectively. The value 0 indicates minimum value, while the value 255 indicates maximum value. The last byte specifies the transparency level of the pixel. The value zero indicates an opaque pixel, while the value 255 indicates full transparency. The value shall be coded as the big-endian octet string representation of the integer value.

9.2.6 UserInput Registers

DAVIC specifies the following InputEventRegisters defined in Table 9-4.

Table 9-4. — InputEventRegisters

<i>Register #</i>	<i>UserInputEventTag</i>	<i>Name</i>
1	1	Up
	2	Down
	3	Left
	4	Right
	5 - 14	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, respectively
	15	Select
	16	Exit
	17	Help
	18-99	Reserved for future specification.
	>99	Vendor specific
2	1	Up
	2	Down
	3	Left
	4	Right

5-14	Forbidden
15	Select
16	Exit
17	Help
18-99	Reserved for future specification.
>99	Vendor specific

9.2.7 Constraints on the Use of Variables

The use of IntegerVariables shall be restricted as defined in Table 9-5.

Table 9-5. — Constraints to IntegerVariables

<i>type of value</i>	<i>default value</i>	<i>Size</i>	<i>min value</i>	<i>max. value</i>
signed integer	0	32 bits	-214783648	214783647

9.2.8 Semantic Constraints on the MHEG-5 Applications

Table 9-6 provides a list of constraints that each MHEG-5 applications shall apply to features of ISO/IEC 13522-5. STUs may support the features to a higher or lesser degree.

Table 9-6. — Feature Constraints

<i>Feature</i>	<i>Constraint</i>
SceneCoordinateSystem(X,Y)	The following coordinate systems are supported: 720x576, 720x480, 640x480 and 1080x1920.
SceneAspectRatio(W,H)	The following aspect ratios are supported: 1/1, 4/3 and 16/9.
MultipleRTGraphicsStreams(N)	One active RTGraphic stream at a time.
MultipleAudioStreams(N)	One active MPEG-1 audio stream at a time and one active AIFF-C audio stream at a time.
MultipleVideoStreams(N)	One active video stream at a time.
OverlappingVisibles(N)	No constraint.
Cloning	Mandatory

9.2.9 EngineEvent

The DAVIC application domain reserves no particular value of EngineEvent. This is left to the application developer.

9.2.10 GetEngineSupport

DAVIC specifies no other strings for the GetEngineSupport action than the ones listed in ISO/IEC 13522-5.

9.2.11 TransitionEffect Parameter of the TransitionTo Elementary Action

To support transition effects between Still Picture Compositions in ISO/IEC 16500, the TransitionEffect parameter of the TransitionTo elementary action can be used, as specified in this Clause. For a graphical description of the transition effects see Annex M.

The TransmissionEffect parameter consists of 32 bits. The structure of the parameter is as follows :

Transition Effect Parameter: [parameter_1] [parameter_2] [parameter_3] [parameter_4] [parameter_5]

[parameter_1] : Parameter_1 is 7 bits field of type uimsbf which defines transition effect type

[parameter_2] : Parameter_2 is 3 bits field of type uimsbf which defines number of splits

[parameter_3] : Parameter_3 is 6 bits field of type uimsbf which defines duration of effect time

[parameter_4] : Parameter_4 is 8 bits field of type uimsbf which defines horizontal position of effect start

[parameter_5] : Parameter_5 is 8 bits field of type uimsbf which defines vertical position of effect start

Table 9-7. — TransitionEffect parameter of “TransitionTo” action

No.	Transition Effect type	Abbrevia- tion	Para- meter_1	Para- meter_2	Para- meter_3	Para- meter_4	Para- meter_5
0	Reserved		00				
1	Simple cut	CUT	01				
2	Dissolve	DIS	02		T		
3	Vertical wipe (top to bottom)	WVD	03		T		
4	Vertical wipe (bottom to top)	WVU	04		T		
5	Horizontal wipe (left to right)	WHR	05		T		
6	Horizontal wipe (right to left)	WHL	06		T		
7	Wipe that vertically opens from the center	WVO	07		T		Y
8	Wipe that vertically closes from both top and bottom	WVC	08		T		Y
9	Wipe that horizontally opens from the center	WHO	09		T	X	
10	Wipe that horizontally closes from both left and right	WHC	0A		T	X	
11	Square wipe (open)	WRO	0B		T	X	Y
12	Square wipe (close)	WRC	0C		T	X	Y
13	Vertical split wipe (top to bottom)	WDD	0D	N	T		
14	Vertical split wipe (bottom to top)	WDU	0E	N	T		
15	Horizontal split wipe (left to right)	WDR	0F	N	T		
16	Horizontal split wipe (right to left)	WDL	10	N	T		
17	Vertical slide-out (top to bottom)	SOD	11		T		
18	Vertical slide-out (bottom to top)	SOU	12		T		
19	Horizontal slide-out (left to right)	SOR	13		T		
20	Horizontal slide-out (right to left)	SOL	14		T		
21	Vertical slide-in (top to bottom)	SID	15		T		
22	Vertical slide-in (bottom to top)	SIU	16		T		

23	Horizontal slide-in (left to right)	SIR	17		T		
24	Horizontal slide-in (right to left)	SIL	18		T		
25	Vertical roll (top to bottom)	RVD	19		T		
26	Vertical roll (bottom to top)	RVU	1A		T		
27	Horizontal roll (left to right)	RHR	1B		T		
28	Horizontal roll (right to left)	RHL	1C		T		
29	Reserved		1D				
30	Reserved		1E				
31	Reserved		1F				
32	Reserved		20				
33	Reserved		21				
34	Reserved		22				
35	Reserved		23				
36	Half vertical roll (top to bottom)	RPD	24		T		Y
37	Half vertical roll (bottom to top)	RPU	25		T		Y
38	Half horizontal roll (left to right)	RPR	26		T	X	
39	Half horizontal roll(right to left)	RPL	27		T	X	
40	Reserved		28-7F				

The symbols used in Table 9-7 indicate the following:

N: An unsigned integer, specifying the number of splits as defined in Table 9-8. For the coordinate systems supported by DAVIC the start positions of the split-effects are defined in Table 9-9.

Table 9-8. — Number of splits

<i>Parameter Value</i>	<i>Number of splits</i>
0x0	0
0x1	2
0x2	3
0x3	4
0x4	5
0x5	8
0x6	10
0x7	Reserved

Table 9-9. — Start positions of split-effects with n splits per Coordinate System

<i>Transition Effect number</i>	<i>Transition Effect type with Split Effect</i>	<i>Start position in Coordinate System (x * y)</i> <i>Supported coordinate systems by DAVIC : (720 * 576), (720 * 480), (640 * 480), and (1920 * 1080). Note : the most left pixel and the top line have index value zero.</i>

13	Vertical split wipe (top to bottom)	lines $(y*(n-k)/n)$, where $k=n, n-1, \dots, 1$ and with truncation to zero after division.
14	Vertical split wipe (bottom to top)	lines $((y-1) - (y*(n-k)/n))$, where $k=n, n-1, \dots, 1$ and with truncation to zero after division.
15	Horizontal split wipe (right to left)	pixels $(x*(n-k)/n)$, where $k=n, n-1, \dots, 1$ and with truncation to zero after division.
16	Horizontal split wipe (left to right)	pixels $((x-1) - (x*(n-k)/n))$, where $k=n, n-1, \dots, 1$ and with truncation to zero after division.

- T: An unsigned integer, specifying the duration time of the effect, with an inclusive range of 0 – 63 indicating the duration time of the transition effect in units of 0.5 seconds.
- X: An unsigned 8 bit integer, specifying the horizontal start position of the effect. The horizontal start position indicates the horizontal index of the pixel of the coordinate system where the effect starts. The most left pixel of the coordinate system has index zero. For the four coordinate systems supported by DAVIC, 720 x 576, 720 x 480, 640 x 480 and 1920 x 1080 (see Table 9-6) the horizontal index of the pixel where the effect starts is defined by the formula in Table 9-10. If an index value larger than the index of the most right pixel of the coordinate system is indicated, then the effect is indicated to start at the most right pixel. Table 9-10 also specifies the range permitted for X.

Table 9-10. — Coding of horizontal index of pixel where the effect starts

Horizontal resolution of Coordinate System	Horizontal index of pixel where effect starts	Permitted range of X (inclusive)
640	$4*X$	0 - 160
720	$4*X$	0 - 180
1920	$8*X$	0 - 240

- Y: An unsigned 8 bit integer, specifying the vertical start position of the effect. The vertical start position indicates the vertical index of the line of the coordinate system where the effect starts. The top line of the coordinate system has index zero. For the four coordinate systems supported by DAVIC, 720 x 576, 720 x 480, 640 x 480 and 1920 x 1080 (see Table 9-6) the vertical index of the line where the effect starts is defined by the formula in Table 9-11. The formula is defined such that the effect can be defined to start in the exact center position of the coordinate system. If an index value larger than the index of the bottom line of the coordinate system is indicated, then the effect is indicated to start at the bottom line. Table 9-11 also specifies the range permitted for Y.

Table 9-11. — Coding of vertical index of line where the effect starts.

Vertical resolution of Coordinate System	Vertical index of line where effect starts	Permitted range of Y (inclusive)
480	$2*Y$	0 - 240
576	$4*Y$	0 - 144
1080	$8*Y + 4$	0 - 134

9.2.12 MHEG-5 Resident Programs

In order to improve the functionality of MHEG-5 applications, DAVIC has defined a set of MHEG-5 resident programs. These MHEG-5 resident programs offer the following facilities in a platform independent fashion :

- The ability to retrieve and manipulate date information
- The ability to request a random number