Table A4-22 — Function P9SPICR : Information Representation for Real-Time streams of Still Pictures

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Still Pictures - Specifies the use of MPEG-2 I frames	Part 9 Section 6.11.1		IDB
2	Types of Monomedia Components – Identifies Still Pictures as stream or as stand-alone component	Part 9 Section 7.1	Support of Still Pictures as a stand-alone component is not a part of P9SPICR	IDB
3	Real-time and Stored Monomedia Streams - identifies a stream of still pictures as an instance of both real-time and stored streams	Part 9 Section 7.2	Support of Still Pictures as a stored stream is not a part of P9SPICR	IDB
4	Carriage of Monomedia Streams in PES Packets - Specifies that streams of Still Pictures are contained in PES packets	Part 9 Section 7.3		IDB
5	Packetization of MPEG and ATSC-defined Components – Specifies some details with respect to the way Streams of Still pictures are packaged	Part 9 Section 7.3.1		IDB
6	Transport of Real-Time Streams	Part 9 Section 8.1		IDB

Table A4-23 — Function P9SPICS: Information Representation for stored streams of Still Pictures

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Still Pictures - Specifies the use of MPEG-2 I frames	Part 9 Section 6.11.1		IDB
2	Types of Monomedia Components - Identifies Still Pictures as stream or as stand-alone component	Part 9 Section 7.1	Support of Still Pictures as a stand-alone component is not a part of P9SPICS	IDB
3	Real-time and Stored Monomedia Streams – identifies a stream of still pictures as an instance of both real-time and stored streams	Part 9 Section 7.2	Support of Still Pictures as a Real-Time stream is not a part of P9SPICS	IDB
4	Carriage of Monomedia Streams in PES Packets - Specifies that streams of Still Pictures are contained in PES packets	Part 9 Section 7.3		IDB
5	Packetization of MPEG and ATSC-defined Components - Specifies some details with respect to the way Streams of Still pictures are packaged	Part 9 Section 7.3.1		IDB
6	Transport of Stored Streams	Part 9 Section 8.2		IDB

Table A4-24 — Function P9SPB: Information Representation for Still Picture bitmaps

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Still Pictures – Specifies the use of MPEG-2 I frames	Part 9 Section 6.11.1		IDB
2	Types of Monomedia Components - Identifies Still Pictures as stream or as stand-alone component	Part 9 Section 7.1	Support of Still Pictures as a stream is not a part of P9SPB	IDB
5	Packetization of MPEG and ATSC-defined Components - Specifies some details with respect to the way Streams of Still pictures are packaged	Part 9 Section 7.3.1		IDB
6	Transport of Stand-alone Monomedia Components	Part 9 Section 8.3		IDB

A.4.4.3.2 Compressed Graphics

DAVIC specifies the use of DVB subtitling for streams of graphics pictures encoded as bitmaps which can be used for e.g., subtitling and animations. Graphics streams can be used as a stream which is played directly from the network and also as a stored stream which is played from memory under control of an application. A third possibility is the use of a single Compressed Graphics bitmap as a stand-alone component. The following tables: indicate the clauses from DAVIC Part 9 relevant to compressed graphics in the context of the *Enhanced Digital* Broadcast Contour for the functions identified above.

Table A4-25 — Function P9CGR Information Representation for Compressed Graphics Real-Time streams

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Compressed Graphics – Specifies the use of DVB subtitling	Part 9 Section 6.12.2 , 6.12.2.1		IDB
2	Types of Monomedia Components – Identifies Compressed Graphics as a stream or as a stand-alone component	Part 9 Section 7.1	Support of Compressed Graphics as a stand-alone component is not a part of P9CGR	IDB
3	Real-time and Stored Monomedia Streams - Identifies Compressed Graphics an instance of both real-time and stored streams	Part 9 Section 7.2	Support of stored streams is not a part of P9CGR	IDB
4	Carriage of Monomedia Streams in PES Packets - Specifies that Compressed Graphics is carried in PES packets	Part 9 Section 7.3		IDB
5	Packetization of DVB-defined Components - Specifies that the packetization of Compressed Graphics conforms to the DVB subtitling specification	Part 9 Section 7.3.2		IDB
6	Transport of Real-Time Streams	Part 9 Section 8.1		IDB

Table A4-26 — Function P9CGS Information Representation for Compressed Graphics stored streams

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Compressed Graphics - Specifies the use of DVB subtitling	Part 9 Section 6.12.2, 6.12.2.1		IDB
2	Types of Monomedia Components – Identifies Compressed Graphics as a stream or as a stand-alone component	Part 9 Section 7.1	Support of Compressed Graphics as a stand-alone component is not a part of P9CGS	IDB
3	Real-time and Stored Monomedia Streams – Identifies Compressed Graphics an instance of both real-time and stored streams	Part 9 Section 7.2	Support of real-time streams is not a part of P9CGS	IDB
4	Carriage of Monomedia Streams in PES Packets – Specifies that Compressed Graphics is carried in PES packets	Part 9 Section 7.3		IDB
5	Packetization of DVB-defined Components - Specifies that the packetization of Compressed Graphics conforms to the DVB subtitling specification	Part 9 Section 7.3.2		IDB
6	Transport of Stored Streams	Part 9 Section 8.2		IDB

Table A4-27 — Function P9CGB Information Representation for Compressed Graphics Bitmaps

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Compressed Graphics - Specifies the use of DVB subtitling and specifics how single bitmaps are coded.	Part 9 Section 6.12.2, 6.12.2.2		IDB
2	Types of Monomedia Components - Identifies Compressed Graphics as a stream or as a stand-alone Component	Part 9 Section 7.1	Support of Compressed Graphics as a stream is not a part of P9CGB	IDB
3	Transport of stand-alone Components	Part 9 Section 8, 8.3		IDB

A.4.4.3.3 Compressed Character Data Streams

DAVIC defines a format for compressed character data streams. These streams can be used as a stream which is played directly from the network and also as a stored stream which is played from memory under control of an application. The following tables: indicate the clauses from DAVIC Part 9 relevant to compressed character data streams graphics in the context of the *Enhanced Digital* Broadcast Contour for the functions identified above.

Table A4-28 — Function P9CCR Information Representation for Compressed Character Data Real-Time streams

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Compressed Character Data	Part 9 Section 6.13		IDB
2	Types of Monomedia Components – Identifies Compressed Character Data Graphics as a stream	Part 9 Section 7.1		IDB
3	Real-time and Stored Monomedia Streams - Identifies Compressed Character Data as an instance of both real-time and stored streams	Part 9 Section 7.2	Support of stored streams is not a part of P9CCR	IDB
4	Carriage of Monomedia Streams in PES Packets - Specifies that Compressed Graphics is carried in PES packets	Part 9 Section 7.3		IDB
5	Packetization of DAVIC-defined componetns	part 9 Section 7.3.2, Annex D		
6	Transport of Real-Time Streams	Part 9 Section 8.1		IDB

Table A4-29 — Function P9CCS Information Representation for Compressed Character stored streams

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Compressed Character Data	Part 9 Section 6.13		IDB
2	Types of Monomedia Components - Identifies Compressed Character Data Graphics as a stream	Part 9 Section 7.1		IDB
3	Real-time and Stored Monomedia Streams - Identifies Compressed Character Data as an instance of both real-time and stored streams	Part 9 Section 7.2	Support of real-time streams is not a part of P9CCS	IDB
4	Carriage of Monomedia Streams in PES Packets - Specifies that Compressed Graphics is carried in PES packets	Part 9 Section 7.3		IDB
5	Packetization of DAVIC-defined componetns	part 9 Section 7.3.2, Annex D		
6	Transport of Stored Streams	Part 9 Section 8.2		IDB

A.4.4.4 Linear Audio

Linear audio is intended as an alternative means to present audio information in addition and possibly simultaneous with the main audio channel. Applications include audible-feedback to user actions such as key presses, audible indication of a special situation (error condition, timed event).

Table A4-30 indicates the clauses from DAVIC Part 9 relevant to linear audio in the context of the *Enhanced Digital* Broadcast Contour.

Table A4-30 — Function P9LNAD: Information Representation for Stored Linear Audio streams

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Linear Audio - Specifies that Linear Audio is coded as AIFF-C	Part 9 Section 6.9		IDB
2	Types of Monomedia Components - Identifies Linear Audio as a monomedia component of the stream type	Part 9 Section 7.1		IDB
3	Real-time and Stored Monomedia Streams - Identifies linear audio as an instance of a stored stream	Part 9 Section 7.2		IDB
4	Carriage of Monomedia Streams in PES Packets - Specifies that Linear Audio is contained in PES packets	Part 9 Section 7.3		IDB
5	Packetization of DAVIC-defined Components	Part 9 Section 7.3.3		IDB
6	Transport of Stored Streams	Part 9 Section 8, 8.2		IDB
7	Coding of Linear Audio -Specifies the AIFF-C format	Annex B		IDB
8	Packetization of DAVIC defined Monomedia Components in PES packets - specifies details about how AIFF-C is contained in PES packets	Annex D		IDB

A.4.4.5 Text and Characters

DAVIC has selected a subset of the HTML 3.2 specification as a coding method for text with associated hyper links (corresponding to the MHEG-5 Hypertext Class anchor) . The coding of characters is based on Unicode to support multilingual text.

DAVIC has covered internationalization aspects by referencing work of the World Wide Web Consortium.

Detailed control of text presentation is provided by (Cascading) Style Sheets defined by the World Wide Web Consortium.

Table A4-31 indicates the clauses from DAVIC Part 9 relevant to text rendering issues in the context of the *Enhanced Digital* Broadcast Contour

Table A4-31 — Function P9TXT: Information Representation for Text and Characters

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Character Information - Specifies the use of Unicode	Part 9 Section 6.1		IDB
2	Text Information - Specifies the use of a subset of HTML 3.2 for text markup coding	Part 9 Section 6.2.x		IDB
3	Types of Monomedia Components - Identifies Text as a stand-alone component	Part 9 Section 7.1		IDB
3	Transport of stand-alone Components	Part 9 Section 8, 8.3		IDB

A.4.4.6 Service Information

Service Information data forms part of the bitstream to provide the user with information to assist in the selection of services and/or events within the bitstream. DAVIC has selected ETSI ETS 300 468 (DVB) as its specification for the representation for Service Information. Table A4-32 indicates the clauses from DAVIC Part 9 relevant to text rendering issues in the context of the *Enhanced Digital* Broadcast Contour

Table A4-32 — Function P9SI: Information Representation for Service Information

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Service Information- Specifies the use of DVB-SI for the coding of Service Information	Part 9 Section 6.5		IDB

A.4.4.7 Content Loading

DAVIC has defined tools for Content Packaging. A Structure as well as Metadata has been defined. *Table A*4-33 indicates the clauses from DAVIC Part 9 relevant to text rendering issues in the context of the *Enhanced Digital* Broadcast Contour

Table A4-33 — Function P9CLND: Information Representation for Content Loading

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Content Packaging and Metadata - Introduction	Part 9 Section 11		IDB
2	Content Package Structure	Part 9 Section 11.1.x		IDB
3	Content Metadata	Part 9 Section 11.2.x		IDB

A.4.4.8 Application Format

Background Information:

Applications pose widely varying requirements to the Interactive Engine. Broadly speaking, two language styles are currently in active use and therefore included in the *Enhanced Digital* Broadcast Contour as is the case in DAVIC:

1. A functional/declarative style. A declarative language consists of a collection of high-level explicitly specified functional objects. The possible interactions between these functional objects are also explicitly defined. Applications take the form of a interconnected set of objects. The functional style lends itself naturally to the implementation of services requiring navigation within a collection of “pages” containing information. Due to the high-level nature of the interface and the fact that there is no direct access from within applications to the infrastructure of the receiver applications reliable and safe execution is easily achieved.
2. A procedural style (Script/byte-code). Not all services can be expressed in the functional/declarative style. Examples are services that rely on extensive data manipulation and/or direct access to the receiver’s infrastructure. In this case a procedural paradigm is more appropriate. This requires a so-called byte-code interface such as JAVA.

Specifications for an interactive engine supporting both styles are given in DAVIC Part 9 and are detailed below.

MHEG-5 has been selected as the declarative engine this is specifically designed for TV-oriented navigation-like services. *Table A*4-34 indicates the clauses from DAVIC Part 9 and Part 5 relevant for the *Enhanced Digital* Broadcast Contour. This includes the mapping of MHEG-5 Elements to DSM-CC U-U which is the method used by the application to access its data either from the broadcast stream or from a point-to-point link (e.g., a telephone modem connection).

Table A4-34 — Function P9DAPP: Information Representation for Declarative Applications

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Application Interchange Format - Specifies the use of MHEG-5 for the coding of Declarative Applications	Part 9 Section 9.1	Only the first paragraph applies to the coding of Declarative Application	IDB
2	MHEG-5 profile for the DAVIC application domain - Specific details for the use of MHEG-5 in DAVIC	Part 9 Section 9.2.1, 9.2.2, 9.2.3, 9.2.4, 9.2.5, 9.2.6, 9.2.7, 9.2.8, 9.2.9, 9.2.10, 9.2.13		IDB
3	Mapping of MHEG-5 Elements to DSM-CC U-U - Specifies how MHEG-5 objects are retrieved using DSMCC	Part 9 Section 9.3.1, 9.3.2, 9.3.3, 9.3.4, 9.3.5		IDB
4	Reference Decoder Model- specifies the size and other relevant bounds to MHEG applications	Part 9 Section 10.1, 10.2, 10.3, 10.4		IDB
5	Run-time execution environment	Part 9 Section 9.6, 9.6.1	Implementation of the JAVA VM and packages is not part of P9DAPP	IDB
6	User Input Events Specifies how User Input (e.g., RC keys) is mapped onto application level events	Part 9 Section 9.6.2		IDB

MHEG-5 provides a mechanism, the InterchangedProgram Class, to execute procedural code from within a MHEG application. DAVIC has selected this procedural code to be JAVA VM. DAVIC has also specified a core set of JAVA APIs. *Table A*4-35 indicates the clauses from DAVIC Part 9 and MHEG-6 relevant for the *Enhanced Digital* Broadcast Contour.

Table A4-35 — Function P9PAPP: Information Representation for Procedural Applications

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Application Interchange Format - Specifies JAVA for the coding of procedural applications	Part 9 Section 9.1		IDB
2	Core set of Java APIs	Part 9 Section 9.4		IDB
3	java.lang - Minimal set of classes to run Java code	Part 9 Section 9.4.1		IDB
4	java.util - Utility features (strings, date handling, math)	Part 9 Section 9.4.2		IDB
5	java.io - access to non real time streams of data	Part 9 Section 9.4.3		IDB
6	iso.mheg5 - access to MHEG-5 objects	Part 9 Section 9.4.4		IDB
7	davic.dsmccuu -access to the DSMCC U-U interface for network data access	Part 9 Section 9.4.5	Support of this API is optional	IDB

A.4.4.9 Reference Decoder Model

DAVIC defines a Reference Decoding Model (RDM) which specifies semantic constraints on delivery, handling and decoding of content. The RDM does not describe any specific receiver architecture.

The RDM provides models for data delivery, memory usage for code and content objects and timing for object handling and instruction execution.

DAVIC application developers use the RDM as a virtual platform for application development and as the tool for verification of application correctness, without requiring any application testing at each specific STU implementation of the DAVIC specification.

*Table A*4-36 indicates the clauses from DAVIC Part 9 relevant to the reference decoding model in the context of the *Enhanced Digital* Broadcast Contour.

Table A4-36 — Function P9RDM Reference Decoder Model

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Scope - The RDM specifies semantic constraints on delivery, handling and decoding of content	Part 9 Section 10.1		IDB
2	Reference Decoding Model - Specification of the Reference Decoder Model	Part 9 Section 10.2		IDB
3	DAVIC Application Resource Descriptor - Descriptor which describes resources needed to execute a specific application on a receiver	Part 9 Section 10.3		IDB
4	Minimum DAVIC 1.2 STU requirements	Part 9 Section 10.4		IDB
5	Support for Graphics in STU	Part 9 Section 10.5		IDB
6	Persistent Memory	Part 9 Section 10.6		IDB

A.4.5 Security

DAVIC provides several kinds of tools for security. Two sets of tools deal with interfaces to detachable security devices. They are listed in *Table A*4-37 and *Table A*4-38 below. Also a set of tools, listed in *Table A*4-41 below, for the secure download (with respect to integrity, source and freshness of the information) of information to the STU has been defined.

Table A4-37 — Function P10CA0: Security Tool for Conditional Access based on PCCard Form Factor

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	S1 Scrambling - Describes the methodology of rendering information unintelligible to prevent use of protected material by unauthorised individuals	Part 10 Section 10.1		IDB
2	Security Interfaces - describes the approach adopted by DAVIC to put the vital security elements of the SCS on detachable units which can be easily exchanged	Part 10 Section 9		IDB
3	Security Interface CA0 - Specifies the use of the DVB-CI as a DAVIC Security Interface (based on the PCCard FormFactor)	Part 10 Section 10 and Section 13		IDB

Table A4-38 — Function P10CA1: Security Tool for Conditional Access based on SmartCard

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	S1 Scrambling - Describes the methodology of rendering information unintelligible to prevent use of protected material by unauthorised individuals	Part 10 Section 10.1		IDB
2	Security Interfaces - describes the approach adopted by DAVIC to put the vital security elements of the SCS on detachable units which can be easily exchanged	Part 10 Section 9		IDB
3	Security Interface CA1 - Specifies the DAVIC defined Security Interface CA1 (implemented on a ISO-7816) SmartCard	Part 10 Section 12.x		IDB

Table A4-39 — Function P10DL: Security Tool for Secure Download

#	Description	DAVIC Clause	Additional Implementation Information	Common with Contour
1	Secure Download - a mechanism for protecting software and data downloads to the STU. The integrity, source and freshness of the data is protected by this mechanism; however, confidentiality of the data is not provided.	Part 10 Section 7.6.x		IDB

Annex B
(normative)
Interactive Digital Broadcast (IDB) Technology Domain

B.1 Scope

This Appendix specifies the Technology Domain of the Interactive Digital Broadcast (IDB) contour. Figure B-1 shows a system configuration implementing the IDB contour.

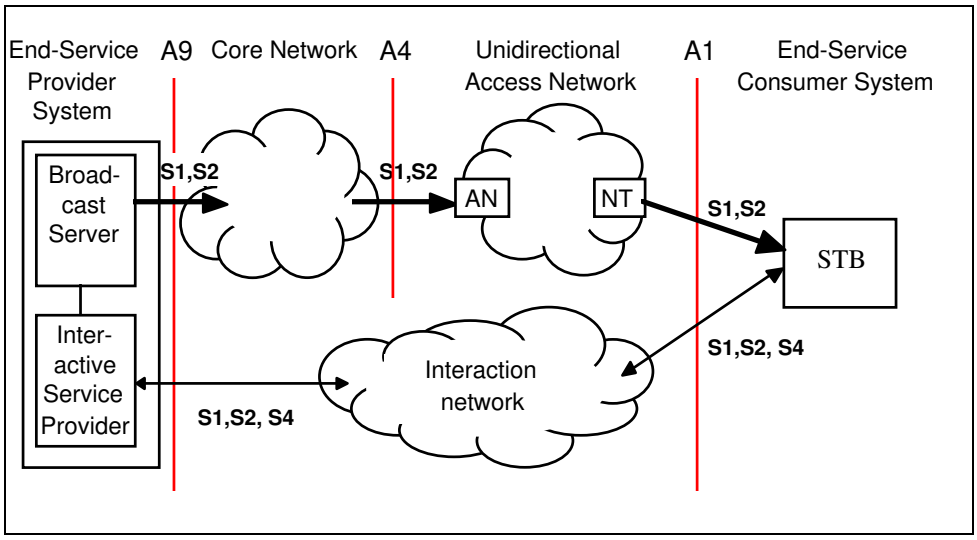

Figure B-1 — Configuration of Interactive Digital Broadcast System

B.2 Overview of system functions

Table B2-1, Table B2-2, Table B2-3, Table B2-4, Table B2-5 and Table B2-6 list the system functions relevant to the Interactive Digital Broadcast Contour. As described above, system functions are a collection of DAVIC clauses which together realize a complete system function. For each of the system functions it is indicated which DAVIC Clause implements the function, the Reference Point to which the function applies, the system location where the function is implemented and other system functions which shall be implemented if the function is implemented. An acronym is assigned to each function for further reference. This acronym is used in subclause B.4 of this Annex where detailed lists of DAVIC tools belonging to each function are given. Note that the functions defined in Table B2-5 and Table B2-6 are already sufficiently detailed so that these are not addressed in subclause B.4 of this Annex.

Note that these system functions have been defined on the basis of DAVIC 1.2 tools. Revisions of this technology contour may include new functions when suitable tools are identified according to the DAVIC Rules for technical procedure.