Doc. A/56
03 Jan 96

SYSTEM INFORMATION FOR DIGITAL TELEVISION

ATSC STANDARD

ADVANCED TELEVISION SYSTEMS COMMITTEE
James C. McKinney, Chairman
Dr. Robert Hopkins, Executive Director

Blank Page

SYSTEM INFORMATION FOR DIGITAL TELEVISION

ATSC STANDARD

Table of Contents

List of Figures

List of Tables

Blank Page

SYSTEM INFORMATION FOR DIGITAL TELEVISION

ATSC STANDARD

1. SCOPE

1.1 Purpose

This document defines a Standard for System Information (SI) compatible with digital multiplex bit streams constructed in accordance with ISO/IEC 13818-1 (MPEG-2). The document defines the standard protocol that carries relevant System Information tables contained within packets carried in the transport multiplex. The term *SI* will be used to refer to system-wide information in the Network Packet Identification (PID).

1.2 Application

This document describes tables that are applicable to terrestrial (over-the-air), cable, SMATV, MMDS, and satellite broadcast signals. Only certain messages are applicable in each of the domains. Messages for all domains are given in order to set a context for interoperability between the domains. All tables and messages defined in this document are carried in the Network PID.

1.2.1 Terrestrial Broadcast

At the option of terrestrial broadcasters, certain System Information may be transmitted in the transport stream. If included, the System Information shall be contained in PID 0x1FFC. Only NETWORK TEXT messages and VIRTUAL CHANNEL messages shall be included and shall be subject to the constraints in Sections 1.2.1.1 and 1.2.1.2.

1.2.1.1 Network Text Message

The following constraints apply:

- The transmission_medium field shall be over_the_air.
- The table_type value shall be MNT.

NOTE: The user’s attention is called to the possibility that compliance with this standard may require use of an invention covered by patent rights. By publication of this standard, no position is taken with respect to the validity of this claim, or of any patent rights in connection therewith. The patent holder has, however, filed a statement of willingness to grant a license under these rights on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license. Details may be obtained from the publisher.

1.2.1.2 Virtual Channel Message

The following constraints apply:

- The transmission_medium field shall be over_the_air.
- The channel_type field shall be normal.
- Inverse Channel Maps shall not be included.
- The activation_time field in the VCT_structure() shall be constrained to a value of zero, indicating that the data in the message shall be considered immediately valid (rather than at a future time.)
- The transmitted VCT shall, at minimum, define access for services carried in the transport stream carrying the table itself.

1.2.2 Other Media

Constraints applying to other media are under study.

1.3 Organization

The sections of this document are organized as follows:

- **Section 1** — Provides this general introduction.
- **Section 2** — Lists applicable documents.
- **Section 3** — Provides a list of acronyms and abbreviations used in this document.
- **Section 4** — Describes the basic structure of sections.
- **Section 5** — Describes formats of sections carried in the Network PID.
- **Section 6** — Describes multilingual character strings.
- **Annex A —** Discusses guidelines for interoperability among other media.
- **Annex B** — Provides an overview of tables defined in this System Information Standard.

2. REFERENCES

The following documents are applicable to this System Information Standard:

1. ATSC Standard A/52 (1995), Digital Audio Compression (AC-3).
2. ATSC Standard A/53 (1995), ATSC Digital Television Standard.
3. EBU Tech. 3264-E, Specification of the EBU Subtitling Data Exchange Format, European Broadcasting Union, February 1991.
4. ISO 639, Code for the Representation of Names of Languages, 1988.
5. ISO 639-2, Part 2 — Code for the Representation of Names of Languages, Committee Draft, dated 1993-01-06.
6. ISO 3166, Codes for the Representation of Names of Countries, 1988-08-15.
7. ISO/IEC 10646-1:1993, Information technology — Universal Multiple-Octet Coded Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane.
8. ISO/IEC 11172-1, Information Technology — Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s — Part 1: Systems.
9. ISO/IEC 11172-2, Information Technology — Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s — Part 2: Video.
10. ISO/IEC 11172-3, Information Technology — Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s — Part 3: Audio.
11. ISO/IEC 13818-3:1994, Information Technology — Coding of moving pictures and associated audio — Part 3: Audio.
12. ISO/CD 13522-2; 1993, Information Technology — Coded representation of multimedia and hypermedia information objects — Part 1: Base notation.
13. ISO/IEC 8859**,** Information Processing — 8-bit Single-Octet Coded Character Sets, Parts 1 through 10.
14. ITU-T Rec. H. 222.0 | ISO/IEC 13818-1:1994, Information Technology — Coding of moving pictures and associated audio — Part 1: Systems.
15. ITU-T Rec. H. 262 | ISO/IEC 13818-2:1994, Information Technology — Coding of moving pictures and associated audio — Part 2: Video.

3. DEFINITIONS

3.1 Compliance Notation

As used in this document, *"shall"* or *"will"* denotes a mandatory provision of the standard. *"Should"* denotes a provision that is recommended but not mandatory. *"May"* denotes a feature whose presence does not preclude compliance, that may or may not be present at the option of the implementer.

3.2 Definition of Terms

The following terms are used throughout this document:

section: A data structure comprising a portion of an *ISO/IEC 13818-1*-defined table, such as the Program Association Table (PAT), Conditional Access Table (CAT), or Program Map Table (PMT). The term conforms to MPEG terminology. All sections begin with the table_ID and end with the CRC_32 field. Sections are carried in non-PES[1] streams; their starting points within a packet payload are indicated by the pointer_field mechanism defined in the *ISO/IEC 13818-1 Systems* document.

message*:* The more general term *message* is used interchangeably with *section*, especially to refer to non-table-oriented data structures such as, for example, the SYSTEM TIME message. Likewise, the term *message* is used to refer to a data structure that may deliver portions of various types of tables. The NETWORK INFORMATION message, for example, defines portions of several types of network tables.

program element: A generic term for one of the elementary streams or other data streams that may be included in a program.

program: A collection of program elements. Program elements may be elementary streams. Program elements need not have any defined time base; those that do have a common time base and are intended for synchronized presentation. The term *program* is also used in the context of a "television program" such as a scheduled daily news broadcast. The distinction between the two usages should be understood by context.

service*: ISO/IEC 13818-1* uses the term *program* to refer to a collection of program elements without regard to time. In this System Information Standard, the term *service* is used in this same context to denote a collection of elementary components. Usage of the term *service* clarifies certain discussions that also involve the notion of the term *program* in its traditional meaning — in, for example, the statement, "A video service carries a series of programs."

stream: An ordered series of bytes. The usual context for the term *stream* involves specification of a particular PID (such as the "Program Map PID stream"), in which case

[1] Packetized Elementary Stream.

the term indicates a series of bytes extracted from the packet multiplex from packets with the indicated PID value.

3.3 Acronyms and Abbreviations

The following acronyms and abbreviations are used within this specification:

bslbf	bit serial, leftmost bit first
CAT	Conditional Access Table
CDT	Carrier Definition Table
CRC	cyclic redundancy check
ECM	Entitlement Control Message
EMM	Entitlement Management Message
GA	Grand Alliance
GMT	Greenwich Mean Time
GPS	Global Positioning System
IRD	Integrated Receiver-Decoder
MCPT	Multiple Carriers per Transponder
MMT	Modulation Mode Table
MPEG	Moving Picture Experts Group
NVOD	Near Video On Demand
PAT	Program Association Table
PCR	Program Clock Reference
PES	Packetized Elementary Stream
PID	Packet Identification
PMT	Program Map Table
PTS	Presentation Time Stamp
rpchof	remainder polynomial coefficients, highest order first
SECAM	Sequential Couleur Avec Memoire
SIT	Satellite Information Table
TAI	International Atomic Time[2]
TDT	Transponder Data Table
TNT	Transponder Name Table
TS	Transport Stream
UTC	Universal Coordinated Time[2]
uimsbf	unsigned integer, most significant bit first
VCT	Virtual Channel Table
VCN	Virtual Channel Number

[2] Reversal of acronym letters is due to the translation from the French.

3.4 Section and Data Structure Syntax Notation

This document contains symbolic references to syntactic elements. These references are typographically distinguished by the use of a different font (e.g., restricted), may contain the underscore character (e.g., sequence_end_code) and may consist of character strings that are not English words (e.g., dynrng).

The formats of sections and data structures in this document are described using a C-like notational method employed in *ISO/IEC 13818-1*. Extensions to this method are described in the following sections.

3.4.1 Field Sizes

Each data structure is described in a table format wherein the size in bits of each variable within that section is listed in a column labeled "Bits." The column adjacent to the bits column is labeled "Bytes" and indicates the size of the item in bytes. For convenience, several bits within a particular byte or multi-byte variable may be aggregated for the count. An example follows:

	Bits	Bytes	Bit Number / Description
foo_section(){			
section_syntax_indicator	1	1	
...			
if (section_syntax_indicator) {			
table_extension	16	(2)	uimsbf
ISO_reserved	2	(1)	bslbf
version_number	5		uimsbf
current_next_indicator	1		bslbf {next, current}
...			
}			
...			

In the byte count column, items that are conditional (because they are within a loop or conditional statement) are parenthesized. Nested parentheses are used if the loops or conditions are nested.

4. MESSAGE STRUCTURE

This section describes details of message structure and transport of Network tables within the MPEG-2 multiplex.

4.1 Framing and Synchronization

Tables and messages defined in this System Information Standard are structured in the same manner used for carrying *ISO/IEC 13818-1* -defined PSI tables. The MPEG-defined 32-bit CRC is required.

4.2 Table ID Ranges and Values

Table 4.1 defines table_ID ranges and values.

Table 4.1 Table ID Ranges and Values

Table ID Value (hex)	Sections	Stream	Ref.
	ISO/IEC 13818-1 Sections:		
0x00	PROGRAM ASSOCIATION	0	Ref [14]
0x01	CONDITIONAL ACCESS	1	Ref [14]
0x02	TS PROGRAM MAP	Prog. Map	Ref [14]
0x03-0x3F	[ISO Reserved]		
	User Private Sections:		
0x40-0xBF	[User Private]		
	System Information Sections:		
0xC0-0xC1	[Reserved]	-	-
0xC2	NETWORK INFORMATION	Network	Sec. 5.1
0xC3	NETWORK TEXT	Network	Sec. 5.2
0xC4	VIRTUAL CHANNEL	Network	Sec. 5.3
0xC5	SYSTEM TIME	Network	Sec. 5.4
0xC6-0xFE	[Reserved]	-	-
0xFF	**Inter-message Filler**		

Messages defined in this System Information Standard, and any created as user extensions to it are considered “private” with respect to *ISO/IEC 13818-1*. Message types 0x40 through 0xBF are user defined (outside the scope of this System Information Standard).

The maximum total length of any message section carried in the Network PID is 1024 bytes. This total includes table_ID, CRC, and everything in between.

4.3 Extensibility

This System Information Standard describes a number of tables and messages delivered in the Network PID. The System Information Standard is designed to be extensible via the following mechanisms:

1. **Reserved Fields:** Fields in this System Information Standard marked reserved are reserved for use either when revising this System Information Standard, or when another standard is issued that builds upon this one. See Section 4.4 below.
2. **Standard Table Types:** As indicated in Table 4.1, table_ID values in the range 0xC0, 0xC1, and 0xC6 through 0xFE are reserved for use either when revising this System Information Standard, or when another standard is issued that builds upon this one.
3. **User Private Table Types:** As indicated in Table 4.1, table_id values in the range 0x40 through 0xBF are reserved for "user private" use. The format of user private tables carried in the Network PID shall conform to the syntax described in Figure 4.1.
4. **User Private Descriptors:** Privately defined descriptors may be placed at designated locations throughout the messages described in this System Information Standard. Ownership of one or more user private descriptors is indicated by the presence of an MPEG registration_descriptor() preceding the descriptor(s).

	Bits	Bytes	Bit Number / Description
network_private_message(){			
table_ID	8	1	uimsbf (0x40 <= table_ID <= 0xBF)
section_syntax_indicator	1		bslbf
zero	1		bslbf
ISO_reserved	2		bslbf
section_length	12	2	uimsbf
if (section_syntax_indicator==1) {			
table_extension	16	(2)	uimsbf
ISO_reserved	2	(1)	bslbf
version_number	5		uimsbf
current_next_indicator	1		bslbf {next, current}
section_number	8	(1)	uimsbf
last_section_number	8	(1)	uimsbf
}			
zero	3	1	bslbf
protocol_version	5		see Section 4.4.1
format_identifier	32	4	uimsbf
private_message_body()	N*8	N	
CRC_32	32		rpchof
}			

Figure 4.1. Network private message format.

4.4 Reserved Fields

reserved — Fields in this System Information Standard marked "reserved" shall not be assigned by the user, but shall be available for future use. Decoders shall disregard reserved fields for which no definition exists that is known to that unit. Fields marked "reserved" shall be set to a value of zero until such time as they are defined and supported.

ISO_reserved — Fields in this System Information Standard marked "ISO_reserved" are reserved under *ISO/IEC 13818-1* and hence shall not be assigned by the user. Unless otherwise specified within the *ISO/IEC 13818-1* standard, all ISO reserved bits shall be set to '1.'

user_private — Indicates the bit or bit field is not defined within the scope of this System Information Standard. The owner of the bit, and hence the entity defining its meaning, is derived via its context within a message.

zero — Indicates the bit or bit field shall be the value zero.

4.5 Network Private Message Syntax

Figure 4.1 defines the syntax for user private sections and messages carried in the Network PID. The MPEG-defined CRC is required. Please refer to *ISO/IEC 13818-1* for definition of MPEG-standard fields.

4.5.1 Protocol Version

protocol_version — A 5-bit unsigned integer field whose function is to allow, in the future, any defined table type to carry parameters that may be structured fundamentally differently than those defined in the current protocol. At present, all defined message types in this protocol are defined for protocol_version zero only. Nonzero values of protocol_version may only be processed by Decoders designed to accommodate the later versions as they become standardized.

4.5.2 Format Identifier

format_identifier — A 32-bit unsigned integer value which unambiguously identifies the entity defining this network_private_message() syntax. Values for format_identifiers shall be obtained from ATSC.

5. MESSAGE FORMATS — NETWORK PID

The following sections define the formats of messages as they appear within the Network PID of the Transport Stream.

5.1 Network Information Message

The NETWORK INFORMATION message is carried in the Network PID Stream, and delivers sections of nontextual tables applicable system-wide. The tables include:

- Carrier Definition Tables (CDTs) for each transmission medium (cable, satellite, over the air, MMDS and SMATV)
- Modulation Mode Tables (MMTs) for each transmission medium
- Satellite Information Table (SIT)
- Transponder Data Table (TDT)

The message format consists of

- The table index pointing to the first record in the table to be defined in this message
- The number of records being defined in this message
- The table type (CDT, MMT, and so on)
- If the table is a TDT type, the satellite ID

Figure 5.1 shows the format of the NETWORK INFORMATION message.

5.1.1 Table ID

The table_ID of the NETWORK INFORMATION message is 0xC2.

5.1.2 Common Data

Certain data is common to all types of data delivered in the NETWORK INFORMATION message.

first_index — An 8-bit unsigned integer number in the range one to 255 that indicates the index of the first record to be defined in this message. If more than one record is provided, the additional records define successive table entries following first_index. The value zero is illegal and shall not be specified.

number_of_records — An 8-bit unsigned integer number that defines the number of records being defined in this message. The minimum allowed value is one. The maximum is limited by the maximum allowed length of the message.

transmission_medium — A 4-bit field that defines the transmission medium for which the data in this NETWORK INFORMATION message applies. Table 5.1 defines the coding.

	Bits	Bytes	Bit Number / Description
network_info_message(){			
table_ID	8	1	uimsbf value 0xC2
zero	2	2	bslbf
ISO_reserved	2		bslbf
section_length	12		uimsbf
zero	3	1	
protocol_version	5		Sec. 4.4.1
first_index	8	1	uimsbf range 1-255
number_of_records	8	1	uimsbf
transmission_medium	4		uimsbf see Table 5.1
table_type	4		uimsbf see Table 5.2
if (table_type==TDT) {			
satellite_ID	8	(1)	uimsbf range 0-255
}			
for (i=0; i<number_of_records; i++) {			
if (table_type==CDT) {			
CDT_record()		((5))	
}			
if (table_type==MMT) {			
MMT_record()		((6))	
}			
if (table_type==SIT) {			
SIT_record()		((4))	
}			
if (table_type==TDT) {			
TDT_record()		((6))	
}			
descriptors_count	8	(1)	uimsbf range 0-255
for (i=0; i<descriptors_count; i++) {			
descriptor()	*	((*))	optional
}			
}			
for (i=0; i<N; i++) {			
descriptor()	*	(*)	optional
}			
CRC_32	32	4	rpchof
}			

Figure 5.1. Network information message format.

Table 5.1 Transmission Medium

transmission_medium	meaning
0	cable
1	satellite
2	MMDS
3	SMATV
4	over_the_air
5-15	reserved

A NETWORK INFORMATION message received with transmission_medium indicating an unknown or unsupported medium shall be discarded.

table_type — A 4-bit value that defines the type of table delivered in the message. One instance of a NETWORK INFORMATION message can define entries within at most one type of table. The table_type parameter is defined in Table 5.2.

Table 5.2 Table Type

table_type	meaning
0	invalid
1	**CDT** — Carrier Definition Table
2	**MMT** — Modulation Mode Table
3	**SIT** — Satellite Information Table
4	**TDT** — Transponder Data Table
5-15	reserved

A value of zero for table_type is undefined, and may be used to indicate *unknown* or *invalid*.

A NETWORK INFORMATION message received with table_type indicating an unknown or unsupported table type shall be discarded.

5.1.2.1 Satellite ID

The satellite_ID is included in the message if the table_type is TDT.

satellite_ID — An 8-bit unsigned integer value in the range zero to 255 that identifies the satellite to which data in this message applies.

5.1.2.2 Descriptors Count

descriptors_count — An 8-bit unsigned integer value in the range zero to 255 representing the number of descriptor blocks to follow.

5.1.3 Carrier Definition Table (CDT)

Figure 5.2 defines the structure of the CDT_record(). The CDT defines sets of carrier frequencies to support cable, satellite, MMDS, and SMATV transmission media. Separate tables for each of the transmission media may be present in network data; if that is the case, the Decoder shall discard the inapplicable table or tables. A full frequency plan table for one medium is constructed from one or more CDT_record() structures, each defining a starting frequency, a number of carriers, and a frequency spacing for carriers in this group.

The specified carrier represents the nominal center of the spectral band for all modulation methods, including analog. Carrier frequencies in the table thus represent the data carrier frequency for digital transmissions modulated using QAM or PSK.[3]

[3] Note that transmission systems using VSB modulation transmit spectra that are not symmetrical about the carrier or pilot tone. Acquisition of a VSB-modulated signal involves computation of the pilot tone (or in analog VSB, the picture carrier) location relative to the center of the band. For example, for the ATSC

(continued on next page)

Each CDT_record represents a definition of N carriers. The first_index parameter reflects the index of the first carrier in the group. If the message includes more than one CDT_record(), the carrier index of the second group would be first_carrier plus the number of carriers defined in the first group. References to the Carrier Definition Table, such as the CDT_reference in the TDT_record(), are to the carrier's index (a carrier defined within a CDT_record()), *not* to the index of a CDT_record() itself.

Note that the carriers, as defined by one or more CDT_record()s, may or may not end up sorted in order of increasing carrier frequency. Certain frequency plans may be specified by overlapping two or more CDT_record()s, each of which defines equally-spaced carriers.

Note also that carriers may be defined that are currently not in use. To facilitate the compressed delivery format, carriers may be defined that do not reflect reality. An example: carriers at 1, 2, 4, 5, 7, 8 MHz could be defined as eight carriers at 1MHz spacing (3 MHz and 6 MHz do not really exist, or are not currently in use).

	Bits	Bytes	Bit Number / Description
CDT_record(){			
number_of_carriers	8	1	uimsbf
spacing_unit	1	2	bslbf see Table 5.3
reserved	1		bslbf reserved
frequency_spacing	14		uimsbf range 1-16,383 units of 10 or 125kHz
frequency_unit	1	2	bslbf see Table 5.4
first_carrier_frequency	15		uimsbf range 0-32,767 units of 10 or 125kHz
}			

Figure 5.2. CDT record format.

number_of_carriers — An unsigned integer in the range one to 255 that represents the number of carriers whose frequency is being defined by this CDT_record().

spacing_unit — A 1-bit field identifying the units for the frequency_spacing field. Table 5.3 defines the coding for spacing_unit.

Table 5.3 Spacing Unit

spacing_unit	meaning
0	10 kHz spacing
1	125 kHz spacing

(continued from previous page)

Digital Television Standard (Ref. [2]), where the channel bandwidth is 6 MHz, the pilot tone is located 310 kHz above the lower edge of the channel, or 2.690 MHz below the specified center of the band. Similarly, for analog NTSC, the picture carrier is 1.25 MHz above the lower edge of the channel, or 1.75 MHz below the specified center of the band.

frequency_spacing — A 14-bit unsigned integer number in the range one to 16,383 that defines the frequency spacing in units of either 10 kHz or 125 kHz, depending upon the value of the spacing_unit parameter. If spacing_unit is zero, indicating 10 kHz, then a value of one indicates 10 kHz spacing, two indicates 20 kHz, and so on. If the number_of_carriers field is one, the frequency_spacing field is ignored. The maximum frequency that can be represented is ((2^14) -1) * 125 kHz = 2048.875 MHz. The minimum frequency spacing is 10 kHz.

frequency_unit — A 1-bit field identifying the units for the first_carrier_frequency field. Table 5.4 defines the coding for frequency_unit.

Table 5.4 Frequency Unit

frequency_unit	meaning
0	10 kHz units
1	125 kHz units

first_carrier_frequency — A 15-bit unsigned integer number in the range zero to 32,767 that defines the starting carrier frequency for the carriers defined in this group, in units of either 10 kHz or 125 kHz, depending on the value of frequency_unit. If only one carrier is defined for the group, the first_carrier_frequency represents its frequency. When the frequency_unit indicates 125 kHz, the first_carrier_frequency can be interpreted as a fractional frequency (1/8 MHz) in the least-significant 3 bits, and an integer number of megahertz in the upper 12 bits. The range of frequencies that can be represented is zero to ((2^15) -1) * 125 kHz = 4095.875 MHz.

For satellite use, carrier frequencies specified in the CDT are defined relative to the point in the receiver following block conversion of the downlink signal — i.e., at the L-band input to the receiver. The Local Oscillator (L.O.) frequency offset is defined by the frequency_band parameter defined in the Satellite Information Table for each satellite. In addition, a convention for synthesis of the tuned frequency is assumed (either high side or low side), again based on frequency band. Table 5.5 defines the L.O. frequency and conversion method for each band.

To convert a frequency specified in the CDT to the corresponding downlink frequency, for the high-side conversion method, the CDT frequency is subtracted from the L.O. frequency appropriate to the satellite. For low-side conversion, the CDT frequency is added to the L.O. frequency.

Table 5.5 Local Oscillator Frequencies vs. Frequency Band

Frequency Band	L. O. Frequency	Conversion
C Band	5.150 GHz	High side
Ku (FSS)	10.750 GHz	Low side
Ku (BSS)	11.250 GHz	Low side

5.1.4 Modulation Mode Table (MMT)

Figure 5.3 defines the structure of the MMT_record().

	Bits	Bytes	Bit Number / Description
MMT_record(){			
transmission_system	4	1	uimsbf see Table 5.6
inner_coding_mode	4		uimsbf see Table 5.7
split_bitstream_mode	1	1	bslbf {no, yes}
reserved	2		bslbf reserved
modulation_format	5		uimsbf see Table 5.8
reserved	4	4	bslbf reserved
symbol_rate	28		uimsbf units: one symbol per sec.
}			

Figure 5.3. MMT record format.

5.1.4.1 Transmission System

transmission_system — A 4-bit field that identifies the transmission standard employed for the waveform defined by this MMT record. Table 5.6 defines the coding for transmission_system.

Table 5.6 Transmission System

transmission_system	meaning
0	**unknown** — The transmission system is not known.
1	**ITU_T_annex_1** — The transmission system conforms to the ITU ETSI standard covering cable distribution, Annex 1.
2	**ITU_T_annex_2** — The transmission system conforms to the ITU North American standard covering cable distribution, Annex 2.
3	**ITU_R** — The transmission system conforms to the ITU ETSI standard covering satellite distribution.
4	**ATSC** — The transmission system conforms to the ATSC Digital Television Standard (Ref. [2]).
5	**DigiCipher** — The transmission system conforms to the General Instrument DigiCipher® II System for satellite distribution of compressed audio and video.
6-15	reserved

5.1.4.2 Inner Coding Mode

inner_coding_mode — A 4-bit field that indicates the coding mode for the inner code associated with the waveform described in this MMT record. The following values are currently defined: 5/11, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, and 7/8. Coding of the inner_coding_mode field is shown in Table 5.7.

5.1.4.3 Modulation Format

modulation_format — A 5-bit field that defines the basic modulation format for the carrier. Table 5.8 defines the parameter.

Table 5.7 Inner Coding Mode

inner_coding_mode	meaning
0	rate 5/11 coding
1	rate 1/2 coding
2	reserved
3	rate 3/5 coding
4	reserved
5	rate 2/3 coding
6	reserved
7	rate 3/4 coding
8	rate 4/5 coding
9	rate 5/6 coding
10	reserved
11	rate 7/8 coding
12-14	reserved
15	none — indicates that the waveform does not use concatenated coding

Table 5.8 Modulation Format

modulation_format	meaning
0	**unknown** — The modulation format is unknown.
1	**QPSK** — The modulation format is QPSK (Quadrature Phase Shift Keying).
2	**BPSK** — The modulation format is BPSK (Binary Phase Shift Keying).
3	**OQPSK** — The modulation format is offset QPSK.
4	**VSB_8** — The modulation format is 8-level VSB (Vestigial Sideband).
5	**VSB 16** — The modulation format is 16-level VSB.
6	**QAM 16** — The modulation format is 16-level Quadrature Amplitude Modulation (QAM).
7	**QAM 32** — 32-level QAM
8	**QAM 64** — 64-level QAM
9	**QAM 100** — 100-level QAM
10	**QAM 128** — 128-level QAM
11	**QAM 144** — 144-level QAM
12	**QAM 196** — 196-level QAM
13	**QAM 256** — 256-level QAM
14	**QAM 400** — 400-level QAM
15	**QAM 512** — 512-level QAM
16	**QAM 576** — 576-level QAM
17	**QAM 784** — 784-level QAM
18	**QAM 1024** — 1024-level QAM
19-31	reserved

5.1.4.4 Symbol Rate

symbol_rate — A 28-bit unsigned integer field that indicates the symbol rate in units of one symbol per second associated with the waveform described in this MMT record.

5.1.5 Satellite Information Table (SIT)

Figure 5.4 defines the structure of the SIT_record().

	Bits	Bytes	Bit Number / Description
SIT_record(){			
satellite_ID	8	1	uimsbf
you_are_here	1	2	bslbf {no, yes}
frequency_band	2		uimsbf see Table 5.9
out_of_service	1		bslbf {no, yes}
hemisphere	1		bslbf see Table 5.10
orbital_position	11		uimsbf units: 0.1 deg.
polarization_type	1	1	bslbf see Table 5.11
reserved	1		bslbf reserved
number_of_transponders	6		uimsbf range 0-63 (1-count)
}			

Figure 5.4. SIT record format.

satellite_ID — An 8-bit unsigned integer number in the range zero to 255 that identifies the satellite associated with the data defined in this record.

you_are_here — A Boolean flag that indicates, when set, that the satellite_ID field reflects the satellite carrying the Transport Stream that delivered this NETWORK INFORMATION message. The you_are_here flag may be used to verify dish alignment during installation of a C-band or Ku-band IRD having a movable dish. All NETWORK INFORMATION messages having you_are_here bits set true must be locally originated at an Encoder, rather than being originated at a central site and carried in the Network stream.

frequency_band — A 2-bit field that indicates the frequency band associated with the CDT record. Table 5.9 defines the coding.

Table 5.9 Frequency Band

frequency_band	meaning
0	C Band
1	Ku Band (FSS)
2	Ku Band (BSS)
3	reserved

Refer to Table 5.5 for a description of how frequency_band is used to specify downlink frequencies for the satellite CDT.

out_of_service — A Boolean flag that, when set, indicates that the satellite given by satellite_ID is permanently out of service; i.e., it has failed or has been retired without replacement. When the flag is false, the satellite is currently in (at least partial) operation.

The out_of_service flag may be used as a signal to delete satellite and transponder records from Decoder memory.

hemisphere—A 1-bit field indicating whether the satellite given by satellite_ID resides in the western (value zero) or eastern (value one) hemisphere. The coding for hemisphere is given in Table 5.10.

Table 5.10 Hemisphere

hemisphere	meaning
0	Western hemisphere
1	Eastern hemisphere

orbital_position — An 11-bit unsigned integer number in the range zero to 1800 representing the orbital position of the satellite given by satellite_ID, in units of 0.1° longitude. The longitudinal coordinates provided in the table are West longitude for products fielded in the western hemisphere, and East longitude for products fielded in the eastern hemisphere.

polarization_type — A one-bit field that indicates whether the satellite uses a linear polarization method (horizontal/vertical planes), or a circular method (left/right circular polarization). Table 5.11 defines polarization_type.

Table 5.11 Polarization Type

polarization_type	meaning
0	Linear polarization
1	Circular polarization

number_of_transponders — A 6-bit unsigned integer number in the range zero to 63 representing one less than the number of transponders (carriers) associated with the satellite being defined. A value of 23 in the field indicates that the satellite is associated with 24 transponders (carriers), for example.

5.1.6 Transponder Data Table (TDT)

Figure 5.5 defines the structure of the TDT_record().

transport_type — A 1-bit field identifying the type of transport carried on this transponder as either being an MPEG-2 transport (value zero), or not (value one). Table 5.12 defines the coding.

Table 5.12 Transport Type

transport_type	meaning
0	MPEG-2 transport
1	non-MPEG-2 transport

	Bits	Bytes	Bit Number / Description
TDT_record(){			
transport_type	1	1	bslbf see Table 5.12
polarization	1		bslbf see Table 5.13
transponder_number	6		uimsbf range 0-63
CDT_reference	8	1	uimsbf range 1-255
if (transport_type==MPEG_2) {			
MMT_reference	8	(1)	uimsbf range 1-255
VCT_ID	16	(2)	uimsbf range 0-0xFFFF
root_transponder	1	(1)	bslbf {no, yes}
reserved	7		bslbf reserved
} else { /* non-MPEG_2 */			
wide_bandwidth_video	1	(1)	bslbf {no, yes}
reserved	2		bslbf reserved
waveform_standard	5		uimsbf see Table 5.14
audio_mode()	24	(3)	
}			
}			

Figure 5.5. TDT record format.

polarization — A 1-bit field identifying the polarization of the carrier associated with a satellite transponder. A value of zero indicates horizontal (or left) polarization; a value of one indicates vertical (or right) polarization. The polarization method used (linear or circular) is given in the SIT record for the satellite as polarization_type. Table 5.13 defines polarization.

Table 5.13 Polarization

polarization	meaning
0	Horizontal (for linearly polarized satellites) or Left (for circularly polarized satellites)
1	Vertical (for linearly polarized satellites) or Right (for circularly polarized satellites)

transponder_number — An unsigned 6-bit integer value in the range zero to 63 that indicates which transponder is being defined by this TDT_record(). A value of zero indicates the first transponder, and so on.

NOTE: Numbering in the TDT may or may not correspond to conventional transponder numbering, that for typical C-band satellites (for example) are numbered one to 24.

CDT_reference — An unsigned 8-bit integer value in the range one to 255 that associates the transponder being defined with a particular carrier frequency via this reference into the Carrier Definition Table. The value zero is illegal and shall not be specified. CDT_reference is used as an index value into the CDT.

5.1.6.1 Standard MPEG-2 Transports

MMT_reference — An 8-bit unsigned integer value in the range one to 255 that references an entry into the Modulation Mode Table (MMT) for satellite transmission medium. For digital waveforms, the MMT_reference associates the transponder/carrier with a digital modulation mode. The value zero is illegal and shall not be specified. MMT_reference is used as an index value into the MMT.

VCT_ID — A 16-bit unsigned integer number in the range zero to 0xFFFF that indicates the Virtual Channel Table ID associated with this transponder.

root_transponder — A Boolean flag that indicates, when set, that the transponder is a root transponder for the Virtual Channel Table indicated in the VCT_ID parameter. When the flag is clear, the transponder is not a root transponder.

5.1.6.2 Other Transport Types

wide_bandwidth_video — A Boolean flag that indicates, when set, that the video is modulated at 10.75 MHz peak deviation. When the flag is clear, the video is modulated in "narrow" format, or 8.2 MHz peak deviation.

waveform_standard — A 5-bit field that identifies the waveform standard associated with the transponder (carrier). Table 5.14 defines waveform_standard.

Table 5.14 Waveform Standard

waveform_standard	meaning
0	**unknown** — The waveform type is unknown or undefined at this time.
1	**NTSC** — The waveform is standard NTSC
2	**PAL 625** — The waveform is standard 625-line PAL
3	**PAL 525** — The waveform is standard 525-line PAL
4	**SECAM** — The waveform is standard SECAM
5	**D2-MAC** — The waveform is D2-MAC
6	**B-MAC** — The waveform is B-MAC
7	**C-MAC** — The waveform is C-MAC
8	**DCI** — The waveform conforms to the General Instrument DigiCipher® I scrambling standard
9	**VideoCipher** — The waveform conforms to the General Instrument VideoCipher® scrambling standard
10	**RCA DSS** — The waveform conforms to the RCA DSS system
11	**Orion** — The waveform is scrambled using the TvCom (formerly Oak) Orion system
12	**Leitch** — The waveform is scrambled using the Leitch system
13-31	reserved

5.1.6.3 Audio Mode

Figure 5.6 describes the format of the audio_mode() structure.

	Bits	Bytes	Bit Number / Description
audio_mode(){			
wide_bandwidth_audio	1	3	bslbf see Table 5.15
companded_audio	1		bslbf {no, yes}
matrix_mode	2		uimsbf see Table 5.16
subcarrier_2_offset	10		uimsbf range 0-1023, units of 0.0lMHz above 5.0 MHz
subcarrier_1_offset	10		uimsbf range 0-1023, units of 0.0lMHz above 5.0 MHz
}			

Figure 5.6. Audio mode structure format.

wide_bandwidth_audio — A 1-bit field indicating whether the audio subcarrier(s) defined for this transponder/carrier are modulated in narrow bandwidth (value zero) or wide bandwidth (value one) format. Table 5.15 defines the coding.

Table 5.15 Wide Bandwidth Audio

wide_bandwidth_audio	meaning
0	narrow bandwidth
1	wide bandwidth

companded_audio — A Boolean flag indicating whether audio is transmitted in companded format on this transponder/carrier. A value of zero indicates not companded; a value of one indicates companded audio.

matrix_mode — A 2-bit field that indicates the matrix mode associated with analog audio subcarriers on this transponder/carrier. The field is coded as shown in Table 5.16.

Table 5.16 Matrix Mode

matrix_mode	meaning
0	**mono** — Indicates that subcarrier 1 carries the L+R (mono) audio channel. Subcarrier 2, if present (specified as a different frequency than subcarrier 1), then subcarrier 2 carries cue tones or other audio.
1	**discrete_stereo** — Indicates that subcarrier 1 carries the left audio channel, and subcarrier 2 carries the right audio channel.
2	**matrix_stereo** — Indicates that subcarrier 1 carries the L+R (sum) vector and subcarrier 2 carries the L-R (difference) vector.
3	**reserved**

subcarrier_2_offset — A 10-bit unsigned integer field in the range zero to 1023 that defines the audio subcarrier offset for the second audio subcarrier (if any). The value is given in units of 0.01 MHz above 5.0 MHz. A value of one, for example, indicates a subcarrier offset of 5.01 MHz. If no second subcarrier exists, the value of subcarrier_2_offset is set to equal subcarrier_1_offset.

subcarrier_1_offset — A 10-bit unsigned integer field in the range zero to 1023 that defines the audio subcarrier offset for the primary audio subcarrier. The value is given in units of 0.01 MHz above 5.0 MHz.

5.1.7 Message-End Descriptors

The message may include at its end one or more structures of the form tag, length, data. The number of descriptors present is determined indirectly by processing the section_length field.

5.2 Network Text Message

The NETWORK TEXT message is carried in the network data stream, and delivers sections of textual tables applicable system-wide. The NETWORK TEXT message is defined multilingually. The defined table types include:

- Transponder Name Table (TNT)
- Satellite Text Table (STT)
- Rating Text Table (RTT)
- Rating System Table (RST)
- Currency System Table (CST)
- Source Name Table (SNT)
- Map Name Table (MNT)

Figure 5.7 shows the format of the NETWORK TEXT message.

5.2.1 Table ID

The table_ID of the NETWORK TEXT message is 0xC3.

5.2.2 Multilingual Text

The NETWORK TEXT message carries multilingual text strings, formatted as defined in Section 6. Text strings included in the NETWORK TEXT message shall not include format effectors (defined in Section 6.3). If format effectors are present in a text block, the decoder shall disregard them.

5.2.3 Transmission Medium

transmission_medium — A 4-bit field that defines the transmission medium for which the data in this NETWORK TEXT message applies. Table 5.17 defines transmission_medium.

5.2.4 Table Type

table_type — A 4-bit value that defines the type of table delivered in the message. One instance of a NETWORK TEXT message can define entries within at most one type of table. The table_type parameter is defined in Table 5.18.

	Bits	Bytes	Bit Number / Description
network_text_message(){			
table_ID	8	1	uimsbf value 0xC3
zero	2		bslbf
ISO_reserved	2		bslbf
section_length	12	2	uimsbf
zero	3	1	
protocol_version	5		see Sec. 4.4.1
ISO_639_language_code	24	3	per ISO 639 Part 2
transmission_medium	4		uimsbf see Table 5.17
table_type	4		uimsbf see Table 5.18
if (table_type==TNT) {			
satellite_ID	8	(1)	uimsbf range 0-255
first_index	8	(1)	uimsbf range 0-255
number_of_TNT_records	8	(1)	uimsbf
for (i=0; i<number_of_TNT_records; i++) {			
TNT_record()	*	((*))	
TNT_descriptors_count	8	((1))	range 0-255
for (i=0; i<TNT_descriptors_count; i++) {			
descriptor()	*	((*))	
}			
}			
}			
if (table_type==STT) {			
first_index	8	(1)	uimsbf range 0-255
number_of_STT_records	8	(1)	uimsbf
for (i=0; i<number_of_STT_records; i++) {			
STT_record()	*	((*))	
STT_descriptors_count	8	((1))	uimsbf range 0-255
for (i=0; i<STT_descriptors_count; i++) {			
descriptor()	*	(((*)))	
}			
}			
}			
if (table_type==RTT) {			
rating_region	8	(1)	uimsbf
rating_text_table()	*	(*)	
} else if (table_type==RST) {			
rating_system_table()	*	(*)	
} else if (table_type==CTT) {			
currency_system_table()	*	(*)	
} else if (table_type==SNT) {			
source_name_table()	*	(*)	
} else if (table_type==MNT) {			
map_name_table()	*	(*)	
}			
for (i=0; i<N; i++) {			
descriptor()	*	(*)	optional
}			
CRC_32	32	4	rpchof
}			

Figure 5.7. Network text message format.

Table 5.17 Transmission Medium

transmission_medium	meaning
0	**cable**
1	**satellite**
2	**MMDS**
3	**SMATV**
4	**over_the_air**
5-14	reserved
15	**all** — Indicates the table is applicable to all transmission media

A NETWORK TEXT message received with transmission_medium indicating an unknown or unsupported medium shall be discarded.

Table 5.18 Table Type

table_type	meaning
0	invalid
1	**TNT** — Transponder Name Table
2	**STT** — Satellite Text Table
3	**RTT** — Ratings Text Table
4	**RST** — Rating System Table
5	**SNT** — Source Name Table
6	**MNT** — Map Name Table
7-15	reserved

A NETWORK TEXT message received with table_type indicating an unknown or unsupported table_type shall be discarded.

5.2.5 ISO 639 Language Code

ISO_639_language_code — A 3-byte language code per ISO 639 Part 2 defining the language associated with the text carried in this NETWORK TEXT message. The ISO_639_language_code field contains a three character code as specified by ISO 639 Part 2. Each character is coded into 8 bits according to ISO 8859-1 (ISO Latin-1) and inserted in order into the 24 bit field. The value 0xFFFFFF shall be used in case the text is available in one language only. In the Decoder, value 0xFFFFFF shall represent a “wild card” match when filtering by language.

5.2.6 Transponder Name Table (TNT)

The Transponder Name Table provides textual names associated with satellite transponders, represented as multilingual character strings.

satellite_ID — An 8-bit unsigned integer number in the range zero to 255 that identifies the satellite associated with the data defined in this message

first_index — An unsigned 8-bit integer number in the range zero to 255 that indicates the index of the first TNT record to be defined in this message. If more than one record is provided, the additional records define successive table entries following first_index.

number_of_TNT_records — An unsigned 8-bit integer number that defines the number of records being defined in this message. The minimum allowed value is one. The maximum is limited by the maximum allowed length of the message.

TNT_record() — A data structure defined as shown in Figure 5.8.

	Bits	Bytes	Bit Number / Description
TNT_record(){			
reserved	2	1	bslbf reserved
transponder_number	6		uimsbf range 0-63
reserved	3	1	bslbf reserved
transponder_name_length	5		uimsbf range 0-31 (N)
if (transponder_name_length>0) {			
transponder_name()	8*N	N	
}			
}			

Figure 5.8. TNT record format.

transponder_number — A 6-bit unsigned integer value in the range zero to 63 that identifies the transponder number whose name is being defined by the transponder_name field to follow.

transponder_name_length — A 5-bit unsigned integer value in the range zero to 31 that defines the length, in bytes, of the transponder_name() field. If the value is zero, the transponder_name() field is omitted. The length represents the sum total of all <mode><length><segment> blocks comprising the multilingual text string to follow. Note that the field length does not necessarily correspond with the number of *characters* comprising the name, given that some encoding modes are double-byte.

transponder_name() — The textual name of the transponder, represented in the multilingual format described in Section 6. The name field consists of one or more <mode><length><segment> blocks.

5.2.7 Satellite Text Table (STT)

The Satellite Text Table provides textual names associated with satellites, in both abbreviated and full-length versions, represented as multilingual character strings. The first_index and number_of_records fields have the same definitions for the STT as for the TNT.

Figure 5.9 defines the structure of the STT_record().

satellite_ID — An 8-bit unsigned integer number in the range zero to 255 that identifies the satellite associated with the satellite name data defined in this record.

sat_reference_name_length — A 4-bit unsigned integer value in the range zero or four to 15 that defines the length, in bytes, of the sat_reference_name() field. If the length is zero, the sat_reference_name() field is omitted from the message. The length represents the sum total of all <mode><length><segment> blocks comprising the multilingual text string to follow.

Note that the field length does not necessarily correspond with the number of *characters* comprising the name, given that some encoding modes are double-byte.

	Bits	Bytes	Bit Number / Description
STT_record(){			
satellite_ID	8	1	uimsbf
reserved	4	1	bslbf reserved
sat_reference_name_length	4		uimsbf range 0,4-15 (R)
if (sat_reference_name_length>0) {			
sat_reference_name()	R*8	R	
}			
reserved	3	1	bslbf reserved
full_satellite_name_length	5		uimsbf range 0, 4-31 (L)
if (full_satellite_name_length>0) {			
full_satellite_name()	L*8	L	
}			
}			

Figure 5.9. STT record format.

sat_reference_name() — The abbreviated (reference) name of the satellite, represented in the multilingual format described in Section 6. The field consists of one or more <mode><length><segment> blocks.

full_satellite_name_length — A 5-bit unsigned integer value in the range zero or four to 31 that defines the length, in bytes, of the full_satellite_name() field. If the length is zero, the full_satellite_name() field is omitted from the message. Note that the field length does not necessarily correspond with the number of *characters* comprising the name, given that some encoding modes are double-byte.

full_satellite_name() — The full name of the satellite, represented in the multilingual format identified in Section 6. The field consists of one or more <mode><length><segment> blocks.

5.2.8 Rating Text Table (RTT)

The Rating Text Table provides textual representations for all rating dimensions for one particular rating region. Decoders shall disregard RTT definitions for regions other than the one they are currently assigned to.

The Rating Text Table defines parental rating text associated with particular rating regions. Downloadable rating text allows definitions of ratings to change dynamically. Note that the RTT provides for dynamic redefinition of all aspects of parental ratings control, including:

- The number of defined dimensions (up to six)
- The name of a particular dimension
- The number of defined levels within one dimension
- The textual string representation of any level

For each rating region, the name of each of the dimensions may be defined. If a particular dimension is undefined, the number of levels defined for that dimension shall be specified as zero, and no text data shall follow in the message for that dimension.

For each defined dimension, the textual representation for each of as many as 16 levels is defined. Figure 5.10 describes the format of the Rating Text Table for one ratings region.

For parental rating limits (or ceilings) set by the user, level zero is typically not an option, since it would mean "disallow any program that has a rating defined for this dimension." For these reasons, the textual representation for level zero is typically a null string (zero length).

	Bits	Bytes	Bit Number / Description
rating_text_table(){			
for (i=0; i<6; i++) {			
levels_defined	8	(1)	uimsbf range 0-16
if (levels_defined>0) {			
dimension_name_length	8	((1))	uimsbf (D) range 0-255
dimension_name()	8*D	((D))	
for (i=0; i<levels_defined; i++) {			
rating_name_length	8	(((1)))	uimsbf (R) range 0-255
rating_name_text()	R*8	(((R)))	
}			
}			
}			
}			

Figure 5.10. Rating text table format.

levels_defined — An unsigned 8-bit integer in the range zero to 16 that represents the number of levels defined for this particular dimension. If the number of defined levels is zero, the dimension_name_length and dimension_name() fields shall be omitted from the message.

dimension_name_length — An unsigned 8-bit integer in the range zero to 255 that represents the length, in bytes, of the dimension_name field to follow. The length represents the sum total of all <mode><length><segment> blocks comprising the multilingual text string to follow.

dimension_name() — A text string, in standard multilingual format, describing one particular rating dimension for this rating region. One dimension in the U.S. rating region, for example, is used to describe a program's MPAA rating. The dimension name for that dimension may be defined as "MPAA Rating."

rating_name_length — An unsigned 8-bit integer in the range zero to 255 that represents the length, in bytes, of the rating_name_text() field to follow. The length represents the sum total of all <mode><length><segment> blocks comprising the multilingual text string to follow.

rating_name_text() — A text string, in standard multilingual format, describing one particular rating level within the dimension being described in this loop in the message structure. One of the currently defined rating levels in the U.S. rating region in the MPAA Rating dimension, for example, is “PG-13.”

5.2.9 Rating System Table (RST)

The Rating System Table provides textual representations for each rating region in use in the network. A Decoder may use text from the RST to present the user with a menu of choices of rating system. Figure 5.11 defines the format of the rating_system_table().

	Bits	Bytes	Bit Number / Description
rating_system_table(){			
regions_defined	8	1	uimsbf range 1-63
for (i=0; i<regions_defined; i++) {			
data_length	8	(1)	uimsbf range 4-255
rating_region	8	(1)	uimsbf
string_length	8	(1)	uimsbf (L)
rating_system_text()	8*L	(L)	uimsbf
for (i=0; i<N; i++) {			
descriptor()	*	((*))	as ind. by data_length
}			
}			
}			

Figure 5.11. Rating system table format.

regions_defined — An 8-bit unsigned integer number in the range one to 63 that represents the number of rating regions being textually defined in the NETWORK TEXT message.

data_length — An 8-bit unsigned integer number in the range four to 255 that defines the number of bytes of data to follow in this iteration of the for loop. The data_length parameter is provided to allow an optional variable number of descriptor blocks to be included with each region definition.

rating_region — An 8-bit unsigned integer number that defines the rating region to be associated with the text in rating_system_text().

string_length — An 8-bit unsigned integer number that defines the total length in bytes of the rating_system_text() field to follow. The length represents the sum total of all <mode><length><segment> blocks comprising the multilingual text string to follow.

rating_system_text() — A data structure containing a multilingual text string defining the rating system given by rating_region. Text strings are formatted according to the rules outlined in Section 6.

5.2.10 Currency System Table (CST)

The Currency System Table provides textual representations for each currency region in use in the network. A Decoder may use text from the CST to present the user with a menu of choices of currency system. Figure 5.12 defines the format of the currency_system_table().

regions_defined — An 8-bit unsigned integer number in the range one to 63 that represents the number of currency regions being textually defined in the NETWORK TEXT message.

	Bits	Bytes	Bit Number / Description
currency_system_table(){			
regions_defined	8	1	uimsbf range 1-63
for (i=0; i<regions_defined; i++) {			
data_length	8	(1)	uimsbf range 4-255
currency_region	8	(1)	uimsbf
string_length	8	(1)	uimsbf (L)
currency_system_text()	8*L	(L)	uimsbf
for (i=0; i<N; i++) {			
descriptor()	*	((*))	as ind. by data_length
}			
}			
}			

Figure 5.12. Currency system table format.

data_length — An 8-bit unsigned integer number in the range four to 255 that defines the number of bytes of data to follow in this iteration of the for loop. The data_length parameter is provided to allow an optional variable number of descriptor blocks to be included with each region definition.

currency_region — An 8-bit unsigned integer number that defines the currency region to be associated with the text in currency_system_text().

string_length — An 8-bit unsigned integer number that defines the total length in bytes of the currency_system_text() field to follow. The length represents the sum total of all <mode><length><segment> blocks comprising the multilingual text string to follow.

currency_system_text() — A data structure containing a multilingual text string defining the currency system given by currency_region. Text strings are formatted according to the rules outlined in Section 6.

5.2.11 Source Name Table (SNT)

The format of the source_name_table() is given in Figure 5.13.

number_of_SNT_records — An unsigned 8-bit integer number in the range one to 255 that defines the number of records being defined in this message.

application_type — A Boolean flag that indicates, when set, that the name string being defined is for an application whose ID is given in application_ID. When the flag is clear, the name string being defined is for a source whose ID is given in source_ID.

application_ID — A 16-bit unsigned integer value identifying the application associated with the name string to follow.

source_ID — A 16-bit unsigned integer value identifying the programming source associated with the source name to follow.

name_length — An unsigned 8-bit integer number in the range one to 255 that defines the number of bytes in the name() structure to follow.

source_name() — A multilingual text string defining the name of the source or application, formatted according to the rules defined in Section 6.

	Bits	Bytes	Bit Number / Description
source_name_table(){			
number_of_SNT_records	8	1	uimsbf range 1-255
for (i=0; i<number_of_SNT_records; i++) {			
application_type	1	(1)	bslbf {false, true}
reserved	7		bslbf
if (application_type) {			
application_ID	16	((2))	uimsbf
} else {			
source_ID	16	((2))	uimsbf
}			
name_length	8	(1)	size of source_name() (L)
source_name()	L*8	(L)	multilingual text
SNT_descriptors_count	8	(1)	uimsbf range 0-255
for (i=0; i<SNT_descriptors_count; i++) {			
descriptor()	*	((*))	
}			
}			
}			

Figure 5.13. Source name table format.

SNT_descriptors_count — An unsigned 8-bit integer number in the range zero to 255 that defines the number of descriptors to follow.

5.2.12 Map Name Table (MNT)

The Map Name Table (MNT) provides a textual name for each Virtual Channel Table in the network. The names may be provided multilingually. Figure 5.14 describes the format.

	Bits	Bytes	Bit Number / Description
map_name_table(){			
number_of_MNT_records	8	1	uimsbf
for (i=0; i<number_of_MNT_records; i++) {			
VCT_ID	16	(2)	uimsbf
map_name_length	8	(1)	uimsbf (N)
map_name()	N*8	(N)	multilingual text
MNT_descriptors_count	8	(1)	uimsbf range 0-255
for (i=0; i<MNT_descriptors_count; i++) {			
descriptor()	*	((*))	
}			
}			
}			

Figure 5.14. Map name table format.

number_of_MNT_records — An 8-bit unsigned integer number in the range one to 255 that identifies the number of VCT_ID/MNT data records to follow in the message. The number of records included is further limited by the allowed maximum message length.

VCT_ID — A 16-bit unsigned integer value which defines the ID of the Virtual Channel Table to be associated with the name string given in the map_name() to follow.

map_name_length — An 8-bit unsigned integer value which defines the length, in bytes, of the map_name() structure to follow.

map_name() — A multilingual text string defining the textual name of the Virtual Channel Table whose ID is given in VCT_ID, given in the language defined in ISO_639_language_code.

MNT_descriptors_count — An 8-bit unsigned integer value in the range zero to 255 that defines the number of descriptors to follow.

5.2.13 Message-End Descriptors

The message may include at its end one or more structures of the form tag, length, data. The number of descriptors present is determined indirectly by processing the section_length field.

5.3 Virtual Channel Message

The VIRTUAL CHANNEL message delivers portions of VCTs, and also delivers the Defined Channel Map (DCM) and Inverse Channel Table (ICT). The VIRTUAL CHANNEL message accommodates the needs of all transmission media.

Figure 5.15 shows the format of the VIRTUAL CHANNEL message body.

	Bits	Bytes	Bit Number / Description
virtual_channel_message(){			
table_ID	8	1	uimsbf value 0xC4
zero	2		bslbf
ISO_reserved	2		bslbf
section_length	12	2	uimsbf
zero	3	1	
protocol_version	5		see Sec. 4.4.1
transmission_medium	4		uimsbf see Table 5.1
table_subtype	4		uimsbf see Table 5.19
VCT_ID	16	2	uimsbf
if (table_subtype==DCM) {			
DCM_structure()	*	(*)	
}			
if (table_subtype==VCT) {			
VCT_structure()	*	(*)	
}			
if (table_subtype== ICT) {			
ICT_structure()	*	(*)	
}			
for (i=0; i<N; i++) {			
descriptor()	*	(*)	optional
}			
CRC_32	32	4	rpchof
}			

Figure 5.15. Virtual channel message format.

5.3.1 Table ID

The table_ID of the VIRTUAL CHANNEL message is 0xC4.

5.3.2 Transmission Medium

transmission_medium — A 4-bit field that defines the transmission medium for which the data in this VIRTUAL CHANNEL message applies. Table 5.1 defines the coding.

A VIRTUAL CHANNEL message received with transmission_medium indicating an unknown or unsupported medium shall be discarded.

5.3.3 Table Subtype

table_subtype — A 4-bit field that indicates the table subtype being delivered in this VIRTUAL CHANNEL message. Three subtypes are currently defined, one delivering the Defined Channels Map (DCM), one delivering the Virtual Channel Table (VCT) itself, and one defining the Inverse Channel Table (ICT). Table 5.19 defines table_subtype.

Table 5.19 Table Subtype

table_subtype	meaning
0	**VCT** — Virtual Channel Table
1	**DCM** — Defined Channels Map
2	**ICT** — Inverse Channel Table
3-15	reserved

A VIRTUAL CHANNEL message received with table_subtype indicating an unknown or unsupported table subtype shall be discarded.

5.3.4 Virtual Channel Table ID

VCT_ID — A 16-bit unsigned integer value in the range zero to 0xFFFF indicating the VCT to which the channel definitions in this message apply.

5.3.5 Defined Channels Map Structure

Figure 5.16 shows the format of the DCM_structure().

	Bits	Bytes	Bit Number / Description
DCM_structure(){			
reserved	4	2	bslbf reserved
first_virtual_channel	12		uimsbf range 0-4095
reserved	1	1	bslbf reserved
DCM_data_length	7		uimsbf range 1-127
for (i=0; i<DCM_data_length; i++) {			
range_defined	1	(1)	bslbf {no, yes}
channels_count	7		uimsbf range 1-127
}			
}			

Figure 5.16. DCM structure format.

5.3.5.1 First Virtual Channel

first_virtual_channel — An unsigned 12-bit integer reflecting the first virtual channel whose existence is being provided by this message, for the map identified by the VCT_ID field. The range is zero to 4095.

5.3.5.2 DCM Data Length

DCM_data_length — A 7-bit unsigned integer number in the range one to 127 that defines the number of DCM_data_fields to follow in the message.

5.3.5.3 DCM Data Bytes

The DCM data bytes taken as a whole define which virtual channels, starting at the channel number defined by first_virtual_channel, are defined and which are not. Each DCM_data_field defines two pieces of data: a flag indicating whether this block of channels is defined or not, and the number of channels in the block. The bytes are interpreted in an accumulative way, with a pointer into the Virtual Channel Table assumed initialized to first_virtual_channel. As each byte is processed, the pointer is incremented by the number of channels indicated by the channel count field.

For example, if channels 2-90, 200-210, 400-410, 600-610, 800-810, and 999 were defined, and first_virtual_channel was zero, the DCM data sequence (in decimal) would be the following, where underlined numbers have the range_defined bit set: 2, <u>89</u>, 109, <u>11</u>, 127, 62, <u>11</u>, 127, 62, <u>11</u>, 127, 62, <u>11</u>, 127, 61, <u>1</u>.

5.3.5.4 Range Defined

range_defined — A Boolean flag that indicates, when true, that the number of channels given by channel_count is defined in the VCT, starting at the current pointer value. When the flag is clear, a number of channels equal to channels_count is currently not defined starting at the current pointer value.

5.3.5.5 Channels Count

channels_count — An unsigned 7-bit integer number in the range one to 127 that indicates the number of defined (or undefined) channels in a group.

5.3.6 Virtual Channel Table

Figure 5.17 shows the format of the VCT_structure().

5.3.6.1 Overlays

freq_spec_included, symbol_rate_included, descriptors_included — Boolean flags that indicate, when set, that the associated overlay is present in the message. When the flag is clear, the associated overlay is absent. Note that in the broadcast_virtual_channel() record structure, the only optional overlay is the one identified by descriptors_included.

Therefore, when transmission_medium is equal to over_the_air, the freq_spec_included and symbol_rate_included flags are undefined (reserved).

	Bits	Bytes	Bit Number / Description
VCT_structure(){			
freq_spec_included	1	1	bslbf {no, yes}
symbol_rate_included	1		bslbf {no, yes}
descriptors_included	1		bslbf {no, yes}
reserved	5		bslbf reserved
splice	1	1	bslbf {no, yes}
reserved	7		bslbf reserved
activation_time	32	4	uimsbf units: GPS sec.
number_of_VC_records	8	1	
for (i=0; i<number_of_VC_records; i++) {			
if (transmission_medium==satellite {			
satellite_virtual_channel()	*	(*)	
} else if (transmission_medium==SMATV) {			
SMATV_virtual_channel()	*	(*)	
if (transmission_medium==over_the_air {			
broadcast_virtual_channel()	*	(*)	
} else { /* cable, MMDS*/			
virtual_channel()	*	(*)	
}			
}			

Figure 5.17. VCT structure format.

5.3.6.2 Processing of Local Access Virtual Channel Records

The VIRTUAL CHANNEL message defines two basic types of virtual channels, primary and “local access.” The primary virtual channel records are stored in the Decoder so that any channel can be acquired given its channel number. The idea of a local access point is that a program associated with a particular channel may be carried also within Transport Stream other than the one referenced by the primary virtual channel definition. The local access virtual channel gives the location on *this* TS of a program associated with the primary one.

An example is an IPG service, where one (hidden) virtual channel defines the point of access to a MPEG program carrying one or more streams of IPG data. If it is desired to carry IPG on other Transport Streams, programs on these other Transport Streams could be defined to carry lower-rate data streams carrying IPG data. A “local access” virtual channel definition may be defined and included in network data generated from the uplink control system. In order to allow the Decoder to associate these programs with the primary one, the local access virtual channel would quote the same virtual channel number as the primary entry point.

Local access records do not overwrite or replace records in the Virtual Channel Table. Processing of local access virtual channel records involves building a list upon entry to a transponder or carrier. If programs can be processed in the background, this list is consulted to allow the Decoder to select and decrypt a stream as appropriate.

5.3.6.3 Activation Control and Time

The activation time indicates the time at which the data delivered in the message will be valid.

splice — A Boolean flag that indicates, when set, that the Decoder shall arm video processing hardware to process a splice point if the virtual channel changes described in the message apply to a currently acquired channel, and the activation point is reached. If the activation is immediate or specified as a time that has since passed, splice processing is not used. When the splice flag is clear, the virtual channel change is made directly, without arming video hardware for a splice.

activation_time — A 32-bit unsigned integer field providing the absolute second the virtual channel data carried in the message will be valid, defined in terms of the number of GPS seconds since January 6th, 1980. Refer to Annex B for a discussion of time in this System Information Standard. If the activation_time is in the past, the data in the message shall be considered valid immediately. An activation_time value of zero shall be used to indicate immediate activation.

5.3.6.4 Number of Virtual Channel Records

number_of_VC_records — An 8-bit unsigned integer number in the range one to 255 that identifies the number of virtual_channel() records to follow in the message. The number of records included is further limited by the allowed maximum message length.

5.3.6.5 Virtual Channel Structure for Satellite Transmission Medium

Figure 5.18 shows the format of the satellite_virtual_channel() structure.

5.3.6.5.1 Virtual Channel Number

virtual_channel_number — An unsigned 12-bit integer in the range zero to 4095 reflecting the virtual channel whose definition is being provided by this virtual channel record, for the map identified by the VCT_ID field.

5.3.6.5.2 Application Virtual Channel

application_virtual_channel — A binary flag that indicates, when set, that this virtual channel defines an access point for an application whose ID is given in application_ID. When the flag is clear, the channel is not an application access point, and the 16-bit application_ID/source_ID field is the source_ID.

bitstream_select — A one-bit field that identifies the bit stream associated with programs referenced by this virtual channel number, when the Transport Stream carrying the virtual channel is MPEG-2 compatible, and the Transport Stream is carried on a split bit stream. When the TS is a combined TS, the bitstream_select bit shall indicate stream_1. Table 5.20 defines bitstream_select.

	Bits	Bytes	Bit Number / Description
satellite_virtual_channel(){			
reserved	4	2	bslbf reserved
virtual_channel_number	12		uimsbf range 0-4095
application_virtual_channel	1	1	bslbf {no, yes}
bitstream_select	1		bslbf see Table 5.20
reserved	1		bslbf reserved
transport_type	1		bslbf see Table 5.21
channel_type	4		uimsbf see Table 5.22
if (application_virtual_channel) {			
application_ID	16	(2)	uimsbf
} else {			
source_ID	16	(2)	uimsbf
}			
if (channel_type ==NVOD_access) {			
reserved	4	(2)	bslbf reserved
NVOD_channel_base	12		uimsbf range 1-4095
reserved	16	(2)	bslbf reserved
} else			
if (transport_type==MPEG_2) {			
satellite	8	((1))	uimsbf range 0-255
reserved	2	((1))	bslbf reserved
transponder	6		uimsbf range 0-63
program_number	16	((2))	uimsbf
} else { /* non-MPEG_2 */			
satellite	8	((1))	uimsbf range 0-255
reserved	2	((1))	bslbf reserved
transponder	6		uimsbf range 0-63
reserved	16	((2))	bslbf reserved
}			
if (freq_spec_included) {		(2)	
frequency_unit	1		bslbf see Table 5.23
carrier_frequency	15		uimsbf units: 10 or 125kHz
}			
if (symbol_rate_included) {			
reserved	4	(4)	bslbf
symbol_rate	28		uimsbf units: 1 sym per sec
}			
if (descriptors_included) {			
descriptors_count	8	(1)	uimsbf
for (i=0; i<descriptors_count; i++) {			
descriptor()	*	((*))	
}			
}			
}			

Figure 5.18. Virtual channel structure format for satellite transmission medium.

Table 5.20 Bitstream Select

bitstream_select	meaning
0	stream_0
1	stream_1

5.3.6.5.3 Transport Type

transport_type — A 1-bit field identifying the type of transport carried on this transponder as either being an MPEG-2 transport (value zero), or not (value one). Table 5.21 defines transport_type.

Table 5.21 Transport Type

transport_type	meaning
0	MPEG-2 transport
1	non-MPEG-2 transport

5.3.6.5.4 Channel Type

channel_type — A 4-bit field defining the channel type. Table 5.22 defines channel_type.

Control streams associated with background programs may exist on multiple Transport Streams. A Transport Stream other than the one identified in the virtual channel defining the program may also carry the same (or related) data. When that occurs, a virtual channel record tagged as local_access, associated with the same virtual channel number, will be present to define local access to the program. Decoders supporting background program processing shall store local_access virtual channel definitions separately from the working VCT, and use the local definition when not on the Transport Stream identified in the working map.

Table 5.22 Channel Type

channel_type	meaning
0	**normal** — Indicates that the record is a regular virtual channel record, and does not define a local access, hidden, or NVOD channel. For non-MPEG-2 channels, the waveform_type shall be defined as "normal."
1	**hidden** — Indicates that the record identifies a virtual channel that may not be accessed by the user by direct entry of the channel number (hidden). Hidden channels are skipped when the user is channel surfing, and appear as if undefined if accessed by direct channel entry. Programs constructed for use by specific applications (such as NVOD theaters) utilize hidden virtual channels.
2	**local access** — Indicates that the virtual channel record is not stored in the permanent (or working) map, but overrides the definition in the working map only on the Transport Stream upon which it was received.
3	**NVOD access** — Indicates that the virtual channel is an access point to a number of hidden virtual channels comprising a near video on demand (NVOD) group. Instead of containing a regular virtual channel definition indicating the location of a Transport Stream, the virtual channel record points to the base of a group of hidden virtual channels comprising the theater cine-plex.
4-15	**reserved** — Decoders shall treat virtual channel records of unknown channel_type the same as non-existent (undefined) channels.

5.3.6.5.5 Application ID

application_ID — A 16-bit unsigned integer number in the range 0x0001 to 0xFFFF that identifies the application associated with the virtual channel, on a system-wide basis. One particular program guide application, for example, may look for a program carrying data in its native transmission format by searching through the Virtual Channel Table for a match on its assigned application_ID. In some cases, one application may be able to process streams associated with more than one application ID. The application ID may be used to distinguish content as well as format, for the benefit of processing within the application. The value zero for application_ID shall not be assigned; if specified in a VIRTUAL CHANNEL message, the value zero indicates "unknown" or "inapplicable" for the application_ID/source_ID field.

Refer to Annex B for a discussion of the use of application_ID.

5.3.6.5.6 Source ID

source_ID — A 16-bit unsigned integer number in the range 0x0000 to 0xFFFF that identifies the programming source associated with the virtual channel, on a system-wide basis. In this context, a *source* is one specific source of video, text, data, or audio programming. For the purposes of referencing virtual channels to the program guide database, each such program source is associated with a unique value of source_ID. The source_ID itself may appear in an IPG database, where it tags entries to associate them with specific services. The value zero for source_ID, if used, shall indicate the channel is not associated with a source ID.

5.3.6.5.7 NVOD Channel Base

NVOD_channel_base — A 12-bit unsigned integer number in the range zero to 4095 that defines, for NVOD base channels, the index of the virtual channel record defining the access path for NVOD theater zero.

5.3.6.5.8 Satellite and Transponder

satellite — An 8-bit unsigned integer number in the range zero to 255 that identifies the satellite that carries the programs referenced by this virtual channel. The satellite combined with the transponder identify the point of access for programs accessed through this virtual channel number.

transponder — A 6-bit unsigned integer number in the range zero to 63 that identifies the transponder number on the satellite identified by the satellite field that carries the programs referenced by this virtual channel.

5.3.6.5.9 Program Number

program_number — A 16-bit unsigned integer number that associates the virtual channel number being defined with services defined in the PROGRAM ASSOCIATION and TS PROGRAM MAP messages. Access to elementary streams defined in each virtual channel record involves first acquiring the Transport Stream on the satellite and transponder associated with the virtual channel, then referencing the PROGRAM ASSOCIATION section in PID 0 to find the PID associated with the TS PROGRAM MAP message for this program_number. PIDs for each elementary stream are then found by acquisition of the TS PROGRAM MAP message.

A program_number with value 0x0000 (invalid as a regular program number) is reserved to indicate that the Decoder shall discard the corresponding virtual channel record from the queue of pending virtual channel changes. Records are identified in the pending queue by their activation_time, VCT_ID, and virtual_channel_number. If no pending virtual channel change is found in the Decoder's queue, no action shall be taken for this virtual channel (i.e. the record shall be discarded).

5.3.6.5.10 Frequency Specification Option

The VIRTUAL CHANNEL message may optionally specify the carrier frequency associated with the Transport Stream carrying the service referenced by the virtual channel record. Such specification represents the exceptional case, since the CDT will list carrier frequencies in common use.

A commercial or private SCPC application on satellite may require such specification, for example, because the number of potential frequencies for SCPC carriers may exceed the capacity of the network tables, or because the network operator decides not to include SCPC frequencies in the tables. By use of the frequency specification option the physical transponder can be identified, but instead of the center frequency, the frequency specified in the virtual channel record would be tuned on acquisition.

The frequency specified shall be considered to be an *override* for (not a replacement of) frequency data carried in the tables carried in the NETWORK INFORMATION message. The frequency is applicable only for acquisition of the transport stream carrying the service referenced in the virtual channel record specifying it.

frequency_unit — A one-bit field identifying the units for the carrier_frequency field. Table 5.23 defines the coding.

Table 5.23 Frequency Unit

frequency_unit	meaning
0	10 kHz units
1	125 kHz units

carrier_frequency — An unsigned 15-bit integer number in the range zero to 32,767 that defines the carrier frequency for the carrier, in units of either 10 kHz or 125 kHz, depending on the value of frequency_unit. When the frequency_unit indicates 125 kHz, the carrier_frequency can be interpreted as a fractional frequency (1/8 MHz) in the least-significant 3 bits, and an integer number of megahertz in the upper 12 bits. The range of frequencies that can be represented is zero to ((2^15) -1) * 125 kHz = 4095.875 MHz.

5.3.6.5.11 Symbol Rate Specification Option

The VIRTUAL CHANNEL message may optionally specify symbol rate associated with the Transport Stream carrying the service referenced by the virtual channel record. Such specification represents the exceptional case, since the MMT will list modulation configurations in common use.

symbol_rate — A 28-bit unsigned integer field that indicates the symbol rate in units of one symbol per second associated with acquisition of the TS carrying the service associated with this virtual channel.

5.3.6.5.12 Optional Descriptors

descriptors_count — An 8-bit unsigned integer value in the range zero to 255 that defines the number of descriptors to follow.

5.3.6.6 Virtual Channel Record for SMATV

Figure 5.19 defines the format of the SMATV_virtual_channel() record, used to define VCTs for SMATV transmission medium. The definitions of most parameters are the same for this data structure as for the satellite_virtual_channel() record. Exceptions and additions are listed below.

5.3.6.6.1 Audio Mode

audio_mode() — A data structure defining the location and format of analog audio subcarriers associated with the virtual channel. Refer to Section 5.1.6.3 for the format of the audio_mode() structure.

5.3.6.6.2 CDT Reference

CDT_reference — An unsigned 8-bit integer number in the range one to 255 that identifies the frequency associated with this virtual channel. The value zero is illegal and shall not be specified. The CDT_reference is used as an index into the Carrier Definition Table appropriate to the transmission medium to find a frequency to tune to acquire the virtual channel.

5.3.6.6.3 MMT Reference

MMT_reference — An 8-bit unsigned integer value in the range one to 255 that references an entry in the Modulation Mode Table (MMT) appropriate to the transmission medium. The value zero is illegal and shall not be specified. For digital waveforms, the MMT_reference associates the carrier with a digital modulation mode. For Decoder implementations that support only one set of modulation parameters, in systems in which one modulation method is used for all carriers, storage and processing of the MMT_reference is unnecessary.

	Bits	Bytes	Bit Number / Description
SMATV_virtual_channel(){			
reserved	4	2	bslbf reserved
virtual_channel_number	12		uimsbf range 0-4095
application_virtual_channel	1	1	bslbf {no, yes}
bitstream_select	1		bslbf see Table 5.20
reserved	1		bslbf reserved
transport_type	1		bslbf see Table 5.21
channel_type	4		uimsbf see Table 5.22
if (application_virtual_channel) {			
application_ID	16	(2)	uimsbf
} else {			
source_ID	16	(2)	uimsbf
}			
if (channel_type ==NVOD_access) {			
reserved	4	(2)	bslbf reserved
NVOD_channel_base	12		uimsbf range 1-4095
reserved	24	(3)	bslbf reserved
} else			
if (transport_type==MPEG_2) {			
CDT_reference	8	((1))	uimsbf range 1-255
program_number	16	((2))	
MMT_reference	8	((1))	uimsbf range 1-255
reserved	8	((1))	bslbf reserved
} else { /* non-MPEG-2 */			
CDT_reference	8	((1))	uimsbf range 0-255
scrambled	1	((1))	bslbf {false, true}
reserved	3		bslbf
video_standard	4		uimsbf see Table 5.24
audio_mode()	24	((3))	
}			
if (freq_spec_included) {			
frequency_unit	1	(2)	bslbf see Table 5.23
carrier_frequency	15		uimsbf units: 10 or 125kHz
}			
if (symbol_rate_included) {			
reserved	4	(4)	bslbf
symbol_rate	28		uimsbf units: 1 sym/sec
}			
if (descriptors_included) {			
descriptors_count	8	(1)	
for (i=0; i<descriptors_count; i++) {			
descriptor()	*	((*))	
}			
}			
}			

Figure 5.19. Virtual channel structure format for SMATV transmission medium.

5.3.6.6.4 Non-Standard Channels

scrambled — A 1-bit Boolean flag that indicates, when set, that the associated non-Standard waveform is scrambled. When the flag is clear, the non-Standard waveform is in the clear.

video_standard — A 4-bit field that indicates the video standard associated with this non-Standard virtual channel. Table 5.24 defines video_standard.

Table 5.24 Video Standard

video_standard	meaning
0	**NTSC** — The video standard is NTSC
1	**PAL 625** — The video standard is 625-line PAL
2	**PAL 525** — The video standard is 525-line PAL
3	**SECAM** — The video standard is SECAM
4	**MAC** — The video standard is MAC
5-15	reserved

5.3.6.7 Broadcast Virtual Channel Structure

Figure 5.20 defines the format of the broadcast_virtual_channel() record, used to define VCTs for over-the-air transmission media. The definitions of fields in the broadcast_virtual_channel() record are the same as those previously defined for other media.

5.3.6.8 Virtual Channel Record for Cable and MMDS

Figure 5.21 defines the format of the virtual_channel() record, used to define VCTs for cable and MMDS transmission media. The definitions of all parameters except for path_select are as defined for the satellite_virtual_channel() or SMATV_virtual_channel() structures.

5.3.6.8.1 Path Select

path_select — A 1-bit field that associates the virtual channel with a transmission path. For cable transmission medium, path_select identifies the physical cable that carries the Transport Stream associated with this virtual channel. Table 5.25 defines path_select.

Table 5.25 Path Select

path_select	meaning
0	path 1
1	path 2

5.3.7 Inverse Channel Map

The inverse channel map, once reconstructed in the Decoder from a sequence of VIRTUAL CHANNEL message (ICM subtype), consists of a list of source_ID/virtual_channel pairs, ordered by source_ID. The Decoder may use this table to quickly find the virtual channel carrying the program given by a particular value of source_ID (by binary search), if such a virtual channel exists. One Inverse Channel Map is defined per Virtual Channel Table. The ICM may be constructed from the VCT, or linear searches may be done to resolve source_ID references. Transmission of the ICM is therefore optional.

Virtual channels that provide access points for applications (i.e., ones tagged with the application_virtual_channel flag) are not included in the ICM.

	Bits	Bytes	Bit Number / Description
broadcast_virtual_channel(){			
reserved	4	2	bslbf reserved
virtual_channel_number	12		uimsbf range 0-4095
application_virtual_channel	1	1	bslbf {no, yes}
reserved	2		bslbf
transport_type	1		bslbf see Table 5.21
channel_type	4		uimsbf see Table 5.22
if (application_virtual_channel) {			
application_ID	16	(2)	
} else {			
source_ID	16	(2)	
}			
if (channel_type ==NVOD_access) {			
reserved	4	(2)	bslbf reserved
NVOD_channel_base	12		uimsbf range 1-4095
} else			
if (transport_type==MPEG_2) {			
program_number	16	((2))	uimsbf
} else { /* non-MPEG-2 */			
scrambled	1	((1))	bslbf {no, yes}
reserved	3		bslbf
video_standard	4		uimsbf see Table 5.24
reserved	8	((1))	bslbf
}			
}			
frequency_unit	1	2	bslbf see Table 5.23
carrier_frequency	15		uimsbf units: 10 or 125kHz
if (descriptors_included) {			
descriptors_count	8	(1)	uimsbf
for (i=0; i<descriptors_count; i++) {			
descriptor()	*	((*))	
}			
}			
}			

Figure 5.20. Broadcast virtual channel format.

Figure 5.22 describes the format of the ICM_structure().

5.3.7.1 Record Count

record_count — A 7-bit unsigned integer value in the range one to 127 that represents the total number of source_ID/ virtual_channel pairs defined in this message.

5.3.7.2 Source ID and Virtual Channel

source_ID — A 16-bit unsigned integer number in the range 0x0001 to 0xFFFF that identifies the source associated with the virtual channel, on a system-wide basis. In this context, a "source" is one specific source of video, text, data, or audio programming. For the purposes of referencing virtual channels to the program guide database, each such source is associated with a unique value of source_ID. Value zero for source_ID is illegal.

	Bits	Bytes	Bit Number / Description
virtual_channel(){			
reserved	4	2	bslbf reserved
virtual_channel_number	12		uimsbf range 0-4095
application_virtual_channel	1	1	bslbf {no, yes}
bitstream_select	1		bslbf see Table 5.20
path_select	1		bslbf see Table 5.25
transport_type	1		bslbf see Table 5.21
channel_type	4		uimsbf see Table 5.22
if (application_virtual_channel) {			
application_ID	16	(2)	
} else {			
source_ID	16	(2)	
}			
if (channel_type ==NVOD_access) {			
reserved	4	(2)	bslbf reserved
NVOD_channel_base	12		uimsbf range 1-4095
reserved	16	(2)	bslbf reserved
} else			
if (transport_type==MPEG_2) {			
CDT_reference	8	((1))	uimsbf range 1-255
program_number	16	((2))	
MMT_reference	8	((1))	uimsbf range 1-255
} else { /* non-MPEG-2 */			
CDT_reference	8	((1))	uimsbf range 0-255
scrambled	1	((1))	bslbf {no, yes}
reserved	3		bslbf reserved
video_standard	4		uimsbf see Table 5.24
reserved	16	((2))	uimsbf
}			
if (freq_spec_included) {		(2)	
frequency_unit	1		bslbf see Table 5.23
carrier_frequency	15		uimsbf units: 10 or 125kHz
}			
if (symbol_rate_included) {			
reserved	4	(4)	bslbf
symbol_rate	28		uimsbf units: 1 sym per sec
}			
if (descriptors_included) {			
descriptors_count	8	(1)	uimsbf
for (i=0; i<descriptors_count; i++) {			
descriptor()	*	((*))	
}			
}			
}			

Figure 5.21. Virtual channel format.

virtual_channel — A 12-bit unsigned integer value in the range zero to 4095 that represents the virtual channel, in the Virtual Channel Table given by VCT_ID, associated with the given source_ID. A virtual_channel value of zero indicates that the program given by source_ID is currently not carried in this Virtual Channel Table. Such placeholders are useful in the case where a certain program's existence within a virtual channel table may come and go.

	Bits	Bytes	Bit Number / Description
ICM_structure(){			
reserved	16	2	bslbf reserved
reserved	1	1	bslbf reserved
record_count	7		uimsbf range 1-127
for (i=0; i<record_count; i++) {			
source_ID	16	(2)	
reserved	4	(2)	bslbf reserved
virtual_channel	12		uimsbf range 0-4095
}			
}			

Figure 5.22. ICM structure format.

5.3.8 Message-End Descriptors

The message may include at its end one or more structures of the form tag, length, data. The number of descriptors present is determined indirectly by processing the section_length field.

5.4 System Time Message

The SYSTEM TIME message is used to synchronize Decoders with accurate calendar time, as well as to define IPPV reportback slot timing. Rate of transmission is typically once per minute, at second *00* of each minute.

The processing of the SYSTEM TIME message in the Decoder is time-critical. Delays between reception of the message and processing it increase the inaccuracy of events specified via system time (GPS) seconds. Processing delays should be kept below a maximum of approximately 200 milliseconds.

Figure 5.23 shows the format of the SYSTEM TIME message.

	Bits	Bytes	Bit Number / Description
system_time_message(){			
table_ID	8	1	uimsbf value 0xC6
zero	2		bslbf
ISO_reserved	2		bslbf
section_length	12	2	uimsbf
zero	3	1	
protocol_version	5		see Sec. 4.4.1
reserved	8	1	bslbf
system_time	32	4	uimsbf GPS seconds
GPS_UTC_offset	8	1	uimsbf seconds
for (i=0; i<N; i++) {			
descriptor()	*	(*)	optional
}			
CRC_32	32	4	rpchof
}			

Figure 5.23. System time message format.

5.4.1 Table ID

The table_ID of the SYSTEM TIME message is 0xC6.

5.4.2 System Time

system_time — A 32-bit unsigned integer quantity representing the current system time as the number of GPS seconds since 12 am, January 6th, 1980.

5.4.3 GPS to UTC Offset

GPS_UTC_offset — An 8-bit unsigned integer that defines the current offset in whole seconds between GPS and UTC time standards. To convert GPS time to UTC, the GPS_UTC_offset is subtracted from GPS time. Whenever the International Bureau of Weights and Measures decides that the current offset is too far in error, an additional leap second may be added (or subtracted), and the GPS_UTC_offset will reflect the change.

5.4.4 Message-End Descriptors

The message may include at its end one or more structures of the form tag, length, data. The number of descriptors present is determined indirectly by processing the section_length field.

6. MULTILINGUAL CHARACTER STRINGS

This section describes the format of multilingual character strings in this System Information Standard.

6.1 General Format

The format of multilingual text strings adheres to the following structure. Items in square brackets may be repeated one or more times:

<mode><length><segment> [<mode><length><segment>]

Preceding the one or more instances of mode, length, segment is always a string_length field. This field is described in each instance where multilingual text is used, and may be either 8- or 16-bits in length, as appropriate. The value of string_length represents the sum total of all mode, length, segment blocks comprising the multilingual text string to follow, and serves to indicate the end of the text string structure.

The multilingual text data structure is designed to accommodate the need to represent a text string composed of characters from a variety of alphabets, as well as ideographic characters. Whereas characters could be represented using 16- or 32-bit character codes (as does Unicode [*ISO/IEC 10646-1]*), that form is inefficient and wasteful of transmission bandwidth for strings composed primarily of alphabetic rather than ideographic characters. To accommodate the need to handle Chinese, Japanese, and Korean, modes are defined that allow 16-bit (double byte) character representations in standard formats.

References below to *ISO/IEC 10646-1* (Unicode) shall be to the Basic Multilingual Plane (BMP) within that standard.

mode — An 8-bit value representing the text mode to be used to interpret characters in the segment to follow. See Table 6.1 for definition. Mode bytes in the range zero through 0x3E select Unicode character code pages. Mode byte value 0x3F selects 16-bit Unicode character coding. Mode bytes in the rage 0x40 through 0xFF represent selection of a format effector function such as *underline ON* or *new line*. If mode is in the range 0x40 to 0x9F, then the length/segment portion is omitted. Format effector codes in the range 0x40 through 0x9F involve no associated parametric data, hence the omission of the length/segment portion. Format effector codes in the range 0xA0 through 0xFF include one or more parameters specific to the particular format effector function.

length — An 8-bit unsigned integer number representing the number of bytes in the segment to follow in this block.

segment — An array of bytes representing a character string formatted according to the mode byte.

Figure 6.1 describes the format of the multilingual_text_string().

Table 6.1 Mode Byte Encoding

Mode Byte	Meaning	Language(s) or Script
0x00	Select ISO/IEC 10646-1 Page 0x00	ASCII, ISO Latin-1 (Roman)
0x01	Select ISO/IEC 10646-1 Page 0x01	European Latin (many)[4]
0x02	Select ISO/IEC 10646-1 Page 0x02	Standard Phonetic
0x03	Select ISO/IEC 10646-1 Page 0x03	Greek
0x04	Select ISO/IEC 10646-1 Page 0x04	Russian, Slavic
0x05	Select ISO/IEC 10646-1 Page 0x05	Armenian, Hebrew
0x06	Select ISO/IEC 10646-1 Page 0x06	Arabic[5]
0x07-0x08	Reserved	-
0x09	Select ISO/IEC 10646-1 Page 0x09	Devanagari[6], Bengali
0x0A	Select ISO/IEC 10646-1 Page 0x0A	Punjabi, Gujarti
0x0B	Select ISO/IEC 10646-1 Page 0x0B	Oriya, Tamil
0x0C	Select ISO/IEC 10646-1 Page 0x0C	Telugu, Kannada
0x0D	Select ISO/IEC 10646-1 Page 0x0D	Malayalam
0x0E	Select ISO/IEC 10646-1 Page 0x0E	Thai, Lao
0x0F	Reserved	-
0x10	Select ISO/IEC 10646-1 Page 0x10	Tibetan, Georgian
0x11-0x1F	Reserved	-
0x20	Select ISO/IEC 10646-1 Page 0x20	Miscellaneous[7]
0x21	Select ISO/IEC 10646-1 Page 0x21	Misc. symbols, arrows
0x22	Select ISO/IEC 10646-1 Page 0x22	Mathematical operators
0x23	Select ISO/IEC 10646-1 Page 0x23	Misc. technical
0x24	Select ISO/IEC 10646-1 Page 0x24	OCR, enclosed alpha-num.
0x25	Select ISO/IEC 10646-1 Page 0x25	Form and chart components
0x26	Select ISO/IEC 10646-1 Page 0x26	Miscellaneous dingbats
0x27	Select ISO/IEC 10646-1 Page 0x27	Zapf dingbats
0x28-0x2F	Reserved	-
0x30	Select ISO/IEC 10646-1 Page 0x30	Hiragana, Katakana
0x31	Select ISO/IEC 10646-1 Page 0x31	Bopomopho, Hangul elem.
0x32	Select ISO/IEC 10646-1 Page 0x32	Enclosed CJK Letters, ideo.
0x33	Select ISO/IEC 10646-1 Page 0x33	Enclosed CJK Letters, ideo.
0x34-0x3E	Reserved	-
0x3F	Select 16-bit ISO/IEC 10646-1 mode	all
0x40-0x9F	Format effector (single byte)	see Table 6.2
0xA0-0xFF	Format effector (with parameter[s])	-

[4] When combined with page zero (ASCII and ISO Latin-1), covers Afrikaans, Breton, Basque, Catalan, Croatian, Czech, Danish, Dutch, Esperanto, Estonian, Faroese, Finnish, Flemish, Firsian, Greenlandic, Hungarian, Icelandic, Italian, Latin, Latvian, Lithuanian, Malay, Maltese, Norwegian, Polish, Portugese, Provencal, Ghaeto-Romanic, Romanian, Romany, Slovak, Slovenian, Serbian, Spanish, Swedish, Turkish, and Welsh.

[5] Also Persian, Urdu, Pashto, Sindhi, and Kurdish.

[6] Devanagari script is used for writing Sanscrit and Hindi, as well as other languages of northern India (such as Marathi) and of Nepal (Nepali). In addition, at least two dozen other Indian languages use Devanagari script.

[7] General punctuation, superscripts and subscripts, currency symbols, and other diacritics.

	Bits	Bytes	Bit Number/Description
multilingual_text_string(){			
for (i=0; i<N; i++) {			
mode	8	(1)	uimsbf
if (mode < 0x3F) {			
eightbit_string_length	8	((1))	uimsbf
for (i=0; i<eightbit_string_length; I++) {			
eightbit_char	8	(((1)))	uimsbf
}			
} else if (mode==0x3F) {			
sixteenbit_string_length	8	((1))	uimsbf (even)
for (i=0; i<(sixteenbit_string_length); i+=2) {			
sixteenbit_char	16	(((2)))	uimsbf
}			
} else if (mode >= 0xA0) {			
format_effector_param_length	8	((1))	uimsbf
for (i=0; i<(format_effector_param_length); i++) {			
format_effector_data	8	(((1)))	
}			
}			
}			
}			

Figure 6.1. Multilingual text string format.

6.2 Mode Byte Definition

The mode byte is used to either select an *ISO/IEC 10646-1* code page from the BMP (exact mapping, or in the case of page zero, an extended mapping as defined herein), or to indicate that the text segment is coded in one of a number of standard double-byte formats. Table 6.1 shows the encoding of the mode byte. Values in the zero to 0x33 range select ISO/IEC 10646-1 code pages.

Value 0x3F selects double-byte forms used with non-alphabetic script systems, where the segment consists of a sequence of 16-bit character codes according to the *ISO/IEC 10646-1* standard. Byte ordering is high-order byte first (Motorola 680xx style), a.k.a. *big-endian*.

6.3 Format Effectors

Mode bytes in the 0x40 to 0xFF range are defined as format effectors. Table 6.2 defines the encoding for currently defined single-byte values. Format effectors in the range 0x40 through 0x9F are self-contained, and do not have a length or data field following them. Format effectors in the range 0xA0 through 0xFF include a multi-byte parameter field. No multi-byte format effectors are currently defined.

6.3.1 Line Justification

Values 0x80, 0x81, and 0x82 signify the end of a line of displayed text. Value 0x80 indicates that the text is displayed left justified within an enclosing rectangular region (defined outside the scope of the text string). Value 0x81 indicates that the text is displayed right justified. Value 0x82 indicates that the text is centered on the line. The dimensions and location on the screen of the box into which text is placed is defined outside the scope of the text string itself.

Table 6.2 Format Effector Function Codes

Mode Byte	Meaning
0x40-0x7F	reserved
0x80	new line, left justify
0x81	new line, right justify
0x82	new line, center
0x83	italics ON
0x84	italics OFF
0x85	underline ON
0x86	underline OFF
0x87	bold ON
0x88	bold OFF
0x89-0x9F	reserved

6.3.2 Italics, Underline, Bold Attributes

These format effectors toggle *italics*, underline, and **bold** display attributes. The italics, underline, and bold format effectors indicate the start or end of the associated formatting within a text string. Formatting extends through new lines. For example, to display three lines of bold text, only one instance of the *bold ON* format effector is required.

6.3.3 Processing of Unknown or Unsupported Format Effectors

Receivers shall discard format effectors that are unknown, or known not to be supported within a specific receiver model. If a parameter value carries an undefined value, that format effector shall be discarded.

6.4 Default Attributes

Upon entry to a multilingual text string, all mode toggles (bold, underline, italics) shall be assumed “OFF”.

6.5 Mode Zero

ISO/IEC 10646-1 page zero (U+0000 through U+00FF) includes ASCII in the lower half (U+0000 through U+007F), and Latin characters from ISO 8859-1, *Latin-1*, in U+0090 through U+00FF. This set of characters covers Danish, Dutch, Faroese, Finnish, French, German, Icelandic, Irish, Italian, Norwegian, Portuguese, Spanish and Swedish. Many other languages can be written with this set of letters, including Hawaiian, Indonesian/Malay, and Swahili.

Table 6.3 shows encodings of page zero characters in the range 0x80 through 0x9F (these are undefined within *ISO/IEC 10646-1*).

Table 6.3 Encodings of Columns 8 and 9 of Mode Zero Latin Character Set

	8	9
0	<RESERVED>	<RESERVED>
1	<RESERVED>	<RESERVED>
2	<RESERVED>	<RESERVED>
3	<RESERVED>	<RESERVED>
4	<RESERVED>	<RESERVED>
5	<RESERVED>	<RESERVED>
6	<RESERVED>	<RESERVED>
7	<RESERVED>	<RESERVED>
8	<RESERVED>	U+2030 — <PER MILLE>
9	<RESERVED>	<RESERVED>
A	<RESERVED>	U+266A — <MUSICAL NOTE>
B	<RESERVED>	<RESERVED>
C	<RESERVED>	U+2190 — <LEFT ARROW>
D	<RESERVED>	U+2191 — <UP ARROW>
E	<RESERVED>	U+2192 — <RIGHT ARROW>
F	<RESERVED>	U+2193 — <DOWN ARROW>

6.6 Supported Characters

Support for specific characters and languages depends upon the specific model of Standard-compatible Decoder. Not all Decoders support all defined character sets or character codes. Use of multilingual text must be predicated on the knowledge of limitations in character rendering inherent in different Decoder models for which text is available.

ANNEX A

Informative

INTEROPERABILITY GUIDELINES

1. SCOPE

This Annex provides a set of guidelines for harmonization and interoperability among MPEG-based systems.

2. PROGRAM MAP TABLE PID

The ATSC Digital Television Standard states that the transport bit stream containing a TS PROGRAM MAP section shall not be used to transmit any other kind of PSI table. Certain non-broadcast applications are known to allow for transport of other PSI tables, both standardized and user private, in streams that carry sections of Program Map Tables.

The syntax of user private sections transported in the PMT PID is specified in accordance with the *ISO/IEC 13818-1 Systems* document. In these applications, the 32-bit CRC is required.

3. TABLE IDS

Certain non-broadcast applications are known to be using table_ID values 0xC0 and 0xC1, and values in the range 0x90 through 0x9A for tables carried in the PMT PID. Applications conforming to the System Information Standard allocate table_ID values as shown in Section 4.2 of the System Information Standard for tables carried in the Network PID. User private tables in the Network PID conform to the syntax described in the network_private_message in Section 4.4 of the System Information Standard. The ownership of these tables is indicated by the format_identifier field in that structure.

4. DESCRIPTORS

Descriptor tag values in the range 0x40 through 0xAF are reserved for use in the ATSC Digital Television Standard. Certain non-broadcast applications are known to be using descriptor tags 0x80 and 0x82 through 0x84. In these applications, descriptor tags in the range 0xC0 through 0xFF are user private (outside the scope of the System Information Standard). The ownership of a user private descriptor is indicated by use of an MPEG registration_descriptor() preceding the user private descriptor.

5. STREAM TYPES

Stream types in the range 0x80 through 0xAF are reserved for use in the ATSC Digital Television Standard. Certain non-broadcast applications are known to be using stream types 0x80, 0x82 and 0xA0. Stream types above 0xC0 are user private.

ANNEX B

Informative

SYSTEM INFORMATION OVERVIEW

1. SYSTEM INFORMATION OVERVIEW

In the transmission systems compliant with this System Information Standard, the Transport Stream (TS) delivers services such as audio, video, text and data, as well as control information used for access control and the creation of user interface screens. Transport Streams compliant with this System Information Standard consist of a continuous sequence of fixed-length transport packets, per *ISO/IEC 13818-1*.

The following sections are informative only, and reflect compliance with packet format and transport multiplex methods and formats described in *ISO/IEC 13818-1*, which may be referenced for details.

2. PACKET FORMAT

Packets are formatted byte-aligned, and may be considered to be composed of a four-byte header followed by a 184-byte payload portion, as shown below.

4 bytes	184 bytes	4 bytes	184 bytes
Header	Payload	Header	Payload

For purposes of understanding this SI message stream protocol, the header contents may be simplified to the following items:

- **Packet ID (PID)**—A 13-bit number uniquely identifying the packet contents. A sequence of packets bearing the same PID is called a *stream* or *PID stream*.
- **Encrypted/Clear**—A two-bit field indicating whether the packet payload is encrypted or in the clear, and also indicating packet parity (used to select one of two keys for decryption).
- **Payload unit start indicator**—A bit indicating whether or not the first byte of payload is a pointer into the payload reflecting the start of the next message. A value of zero in the pointer indicates that a message starts immediately following the pointer byte.

Normally, messages are placed back-to-back in the packet payload, with the first byte of another message directly following the last byte of the previous one. If another message is not to start until the following packet, the remainder of the packet must be padded with bytes of value 0xFF. If the first byte of a message is found to be 0xFF, the remainder of the packet is discarded; at that time the pointer mechanism is used to realign with the start of the next message.

Messages starting in one packet may end in the following packet of the same PID (see Figure 1); that is, messages may span packets.

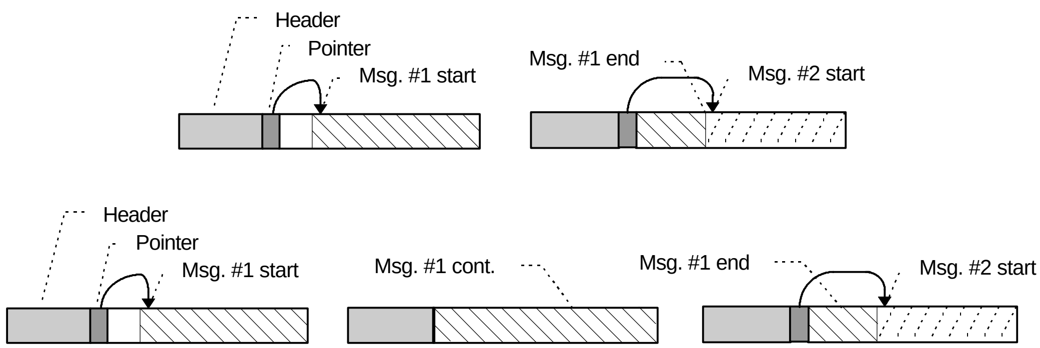

Figure 1. Messages can span packets.

2.1 Service Concept

For the purposes of this System Information Standard, a *service* (MPEG uses the term *program)* is defined as a collection of *elementary* streams, where elementary streams are streams carrying raw compressed video data, compressed audio, subtitles, textual data, or privately defined data channels. Services may consist of several stream components. The most typical is the case in that a service is composed of one video, one or more audio tracks, and zero or more subtitle tracks. A service could consist of just one elementary stream; one single unencrypted data channel is an example. A *radio* service could consist of one audio stream plus one text information stream.

All the component streams comprising a service are typically access controlled and encrypted together. That is, if access is granted to one component, access is granted to all components of the service. If a service is encrypted, it includes a component that carries the parameters related to access control and keys used for decryption. The encryption/control stream is called the ECM stream because it carries ENTITLEMENT CONTROL messages, or ECMs.

Elementary component streams are illustrated in Figure 2.

A service may be defined such that certain components are access controlled differently from others. For example, a data channel may accompany a baseball game and provide player statistics. The data channel stream may be encrypted and access controlled differently from the video/audio components.

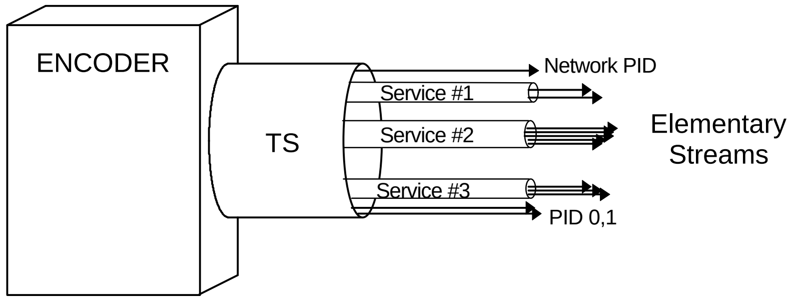

Figure 2. Elementary component streams.

2.2 Stream Types

In Transport Streams compliant with this System Information Standard, the following types of streams are present:

- **Network Stream** — The Network Stream is carried on the Network PID in every TS, and carries data distributed system-wide. The Network Stream carries the NETWORK INFORMATION message, which defines frequency plans, modulation mode tables, satellite and transponder data, and name tables. In addition, the Network Stream carries the VIRTUAL CHANNEL message used to manage access to services for user selection. The Network Stream also carries the SYSTEM TIME message, used to synchronize calendar time.
- **PID 0 (PROGRAM ASSOCIATION Section) Stream** — PID 0 is defined by the *ISO/IEC 13818-1 Systems* document as the stream carrying the PROGRAM ASSOCIATION section. This is the only message that may be present on the stream. Every valid TS has a PID 0. The PROGRAM ASSOCIATION section provides a simplified directory of services on the TS by listing each by MPEG program_number and PID carrying the TS PROGRAM MAP section. The PROGRAM ASSOCIATION section also defines the Network PID.
- **PID 1 (CONDITIONAL ACCESS Section) Stream** — PID 1 is also defined by *ISO/IEC 13818-1*, and carries only the CONDITIONAL ACCESS message, used to define the PID streams carrying ENTITLEMENT MANAGEMENT messages (EMMs) on the TS.
- **Program Map Stream** — Every service on the TS has an associated MPEG Program Map PID stream. The Program Map stream carries the *ISO/IEC 13818-1* TS PROGRAM MAP message, which identifies the PIDs, types, and languages for all elementary streams associated with the program. In some applications, the Program Map stream also carries program-related messages that deal with user interface functionality, such as text defining the program name or offering the program on a pay-per-view basis.
- **ENTITLEMENT CONTROL Message (ECM) Stream** — Entitlement control messages define the program's access requirements.

- **ENTITLEMENT MANAGEMENT Message (EMM) Stream** — ENTITLEMENT MANAGEMENT messages, or EMMs, define access rights for each individual Decoder.
- **Component Streams** — Each service is composed of a number of component streams. Typically these include the video stream, one or more audio streams, one or more subtitle streams, and possibly a data stream. The composition of some services may not include video; an example would be a radio service. One or more subtitle streams or a text stream may be present as well.

2.3 Overview of Relationship Between Streams

Figure 3 shows the relationship between the Network Stream, the ISO-defined streams, the Program Map stream, PID 0 and 1, the EMM streams, and the component streams.

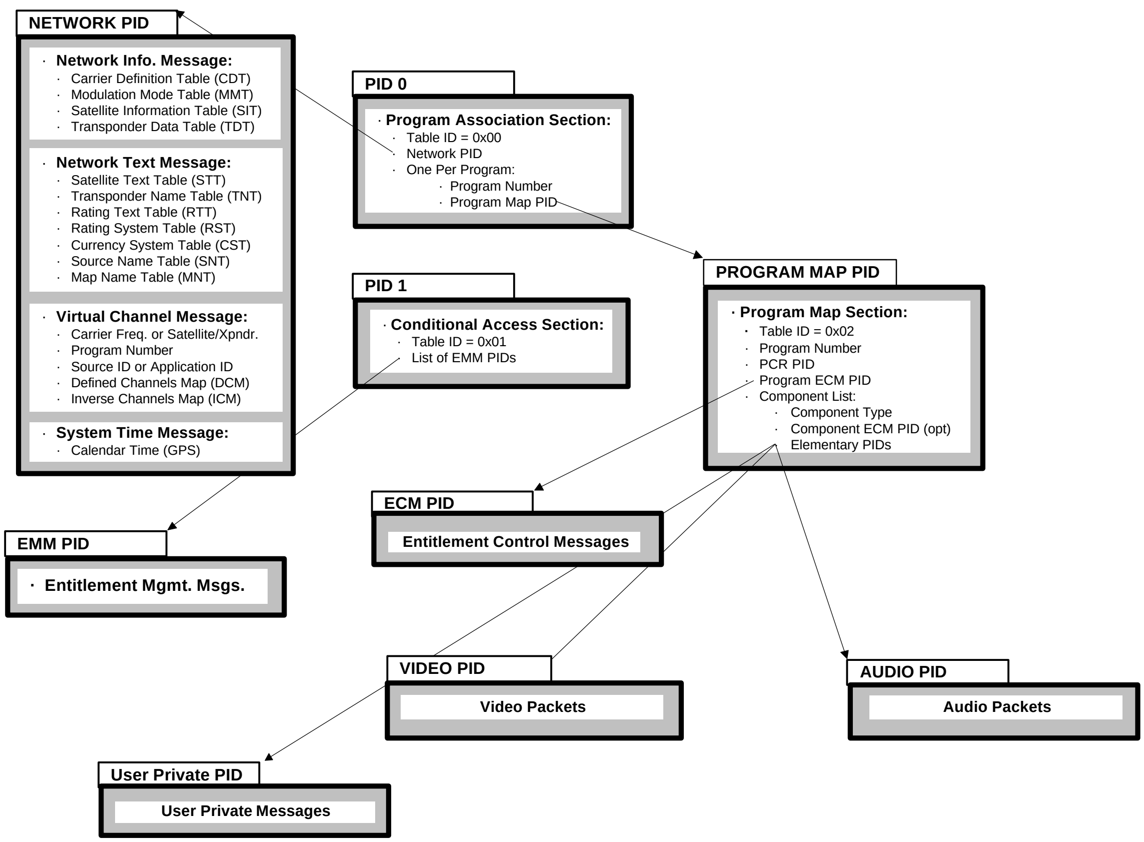

Figure 3. Transport stream relationships.

3. NETWORK INFORMATION

Data delivered on the Network Stream defines information of system-wide applicability such as frequency plans, satellite names and orbital locations, and transponder-specific data such as waveform type carried and name. NETWORK

INFORMATION, NETWORK TEXT, and VIRTUAL CHANNEL messages each carry a portion of one of the following types of tables:

- Carrier Definition Table (CDT)
- Modulation Mode Table (MMT)
- Satellite Information Table (SIT)
- Satellite Text Table (STT)
- Transponder Data Table (TDT)
- Transponder Name Table (TNT)
- Virtual Channel Tables (VCTs)
- Ratings Text Tables (RTTs)
- Rating System Table (RST)
- Currency System Table (CST)
- Source Name Table (SNT)
- Map Name Table (MNT)

These tables are described in the following sections.

3.1 Carrier Definition Table (CDT)

The Carrier Definition Table provides a foundation for the definition of frequency plans by defining a set of carrier frequencies appropriate to a particular transmission medium. The CDT is stored in the Decoder as an array of as many as 255 CDT records, each consisting of:

- Carrier frequency, 15 bits, in units of 10 or 125 kHz

Frequencies in excess of 4 GHz can be represented. For satellite carriers, the specified frequency represents the downconverted value. If a carrier on one satellite shares the definition with one on another satellite (as they often will), those two carriers will refer to one common definition (they share a common CDT record).

Changes to the table are rare, and will only occur in conjunction with the launching of new satellites providing carriers at nonstandard frequencies, or introduction of new frequency allocations by the FCC.

3.2 Modulation Mode Table (MMT)

The Modulation Mode Table provides a foundation for quick acquisition of digitally modulated waveforms. A separate MMT shall be transmitted in Network data for each transmission medium supported by that network. An MMT is stored in the Decoder as an array of up to 255 MMT records, each consisting of:

- Modulation format: QAM, QPSK, BPSK, offset QPSK, VSB

- Transmission system: ITU-T (ETSI Standard), ITU-T (North America), ITU-R (ETSI Standard), DigiCipher®, ANSI
- Symbol rate, in units of 1 Hz
- Inner coding mode, expressed as either “none” or an integer ratio such as 1/2 or 3/4
- For QAM modulation, the number of levels
- Flag indicating whether or not the bit stream is split into separate I and Q components

Each MMT contains entries for each modulation mode currently in use by any digital waveform, plus entries for any modes anticipated to be used. As with the CDT, changes to the table are rare.

Parameters defined within the MMT are not specifically manipulated by Decoders compliant with this System Information Standard, but are referenced by the Decoder when attempting to acquire a digitally encoded and modulated waveform.

3.3 Satellite Information Table (SIT)

The Satellite Information Table provides information on each satellite defined in the network, including:

- Orbital position, 11 bits, 0.1° resolution
- Number of transponders/carriers (maximum 64)
- Frequency band, 2 bits: C or Ku (BSS) or Ku (FSS)

Satellite name information is delivered in the Satellite Text Table (STT).

Satellite Integrated Receiver-Decoders (IRDs) may process SIT and TDT network data to provide the user access to new satellites as they are launched, or to track their new locations if orbital positions are moved.

3.4 Satellite Text Table (STT)

The Satellite Text Table is structured much like the Satellite Information Table, except that it consists of only the abbreviated (reference) name of the satellite (usually two or three characters), as well as the full satellite name. The STT is delivered multilingually, filtered by language in the Decoder.

3.5 Transponder Data Table (TDT)

For each transponder on each satellite, the TDT defines:

- The carrier frequency (in the form of a reference to a satellite CDT)
- Polarization: left/right or horizontal/vertical (as appropriate)
- The transport type: MPEG-2 or not

- For carriers not carrying MPEG-2 transport, the type of waveform carried (NTSC, PAL, or other digital or analog standard)
- The modulation mode data for the transponder, in the form of an MMT reference; applicable only for transponders carrying digitally modulated waveforms.
- An association to a specific VCT, in the form of a VCT ID reference; also applicable for compliant waveforms only
- Also for channels carrying MPEG-2 Transport Streams compliant with this System Information Standard, indicates if the transponder is the root transponder of a VCT
- For clear channels, the audio mode (subcarrier, matrix, mono, digital, narrow, wide, etc.)

Note that, for the sake of completeness, this System Information Standard defines carriers on transponders carrying non-MPEG waveforms, for the benefit of IRDs capable of processing these waveforms. Such IRDs may make use of network data carried on a Standard MPEG-2 TS to assist them in navigation and processing of all available video services available via satellite, not just those carried on MPEG-2 Transport Streams.

The system defined in this System Information Standard supports multiple carriers per transponder (MCPT) operation, hence the total number of carriers on one satellite can exceed the number of transponders on that satellite. For satellite in which MCPT carriers are defined, the carriers are numbered such that the first 24 (for C band) or 32 (for Ku band) transponders defined correspond to the actual physical transponders. Any additional “transponders” defined are actually additional carriers on one or more of the transponders. Up to 64 transponders/carriers can be defined per satellite. Transponder name information is delivered in the Transponder Name Table (TNT).

3.5.1 Satellite ID

SIT records are delivered tagged with a *satellite ID*, an 8-bit number used to index the stored array. Where other data structures refer to a satellite, the satellite ID is used in place of the satellite name, for convenience and storage efficiency.

3.5.2 Waveform Type

The waveform type indicates the type of waveform primarily carried by the transponder, if known. For those transponders that change waveform type during the course of a day, the waveform type will be less useful.

3.6 Transponder Name Table (TNT)

The transponder name is the textual name associated with the transponder (or MCPT carrier), and can apply to either scrambled or (if the system supports them) clear transponders. If the transponder is a root transponder for a Virtual Channel Table, the transponder name represents the program provider or operator of that table, and is (in the case of multiple tables) displayed along with the satellite, transponder, and virtual channel number and name.

The TNT is delivered multilingually in the NETWORK TEXT message.

3.7 Virtual Channel Table (VCT)

The Virtual Channel Table (VCT) is a data structure consisting of up to 4096 channel definition records. Section 4 of this Annex provides an overview of virtual channels and their purpose and usage. Each VCT is associated with a 16-bit reference ID field. Each record in the VCT consists of:

- For satellite applications, the channel's transponder, indicating which transponder carries the channel
- For terrestrial applications, the channel's carrier frequency (via reference to the CDT) and modulation mode (via reference to the MMT)
- For satellite applications, the channel's satellite ID
- The MPEG program number, associating the virtual channel record with a service defined in the PROGRAM ASSOCIATION and TS PROGRAM MAP sections
- For virtual channels associated with programs carried in a program guide, the *source ID*, a number that may be used to link the virtual channel to entries in the Interactive Program Guide (IPG) database
- For virtual channels used as access paths to application code or data (such as IPG), the *application ID*[1]
- An optional direct frequency specification, in case the CDT does not include an entry for the desired frequency (typically used in some satellite SCPC applications)
- An optional direct specification of Transport Stream symbol rate, in case the MMT does not include an appropriate rate

3.7.1 Virtual Channel Table ID

Incoming VCT data is tagged with a 16-bit VCT *ID*, a number identifying the specific VCT to which the data is to be associated.

Many VCTs may be defined in the satellite system, although for terrestrial applications only one will be used per system. If all programmers carrying digital services on a particular satellite could agree to a map definition, there could be just one VCT on that satellite. If programmers do not agree to cooperate, each may define their own VCT.

[1] Source ID and application ID need never be defined in the same virtual channel record, therefore they share a common 16-bit field in the stored map. Channels are defined as for “application access” or not; if they are application access, the field defines the application ID, if not, it defines the source ID.

3.7.2 Transponder

As mentioned, the root transponder may not carry all virtual channels in the VCT. The transponder field in the virtual channel record identifies which transponder carries the virtual channel.

3.7.3 Satellite

For consumer Decoders, VCTs generally do not span satellites; that is, all channels defined in one map are carried on the same satellite. If a particular virtual channel references a satellite other than the root satellite, and that satellite is not available to the receiver without dish movement (that is, it is not co-located), a consumer receiver should treat the channel as unavailable.

Commercial receivers can accept multiple L-band inputs, selected via relay. Satellite references within the VCT accommodate this case.

3.8 Source ID

Source ID is a 16-bit number associated with each program source, defined in such a way that every programming source offered anywhere in the system described in this System Information Standard is uniquely identified. For example, HBO/W has a different assigned source ID than HBO/E, and both are different than HBO-2 or HBO-3. Uniqueness is necessary to maintain correct linkages between an IPG database and virtual channel tables. See Section 4.9 of this Annex for a discussion of the relationship between source_ID, virtual channels, and an IPG database.

3.9 Source Names and Source Name Table (SNT)

The Source Name is a variable length multilingual text string associating a source ID with a textual name. The Source Name Table (SNT) is delivered within the NETWORK TEXT message.

Source name information is delivered in a message format separate from the message containing other information comprising the virtual channel table. Name information is not strictly necessary for channel acquisition, and (depending on the memory management scheme employed in the receiver) may not always be available from memory at acquisition time. Source name information is refreshed often, and is typically available within several seconds of acquisition.

An IPG database may define textual reference names associated with given program sources (referenced by source ID). Such a database may be used to derive virtual channel names in some applications, though in an IPG database the name is generally abbreviated due to display considerations.

Name data is, unlike the regular VCT data, language tagged, so that multilingual source names may be defined. Transmission format for multilingual text is defined to include references to multiple phonetic and ideographic character sets.

3.10 Defined Channels Map (DCM) and Inverse Channels Map (ICM)

For a given Standard-compliant channel, DCM data consists of a series of bytes that, taken as a whole, define which channels in the map are defined, and which are not.

Each Virtual Channel Table has associated with it a table listing source_IDs and their associated virtual channel numbers. The source_ID values are sorted by value from lowest to highest in the table, to facilitate (using a binary search) lookup of a virtual channel given a source ID.

3.11 Ratings Text Table (RTT)

Programs defined in conformance with this System Information Standard may be characterized by their "ratings" parameters. Ratings information is typically utilized in the Decoder's user interface to provide parental control over the kinds of programming for which viewing is allowed. A typical rating parameter in the United States is the one provided by the Motion Picture Association of America (MPAA) for particular movie titles. The system described in this System Information Standard supports up to 256 "rating regions," where a rating region is defined as one subset of the full population of Decoders that conform to a particular standard definition of rating definitions and interpretations.

The MPAA rating is one possible rating dimension; other dimensions are possible. HBO and others, for example, are using a set of program ratings specified in a publication called the *Expanded EDS Program Ratings Specification*, which includes ratings for adult language, and sexual and violence contents.

Decoders designed in conformance with this System Information Standard shall adapt to changes in the definition of ratings in a dynamic fashion. The user interface, which allows the user to set ceilings on each of the various dimensions, is table driven from the RTT, so that changes are automatically accommodated. The change might involve addition of a new dimension, addition of a new level in one dimension, or a change of name of any level or dimension.

3.12 Rating System Table (RST)

Each Decoder is assigned to a rating region, either through an installation or user-preference option, via factory configuration, or via a private message individually addressed to the Decoder.

To support installation or user setup dialogs within the Decoder, each rating region is associated with a descriptive text string via the Ratings System Table. A Standard-compliant television or Decoder may use the RST to offer the user a choice from among several rating regions. The RST defines a list of text strings, each of which is associated (by its index into the table) with a particular rating_region currently in use in the system. In the U.S., the predominant rating system is the Motion Picture Association of America.

3.13 Currency System Table (CST)

Each Decoder is assigned to a currency region, again either through an installation or user-preference option, via factory configuration, or via a private Decoder-specific message.

The Currency System Table defines a list of text strings, each of which is associated (by its index) with a particular currency_region currently in use in the system. If more than one currency region is supported in a given network, Decoders may use the CST to create a user interface dialog to allow the user to select a preference for currency system. In many applications, there will be a one-to-one correspondence between currency region and country, but such a correspondence is not required.

3.14 Map Name Table (MNT)

The Map Name Table provides a multilingual textual name for a given Virtual Channel Table. For satellite media, the name of the root transponder may act as the map name, but for other media, such as broadcast, the MNT provides this function.

3.15 Overview of Downloaded Tables

Figure 4 and Figure 5 show the relationships between the tables of the network data stream for the satellite and cable media cases, respectively. Figure 6 diagrams the relationship between VCTs and an IPG database (shown in simplified form) for the case of over the air broadcast.

4. VIRTUAL CHANNELS

Each Standard-compliant Transport Stream may deliver a large number of television, audio, text, and data services. Virtual channels offer the user a consistent view of services available on any particular Standard-compliant TS, and provide a convenient reference mechanism for various combinations of services (such as television with stereo or secondary audio).

Each virtual channel number defines the access point for a particular programming source. The access point for the target programming is described in terms of its Transport Stream location and *ISO/IEC 13818-1* program_number. Transport Stream location is given as satellite and transponder (for satellite IRDs) or its carrier frequency (for cable). The virtual channels concept also supports non-video components such as data, audio-only (radio), and text (or text plus audio).

4.1 Virtual Channel Table

Upon acquisition of an MPEG-2 Transport Stream compliant with this System Information Standard, the Decoder is provided information defining the VCT, a data structure that maps MPEG-2 and non-MPEG-2 services to user channel numbers, and provides the access path to those services. For satellite transmission, the VCT takes the form of a satellite/transponder reference and (for MPEG-2 programs) MPEG program number. For terrestrial applications, the VCT takes the form of a reference to the Carrier

Definition Table (CDT), Modulation Mode Table (MMT), and (for MPEG-2 programs) MPEG program number. The Decoder may store one or more VCTs in RAM to allow direct acquisition of any given channel.

Carrier Definition Table (CDT)
(for Satellite)

	FREQUENCY (MHz)
0	-
1	1430.0
2	1410.0
3	1390.0
4	1370.0
5	1350.0
	....
255	5909.5

Modulation Mode Table (MMT)
(for Satellite)

	SYM. RATE	CODING MODE	SPLIT	MOD FMT.
0	-	-	-	-
1	19.27	3/4	Y	QPSK
2	19.27	3/4	Y	QPSK
3	19.27	1/2	Y	QPSK
4	19.27	1/2	Y	QPSK
	....	....	....	....
255	19.51	3/4	N	QPSK

Satellite Information Tables (SIT)

SAT ID	ABBRV. NAME	FULL NAME	ORBITAL POS.	NUM. XPNDR.	FREQ. BAND
0	S2	Spacenet 2	69 °W	24	C

Transponder Name Tables (TNT)

	POL.	WVFM. TYPE	CDT REF.	MMT REF.	VCT REF.	ROOT	MTRX MODE	WIDE BW	SC #1	SC #2	TRANSPONDER NAME
1	H	Unk.	1	-	-	-	Mono	Y	6.2	-	Occasional
2	V	Inacc.	2	-	-	-	-	-	-	-	
3	H	Std.	3	2	1	N	-	-	-	-	(Netlink S2)
4	V	Clear	40	-	-	-	Mtrx.	N	6.2	6.8	NASA
5	H	VCII+	41	-	-	-	-	-	-	-	KBL Pittsburgh
6	V	Std.	48	101	0	Y	-	-	-	-	Netlink S2
7	H	Std.	49	30	6	N	-	-	-	-	(Netlink S2)
		....	....	....	....	....	....	....	....	....	....
24	V	Std.	229	29	7	Y	-	-	-	-	Action PPV

Virtual Channel Tables (VCTs)
(for Satellite)

VCT ID
1

	TRANSPONDER	SATELLITE ID.	PROGRAM NO.	SOURCE ID / APP. ID
1	2	0	0x37FF	0x0020
2	2	0	0x2EA9	0x0030
3	6	0	0x0FF2	0x0045
4	12	0	0x121E	0x0067
5	12	0	0x330A	0x0121
6	6	0	0x288E	0x0890
7	6	0	0x288F	0x0045
	....	....	....	....
4095	7	0	0x2003	0x1030

Source Name Table (SNT)
(for Satellite)

SOURCE ID	SOURCE NAME
0x0047	Cineplex
0x0045	WABC
0x0121	WCBS
0x0030	WNBC
0x0890	Discovery
0x1478	E!
0x1479	WIND
---	---

Figure 4. Network data relationships — satellite case.

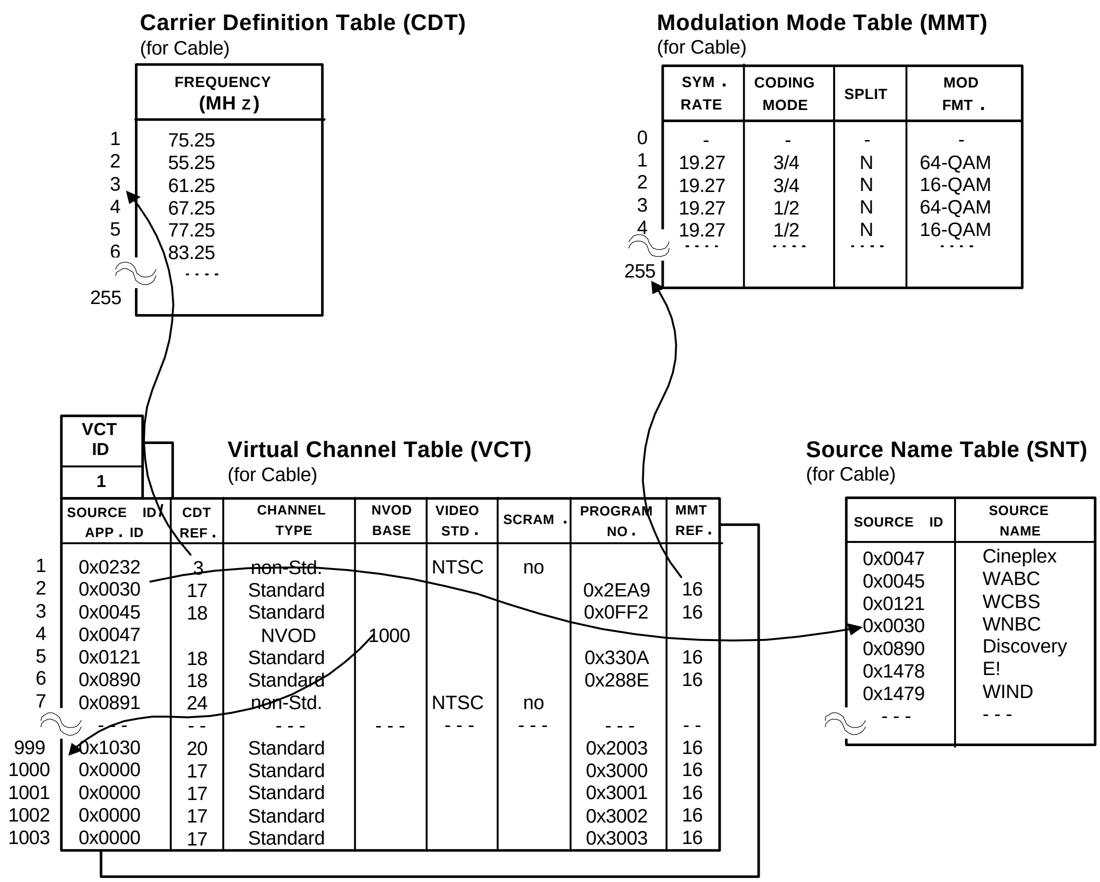

Figure 5. Network data relationships — cable case.

4.2 Defined Channels Map

VCT records are downloaded for defined channels only. Virtual channels for which no definition is available are assumed to be *null* (unavailable) channels. Given that certain channels may cease to exist at some point in time, a mechanism is provided to *undefine* channels no longer in use. The Defined Channels Map (DCM) performs this function by indicating, for a given map, which channels are defined and which are not. The receiver can use this information to eliminate channels no longer in use from the specified VCT.

4.3 Multiple Virtual Channel Tables

In typical cable, SMATV, MMDS, and over-the-air types of networks, any single Decoder will store and process one single VCT. The VCT used within Decoders in one cable system must differ from that used in another, of course, if the systems offer different channel lineups or number their channels differently. The C/Ku-band satellite IRD, on the other hand, must be able to store and process multiple VCTs. The 16-bit VCT_ID is used to differentiate one VCT from another in this System Information Standard.

Interactive Program Guide (IPG)
(for Broadcast)

SOURCE ID	SOURCE NAME	PROGRAM SCHEDULE
0x0047	Cineplex	6pm: Popeye
0x0045	WABC	6:30: Local News; 7:00...
0x0121	WCBS	6pm: CBS Eve. News;
0x0030	WNBC	6pm: NBC News with T...
0x0890	Discovery	6:30: Mayan Ruins; 7:00.
0x1478	E!	7:00: Talk Soup; 8:00...
0x1479	WIND	6:00: Paul Runyan; 7:00.
- - -	- - -	- - -

VCT ID
1

Virtual Channel Table (VCT)
(for Broadcast)

	PROGRAM NO.	SOURCE ID/ APP. ID	FREQ.	CHANNEL TYPE	NVOD BASE	VIDEO STD.	SCRAM.
1		0x0232	56.0MHz	non-Std.		NTSC	no
2	0x2EA9	0x0030	62.0MHz	Standard			
3	0x0FF2	0x0045	92.0MHz	Standard			
4		0x0047	74.0MHz	NVOD	1000		
5	0x330A	0x0121	80.0MHz	Standard			
6	0x288E	0x0890	80.0MHz	Standard			
7		0x0891	98.0MHz	non-Std.		NTSC	no
	- - -	- - -	- -	- - -	- - -	- - -	- - -
999	0x2003	0x1030	68.0MHz	Standard			
1000	0x3000	0x0000	68.0MHz	Standard			
1001	0x3001	0x0000	68.0MHz	Standard			
1002	0x3002	0x0000	68.0MHz	Standard			
1003	0x3003	0x0000	68.0MHz	Standard			

Figure 6. Network data relationships — broadcast case.

One particular programmer may own and operate several uplinks carried on several transponders, aimed at one particular satellite. With the virtual channel definition, the user can access any of the services carried by any of this programmer's uplinks by selecting its *root* transponder and then identifying the channel number of interest. Hidden from the user, the receiver may actually tune to a different transponder to access the requested virtual channel.

Figure 7 shows channel mapping for a hypothetical C-band satellite called Galaxy 9. In the example, two program providers, *B/W Classics* and *Sports TV*, both uplink Standard-compliant waveforms on several transponders. B/W Classics' root transponder is 6, and Sports TV's is 2.

A portion of the Virtual Channel Table for B/W Classics is shown at the left side of the figure. Note that transponders other than the root transponder are associated with certain channels. As the user scans virtual channels, the receiver must tune and acquire the associated transponder, but hide from the user the actual transponder number tuned, displaying instead the root transponder. Furthermore, if the user hits "channel up" at the highest non-hidden virtual channel, the transponder next accessed should be 4, not 20.

Virtual Chan.		Disp. Chan.	Act. Chan.
1	Comedy Classics	6	4
2	Soap Operas	6	4
3	Suspense	6	4
4	Action	6	4
5	Variety	6	6
6	Director’s Choice	6	6
7	Sitcom Alley	6	4
8	Family Hour	6	6
9	News & More	6	19
10	1950’s Revisited	6	4
11	Motown Music	6	19
...			
101	Se Anunciara	6	21
102	Buenos Dias	6	21
103	Jacabo Zabludob	6	21
...			
201	Kartoon Korner	6	21
202	Mickey Mouse	6	21
203	Howdy Doody	6	19
...			
610	Time Tunnel	6	6
611	Space: 1999	6	6
612	Twilight Zone	6	21
...			
999	Guide	6	19

G9

Galaxy 9

103.5° W.

Xpndr.	
1	
2	**Sports TV**
3	
4	(B/W Classics)
5	
6	**B/W Classics**
7	(Sports TV)
8	
9	
10	
11	
12	
13	
14	
15	(Sports TV)
16	
17	
18	
19	(B/W Classics)
20	
21	(B/W Classics)
22	
23	
24	(Sports TV)

Virtual Chan.		Disp. Chan.	Act. Chan.
1	PBA Bowling	2	2
2	Flywt. Boxing	2	2
3	Hvywt. Boxing	2	2
4	Sportfishing	2	2
5	River Rafting	2	2
6	Great Outdoors	2	7
7	NFL Football	2	2
8	Hi Sch. Football	2	7
9	Gymnastics	2	7
10	Horse Racing	2	15
11	Equestrian	2	15
12	Hockey Hilights	2	7
13	Harness Racing	2	15
14	Baseball	2	15
15	Rugby	2	15
...			
80	Downhill Skiing	2	24
81	Snowmobiles	2	24
82	Soccer	2	24
...			
201	Hockey (Fr.)	2	15
202	Soccer (It.)	2	24
203	Water Polo	2	2
...			
999	Guide	2	24

Figure 7. Satellite virtual channels example.

4.4 Changes to a Virtual Channel Table

An important and powerful aspect of the virtual channel idea is the notion that virtual channel definitions can change dynamically. A programmer may want to routinely reconfigure an uplink Encoder to carry more or fewer video services. When this happens, by making changes to the definitions carried in the VCTs, the programmer can cause any Decoder that has acquired a virtual channel that will be moving to retune to the service’s new destination. Or, if the service will be unavailable, to a channel carrying an informational text screen.

Using the virtual channel concept, a program provider is free to move services from one physical place to another on the satellite while keeping the virtual channel number constant. For example, a video service may be moved from one transponder or cable carrier to another without changing the user's method of accessing the channel. If the definition of the currently acquired virtual channel changes, the Decoder reacquires using the changed data.

4.5 Broadcast Virtual Channel Tables

Within one geographic area, a number of digital transport streams may be transmitted VSB-modulated over UHF carriers. Broadcasters in a given area may cooperate among themselves to define and name a virtual channel map, to provide consumers with consistent and flexible access to digital services.

In the simplified example shown in Figure 8, broadcasters in San Diego transmit three carriers, two digital and one analog:

1. centered at 599 MHz, a Standard digital transport stream which encodes two programming services (MPEG program_numbers 0x0890 and 0x0671),
2. at 671 MHz, a Standard digital transport which encodes a single programming service, and
3. with picture carrier at 61.25 MHz, an analog NTSC waveform.

VCT #1 is defined for the San Diego area, and consists of four virtual channels, numbered 3, 6, 7, and 10. Channel 3 corresponds to the numerical designation of the 60-66 MHz channel band as defined by the FCC. Channels 6 and 7 are carried together on a transport stream at 599 MHz, and channel 10 is carried on a TS at 671 MHz.

In Los Angeles, a couple of hundred miles to the north, three carriers are also broadcast:

1. at 719 MHz, a Standard digital transport stream which encodes two programming services (MPEG program_numbers 0x2EA9 and 0x0FF2),
2. at 497 MHz, a Standard digital transport which encodes a single programming service, and
3. at 77.25 MHz, an analog NTSC waveform.

In Los Angeles, users access programming on channels 3, 4, 5, and 7. Channel 5 corresponds to the numerical designation of the 76-82 MHz channel band as defined by the FCC.

The Figure shows an IPG database which references the VCTs via source_ID. Channel 6 in San Diego and Channel 3 in Los Angeles are both CSPAN; only one IPG record is sufficient to describe CSPAN, and the IPG data for CSPAN is usable with either map.

Consumers who live partway between Los Angeles and San Diego may be able to receive all or portions of the analog and digital television services available from either locale. The Standard-compliant receiver must therefore accommodate multiple virtual channel maps, and afford the user the ability to navigate within each map.

A method must be provided to allow the user to switch between maps, and each should be labeled on-screen to avoid possible channel numbering ambiguity. Note that in the example channel 3 exists in both maps — the display should include the map name in order to resolve this ambiguity.

Interactive Program Guide (IPG)
(for Broadcast)

SOURCE ID	SOURCE NAME	PROGRAM SCHEDULE
0x0047	Cineplex	6pm: Popeye
0x0045	WTBS	6:30: Atlanta Braves vs.
0x0121	WCBS	6pm: CBS Eve. News;
0x0030	WNBC	6pm: NBC News with T...
0x0890	CSPAN	6:30: Congressional Upd.
0x1478	E!	7:00: Talk Soup; 8:00...
0x1479	WIND	6:00: Paul Runyan; 7:00.
- - -	- - -	- - -

Virtual Channel Table #1

VCT ID	MAP NAME
1	"San Diego"

VIRTUAL CHAN NO.	PROGRAM NO.	SOURCE ID/ APP. ID	FREQ.	CHANNEL TYPE	VIDEO STD.	SCRAM.
3		0x0047	61.25MHz	non-Std.	NTSC	no
6	0x2EA9	0x0890	599.0MHz	Standard		
7	0x0FF2	0x0671	599.0MHz	Standard		
10	0x330A	0x1199	671.0MHz	Standard		

Virtual Channel Table #2

VCT ID	MAP NAME
2	"Los Angeles"

VIRTUAL CHAN NO.	PROGRAM NO.	SOURCE ID/ APP. ID	FREQ.	CHANNEL TYPE	VIDEO STD.	SCRAM.
3	0x2EA9	0x0890	719.0MHz	Standard		
4	0x0FF2	0x0343	719.0MHz	Standard		
5		0x267A	77.25MHz	non-Std.	NTSC	no
7	0x330A	0x1EF0	485.0MHz	Standard		

Figure 8. Broadcast virtual channel tables example.

The benefits of Virtual Channel Tables for the over-the-air broadcast are similar to those for satellite. The user is provided consistency in navigation and access to programming, and is insulated from knowledge of the actual physical location of a given program source. The broadcaster is given the freedom and flexibility to move carriers from frequency to frequency, to move services from one multiplex to another, and to number services in a flexible and user-friendly manner.

It may be that a broadcaster transmits two or more digital streams in a local area. Given that flexibility in allocation of bandwidth to different program sources is one benefit of digital transport, the broadcaster may wish to move programming from one TS to another, either permanently, or periodically. Where one day, two transport streams each carry four SDTV programs, another day the allocation could be changed such that one carries two and the other six. Or, one could be switched to HDTV or back again to SDTV.

Virtual channels are helpful as well when analog carriers are eventually replaced by digital transport streams. Carrying the above example further, at one point in time, the analog carrier in San Diego on channel 5 could be replaced by a Standard digital TS centered at 79 MHz. Digital receivers would process a change to the virtual channel map definition for “channel 5” pointing them to the channel 5 MPEG program on the new digital TS, and see no apparent change in access.

4.6 Hidden Virtual Channels

A virtual channel may be defined as *hidden* if it is necessary to define a channel that is accessible via means other than the usual direct channel number entry or channel up/down (*surfing*). The GUIDE button on the remote control could be, for example, tied to such a hidden channel to provide the access path to the service carrying the guide database (see the following section).

4.7 Access to Application Data Streams via Virtual Channel

Each virtual channel record provides, in addition to the physical location of the associated Transport Stream, and the *ISO/IEC 13818-1* program number within that TS, a 16-bit identifying tag. Depending upon the type of channel, this tag indicates either the channel’s associated “source ID” or its associated “application ID.” The source ID is present for channels that carry programming described by records in a program guide database, and acts as a linkage to an IPG. Refer to Section 4.9 of this Annex for a description of this linkage and its use.

The application ID tag is present for channels marked as “application access” type. One example of an application that uses this tag is a built-in interactive program guide function. Each application that may be resident in the system (either ROM-based or downloaded dynamically) has assigned to it a unique 16-bit application ID.

To follow the IPG example further, if guide provider X wishes to transport messages carrying his program guide database on a service within a Transport Stream somewhere in the system, he may create a MPEG program and one or more component streams within it, on a selected TS. A hidden virtual channel of the “application access” type is also created, and tagged with an application ID unique to provider X and this specific application (program guide data, in provider X’s transmission format).

The application code can gain access to its data stream by searching the Virtual Channel Table for application access channels that match its application ID, and then acquiring the service pointed to by that virtual channel record. The application ID for any given application is hard-coded into the application itself. As mentioned above, if the Decoder’s remote control unit has a GUIDE button, that button could be hard-wired to one specific application ID.

More than one IPG data service may be available in the system. If this is the case, it may be that a single application (capable of handling streams of different formats, or streams in a common format but originating from different providers), can find and process services associated with more than one application ID. The user may be offered a choice of providers. Alternatively, if one program guide application can be replaced in the Decoder with another, whichever guide application is currently present will acquire its own data stream.

4.8 Replicated Services

The transmission system described in this protocol supports the notion that more than one service may be decrypted simultaneously, even when each are separately access controlled. The service that carries the electronic program guide data, for example, may be replicated on many Transport Streams; a Decoder may be designed to access and decrypt this data in a "background" manner during processing of a standard audio/video service.

Definition of a replicated service involves the following:

- One virtual channel record (per map) defines the service's *primary* entry point.
- Access to the primary entry point, like normal channel access, involves tuning to the transponder associated with the virtual channel.
- If a service is carried on a Transport Stream in addition to the one defined as the primary entry point, a virtual channel defining a local access point may be defined. The local access record:
  - Is identified with the primary service in that the channel number is the same as the primary entry point virtual channel number.
  - Provides a definition used to access a copy of the service within a Transport Stream (defined within the local access record) other than the primary one.

4.9 Virtual Channels and an IPG Database

As mentioned, virtual channels may be tagged with a source_ID, used to associate the channel with entries in an IPG database. Any virtual channel that has (or may have) associated records in an IPG database that utilizes the source ID to uniquely associate the channel with its program source *must* have a source_ID specified. The defined association between each programming source and its source_ID is consistent and defined at the system level for the benefit of any IPG databases which may with to utilize the relationship.

The source_ID can link an IPG database to the Virtual Channel Tables in the following way. If a user is browsing through the interactive guide looking for current programs, he may stop, highlight a certain program, and select it for viewing. The selected program is associated with other programs offered by that programmer, and with the textual name of the programmer or channel itself by the source_ID fields within the database.

The selection process therefore results in the need to find a virtual channel associated with the selected source_ID. The Decoder must search through the VCT looking for a match on source_ID, or (because it is faster) search through the inverse channel maps using a binary search algorithm. Once a match is found, the service can be acquired by following the access path defined in the virtual channel record.

5. REPRESENTATION OF TIME

The International Bureau of Weights and Measures maintains International Atomic Time (TAI)[2], a constant rate time standard based on a number of cesium-based atomic clocks. Cesium beam frequency standards are also used onboard the Global Positioning Satellites (GPS), whose coordinated outputs provide *GPS Time*, another constant rate time standard.

At its inception in 1980, GPS time was usable directly to set time-of-day clocks throughout the world. Due to the gradual slowdown of the Earth as it moves through its solar orbit, however, the length of a year in GPS time has drifted away from the length of a year as measured by the Earth's rotation. Based on observed rotational anomalies, leap seconds are inserted or deleted as required from TAI to yield Universal Coordinated Time, or UTC. Time of day clocks throughout the world are aligned to UTC time. GPS satellites also broadcast the offset between the constant rate GPS time and UTC. Currently, GPS runs approximately 10 seconds ahead of UTC. UTC is what we currently think of as *standard* time (formerly called *Greenwich Mean Time* [GMT]), and is referenced to the time zone at Greenwich, England.

This protocol defines various time-related events and activities, including starting times for programs, text display, changes to VCTs, and others. Standard-compatible headend equipment is synchronized to GPS time by the incorporation of one or more GPS time clocks.[3] The Standard-compliant Decoder is synchronized to system time by the SYSTEM TIME message, which provides time in the form of GPS seconds since week zero of GPS time, January 6th, 1980. The message also provides the current GPS/UTC offset, in whole seconds.

5.1 System Time

GPS satellites typically output GPS time in a format consisting of a week count (Tw) and a seconds within the week count (Ts), where week zero is defined as starting January 6th, 1980. For purposes of building the SYSTEM TIME message, the following formula may be used:

$$T = (Tw * 604{,}800) + Ts$$

There are 604,800 seconds per week.

When converting between GPS seconds and current local time in hours/minutes/seconds, the following factors must be taken into account:

- **GPS to UTC offset —** Given a time represented as GPS seconds, the Decoder first subtracts the GPS/UTC offset to convert to UTC.

[2] The ordering of the letters in the acronym is based on the French language representation.

[3] Companies making such equipment include Trimble Navigation, Trak Microwave, Datum Inc., Socket Communications, and many others.

- **1980 —** The first year of GPS time started on January 6th, yielding 361 days in the first year (1980 was also a leap year).
- **Leap years —** The number of leap years that occurred between the current GPS second and 1980 must be accounted for. A leap year is a year whose number is evenly divisible by four, or, in the case of century years, by 400.

NOTE: According to this rule, the year 2000 *will* be a leap year even though it is a century year, because it is also divisible by 400.

- **Time zones —** Time zones are signed integer values in the range -12 to +13 hours, where positive numbers represent zones east of the Greenwich meridian and negative numbers west of it. Pacific Standard Time (PST) is 8 hours behind standard time, and Eastern Standard Time (EST) is 5 hours behind. The system defined by this System Information Standard accommodates time zones that are not an integral number of hours offset from Greenwich by defining time zone as an 11-bit signed integer number in units of minutes. To convert to local time, the time zone is added to Greenwich time using signed integer arithmetic.
- **Daylight savings time —** If applicable, daylight savings time must be taken into account. On a unit by unit basis, each Decoder may be given a definition for when daylight savings time is entered into in Spring, and when it is exited in Fall. Entry/exit points are given as absolute times (GPS seconds), and hence are given in one second resolution.

5.2 Transmission Format for Event Times

In this messaging protocol, the absolute time of action is specified for most events in terms of an unsigned 32-bit integer number, the count of GPS seconds since January 6th, 1980. This count will not wrap until after the year 2116.[4]

5.3 Time Zones and the International Date Line

Figure 9 shows the relationship between the time of day at the prime meridian at Greenwich, England, and the time of day at the International Date Line, San Diego, and New York.

NOTE: As the Earth rotates (counterclockwise, since the view is looking downward from above the North Pole), the portion of the Earth that is Monday is extended. The rightmost portion of the Earth (that part that is farthest from the Sun) is always a day boundary (except at midnight), as is the International Date Line.

[4] Prior to that time, all initial Decoders will surely be out of service, and new ones can be designed to handle the wrap condition.

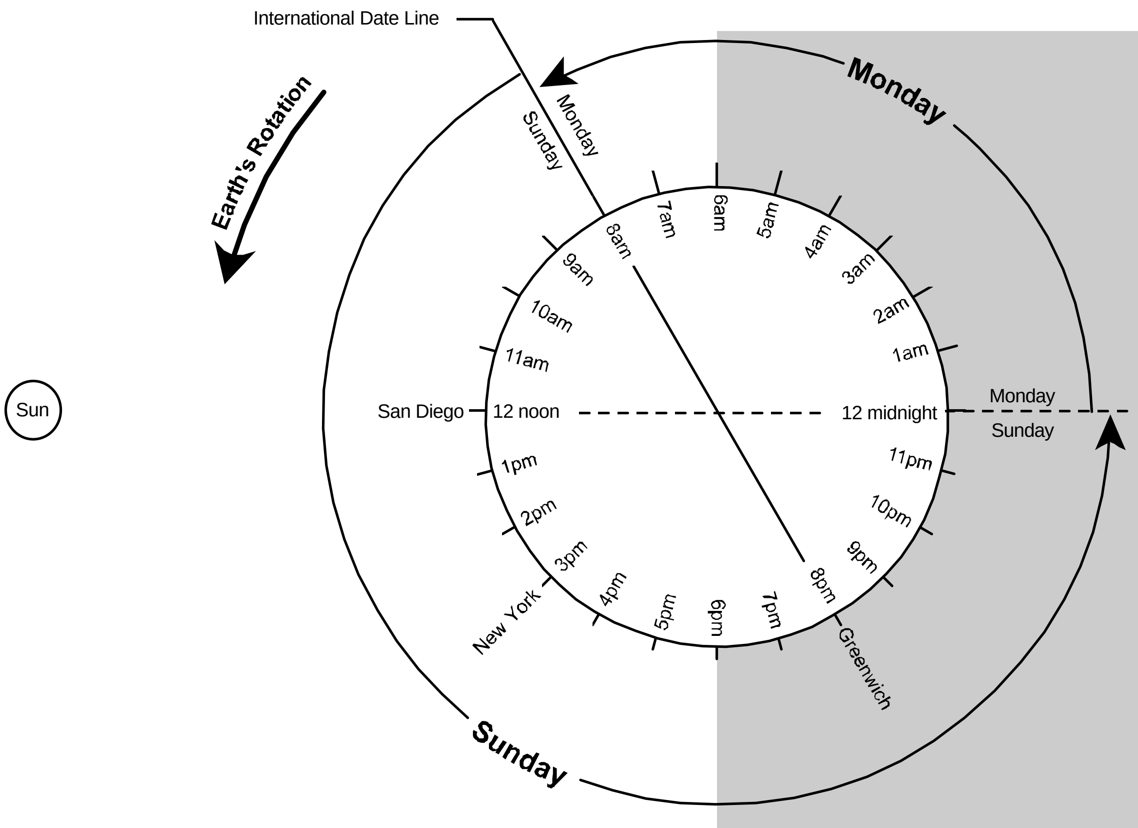

Figure 9. Time zones and the International Date Line.