

ETSI TS 102 809 v1.1.1 (2010-01)

Technická špecifikácia

**Digitálne televízne vysielanie (DVB);
Signalizácia a prenos interaktívnych aplikácií a služieb
v hybridnom vysielaní/širokopásmových prostrediach**

Digital Video Broadcasting (DVB);
Signalling and carriage of interactive applications and services
in Hybrid broadcast/broadband environments



Európsky inštitút pre telekomunikačné normy
European Telecommunications Standards Institute

Dôležité upozornenie pre používateľov tejto slovenskej verzie

ETSI je vlastníkom autorských práv tohto dokumentu ETSI.

V prípade nezrovnalosti medzi anglickou a slovenskou verziou platí anglická verzia tohto dokumentu ETSI.

ETSI neskontroloval preklad a nepreberá žiadnu zodpovednosť za presnosť prekladu tohto dokumentu ETSI.

Anglická verzia tohto dokumentu ETSI sa môže stiahnuť zo stránky:

<http://www.etsi.org/standards-search>

Referenčné číslo

DTS/JTC-DVB-270

Kľúčové slová

broadcasting, DVB, internet, IP, multimedia

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex – France

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

Siret N° 348 623 562 00017 - NAF 742 C
Neziskové združenie registrované
na podprefektúre de Grasse (06) N° 7803/88

Dôležité upozornenie

Jednotlivé kópie tohto dokumentu možno stiahnuť z

<http://pda.etsi.org>

Tento dokument môže byť dostupný vo viacerých elektronických verziách alebo v tlačenej forme. V prípade existujúceho alebo viditeľného rozdielu v obsahu medzi takýmito verziami je referenčnou verziou verzia v prenosnom dokumentovom formáte (Portable Document Format – PDF).

V prípade sporu je referenčným výtlačok vytlačenej na tlačiarňi ETSI z verzie PDF uchovávanej na určenom sieťovom serveri sekretariátu ETSI.

Používatelia tohto dokumentu by mali brať do úvahy, že dokument môže byť revidovaný alebo sa môže zmeniť jeho postavenie. Informácie o postavení tohto dokumentu a ďalších dokumentov ETSI sú dostupné na

<http://portal.etsi.org/tb/status/status.asp>

Ak nájdete v tomto dokumente chyby, svoje pripomienky zašlite na

http://portal.etsi.org/chaicor/ETSI_support.asp

Oznam o autorských právach

Nijaká časť sa nesmie reprodukovat' bez písomného povolenia.
Autorské práva a z toho vyplývajúce obmedzenia sa vzťahujú na reprodukovanie všetkými druhmi médií.

© Európsky inštitút pre telekomunikačné normy 2010.
© Európska vysielacia únia 2010.
Všetky práva vyhradené.

DECT™, PLUGTESTS™, UMTS™, TIPHON™ sú obchodné značky ETSI registrované na prospech jej členov.

3GPP™ je obchodná značka ETSI registrovaná na prospech jej členov a partnerských organizácií 3GPP.

LTE™ je obchodná značka ETSI v súčasnosti registrovaná na prospech jej členov a partnerských organizácií 3GPP.

GSM® a logo GSM sú registrované obchodné značky vo vlastníctve asociácie GSM.

Obsah

Obsah	3
Práva duševného vlastníctva	7
Predhovor	7
Národný predhovor	7
1 Predmet	8
2 Referenčné dokumenty	9
2.1 Normatívne referenčné dokumenty	9
2.2 Informatívne referenčné dokumenty	10
3 Definície a skratky	11
3.1 Definície	11
3.2 Skratky	12
4 Modely aplikácií	14
4.1 Úvod	14
4.2 Spustenie a zastavenie aplikácií	14
4.2.1 Aplikácie súvisiace s vysielacími službami	14
4.2.2 Aplikácie súvisiace s položkou obsahu na požiadanie	15
4.2.3 Aplikácie súvisiace s prevádzkovateľom siete	15
5 Signalizácia interaktívnych aplikácií a služieb	16
5.1 Sémantika	16
5.2 Metadáta aplikácie	16
5.2.1 Úvod	16
5.2.2 Typy aplikácií	17
5.2.2.1 Sémantika	17
5.2.2.2 Kódovanie MPEG-2	17
5.2.2.3 Kódovanie XML	17
5.2.3 Identifikácia aplikácie	17
5.2.3.1 Sémantika	17
5.2.3.2 Kódovanie MPEG-2	18
5.2.3.3 Kódovanie XML	18
5.2.4 Riadiace kódy aplikácie	18
5.2.4.1 Sémantika	18
5.2.4.2 Kódovanie MPEG-2	19
5.2.4.3 Kódovanie XML	19
5.2.5 Profily platformy	19
5.2.5.1 Sémantika	19
5.2.5.2 Kódovanie MPEG-2	20
5.2.5.3 Kódovanie XML	20
5.2.6 Viditeľnosť aplikácie	21
5.2.6.1 Sémantika	21
5.2.6.2 Kódovanie MPEG-2	21
5.2.6.3 Kódovanie XML	21
5.2.7 Priorita aplikácie	21
5.2.7.1 Sémantika	21
5.2.7.2 Kódovanie MPEG-2	22
5.2.7.3 Kódovanie XML	22
5.2.8 Ikony aplikácie	22
5.2.8.1 Sémantika	22
5.2.8.2 Kódovanie MPEG-2	22
5.2.8.3 Kódovanie XML	23
5.2.9 Obmedzenia grafiky	24
5.2.9.1 Sémantika	24
5.2.9.2 Kódovanie MPEG-2	25
5.2.9.3 Kódovanie XML	25
5.2.10 Použitie aplikácie	25
5.2.10.1 Sémantika	25
5.2.10.2 Kódovanie MPEG-2	25

5.2.10.3	Kódovanie XML.....	26
5.2.11	Uložené aplikácie.....	26
5.2.11.1	Sémantika.....	26
5.2.11.2	Kódovanie MPEG-2.....	28
5.2.11.3	Kódovanie XML.....	29
5.2.12	Súbor opisu aplikácie.....	29
5.2.12.1	Opis.....	29
5.2.12.2	Názov súboru opisu aplikácie a jeho umiestnenie.....	29
5.2.12.3	Syntax.....	30
5.2.12.4	Sémantika.....	30
5.3	Tabuľka MPEG-2 a syntax sekcie.....	31
5.3.1	Prehľad.....	31
5.3.1.1	Prehľad spoločnej signalizácie.....	31
5.3.1.2	Prehľad doplnkovej signalizácie na aplikácie prenášané prostredníctvom OC.....	31
5.3.1.3	Ako pridať nový návrh (informatívne).....	31
5.3.2	Špecifická informácia o programe.....	32
5.3.2.1	Tok signalizácie aplikácie.....	32
5.3.2.2	Toky vysielaných dát.....	32
5.3.3	Poznámky.....	33
5.3.3.1	Rezervované.....	33
5.3.3.2	Rezervované na budúce používanie.....	33
5.3.4	Tabuľka informácií o aplikácii.....	33
5.3.4.1	Chybné dáta.....	33
5.3.4.2	Prenos AIT a monitorovanie.....	34
5.3.4.3	Optimalizovaná signalizácia AIT.....	34
5.3.4.4	Viditeľnosť AIT.....	34
5.3.4.5	Definícia subtabuľky do AIT.....	34
5.3.4.6	Syntax AIT.....	34
5.3.4.7	Používanie neverejných deskriptorov v AIT.....	36
5.3.4.8	Kódovanie textu v AIT.....	36
5.3.4.9	Prístup na AIT vo formáte MPEG-2 prostredníctvom širokopásmového pripojenia.....	37
5.3.5	Generické deskriptory.....	37
5.3.5.1	Deskriptor signalizácie aplikácie.....	37
5.3.5.2	Deskriptor identifikátora id vysielania dát.....	38
5.3.5.3	Deskriptor aplikácie.....	40
5.3.5.4	Deskriptor záznamu aplikácie.....	41
5.3.5.5	Deskriptor používania aplikácie.....	43
5.3.5.6	Deskriptory informácií pre používateľa.....	43
5.3.5.7	Deskriptor autorizácie externej aplikácie.....	45
5.3.5.8	Deskriptor obmedzenia grafiky.....	46
5.3.6	Deskriptory prenosového protokolu.....	47
5.3.6.1	Syntax selektora bajtov na prenos OC.....	48
5.3.6.2	Syntax selektora bajtov na prenos interaktívnym kanálom.....	49
5.3.7	Deskriptor umiestnenia jednoduchkej aplikácie.....	50
5.3.7.1	Príklad.....	50
5.3.8	Deskriptor ohraničenia jednoduchkej aplikácie.....	51
5.3.9	Informácie o službe.....	51
5.3.9.1	Deskriptor vysielania dát na oznámenie interaktívnej aplikácie.....	51
5.3.10	Uložené aplikácie.....	54
5.3.10.1	Deskriptor uloženia aplikácie.....	54
5.4	Syntax založená na XML.....	54
5.4.1	Signalizácia aplikácie súvisiacej so službou.....	55
5.4.2	Signalizácia nesúvisiacich aplikácií.....	55
5.4.3	Rozšírenia definované prvkami SD&S.....	55
5.4.3.1	Balíček.....	55
5.4.3.2	Služba IP.....	56
5.4.3.3	Prevádzkovateľ služby.....	56
5.4.4	Definície nových prvkov XML.....	56
5.4.4.1	Zoznam aplikácie.....	56
5.4.4.2	Aplikácia.....	57
5.4.4.3	Identifikátor aplikácie.....	57
5.4.4.4	Deskriptor aplikácie.....	58

5.4.4.5	Deskriptor viditeľnosti.....	58
5.4.4.6	Deskriptor ikony.....	58
5.4.4.7	Pomer strán.....	59
5.4.4.8	Verzia Mhp.....	59
5.4.4.9	Kapacita pamäťi.....	59
5.4.4.10	Typ pamäte.....	59
5.4.4.11	Typ aplikácie.....	60
5.4.4.12	Typ aplikácie DVB.....	60
5.4.4.13	Riadiaci kód aplikácie.....	60
5.4.4.14	Deskriptor špecifickej aplikácie.....	60
5.4.4.15	Abstraktná IP služba.....	60
5.4.4.16	Typ ponúkanej služby.....	61
5.4.4.17	Zistenie aplikácie.....	61
5.4.4.18	Deskriptor používania aplikácie.....	61
5.4.4.19	Typ deskriptora prenosového protokolu.....	62
5.4.4.20	Typ prenosu HTTP.....	62
5.4.4.21	Typ prenosu OC.....	62
5.4.4.22	Príznak typu zložky.....	62
5.4.4.23	Typ deskriptora umiestnenia jednoduchej aplikácie.....	63
5.4.4.24	Typ deskriptora ohraničenia jednoduchej aplikácie.....	63
5.4.5	Záznam sprístupnenia aplikácie.....	63
5.5	Konštantné hodnoty.....	63
6	Odkazovanie na služby DVB.....	65
6.1	Syntax a sémantika URL DVB.....	65
	Syntax a sémantika schémy URL "dvb:" sú definované v [20].	65
6.2	Rozlíšenie URL DVB.....	65
6.2.1	Deskriptor identifikátora služby.....	65
7	Prenos aplikácie.....	67
7.1	Objektový karusel.....	67
7.2	HTTP.....	67
8	Synchronizácia.....	68
8.1	Úvod.....	68
8.2	Odkazovanie.....	68
A.1	Úvod.....	69
A.2	Prvky definované špecifikáciou platformy.....	69
B.1.1	Kľúč k poznámkam.....	70
B.2	Profil objektového karuselu.....	70
B.2.1	Sekcie DSM-CC.....	70
B.2.1.1	Sekcie v rámci paketu TS.....	71
B.2.2	Dátový karusel.....	71
B.2.2.1	Všeobecne.....	71
B.2.2.2	Zobrazenie informácií o sťahovaní.....	71
B.2.2.3	Inicializácia servera sťahovania.....	71
B.2.2.4	Modul informácií.....	72
B.2.2.4.1	Deskriptor návštevy.....	73
B.2.2.4.2	Deskriptor priority kešovania.....	73
B.2.2.5	Informácie o službe sieťového prechodu.....	74
B.2.2.6	Zrušenie sťahovania.....	75
B.2.2.7	Blok sťahovania dát.....	75
B.2.3	Objektový karusel.....	75
B.2.3.1	Správa generického objektu BIOP.....	75
B.2.3.2	Reťazce CORBA.....	76
B.2.3.3	Súbor správy BIOP.....	76
B.2.3.4	Deskriptor typu obsahu.....	77
B.2.3.5	Adresár správy BIOP.....	78
B.2.3.6	Správa služby sieťového prechodu.....	80
B.2.3.7	Interoperabilné odkazy na objekt BIOP.....	80
B.2.3.8	Tok správy BIOP.....	85
B.2.3.9	Tok správy o udalosti BIOP.....	87
B.2.3.10	Ďalšie hodnoty TAPUse.....	89

B.2.4	Vysielanie časových základní a udalostí.....	89
B.2.4.1	Tok a správy toku udalostí	90
B.2.4.2	Deskriptory toku.....	91
B.2.4.3	Sekcie DSM-CC nesúce deskriptory toku	92
B.2.4.4	Vysielanie časových základní	92
B.2.4.5	Vysielanie udalostí.....	93
B.2.4.6	Monitorovanie vysielania časových základní a udalostí	94
B.2.5	Priradovanie a používanie hodnôt transactionId	96
B.2.5.1	Informatívny základ	96
B.2.5.2	Sémantika poľa transactionId DVB	97
B.2.6	Mapovanie objektov do modulov dátového karuselu	97
B.2.7	Kompresia modulov	98
B.2.8	Zostavenie objektového karuselu	98
B.2.8.1	Deskriptor identifikátora karuselu.....	99
B.2.9	Nedostupnosť karuselu.....	100
B.2.10	Poskytovanie viacerých služieb karuselmi	101
B.3	Mapovanie príznaku priradenia	102
B.3.1	Algoritmus rozhodovania na mapovanie príznaku priradenia.....	102
B.3.1.1	TAPUse nie je BIOP_PROGRAM_USE	102
B.3.1.2	TAPUse je BIOP_PROGRAM_USE	103
B.3.2	Príznačky priradenia.....	103
B.3.3	Deskriptor prerušenia príznakov priradenia	103
B.4	Príklad objektového karuselu (informatívne).....	103
B.5	Kešovanie.....	105
B.5.1	Určenie verzie súboru.....	105
B.5.2	Úrovne transparentnosti kešovania	105
B.5.2.1	Transparentné kešovanie	105
B.5.2.2	Semitransparentné kešovanie	106
B.5.2.3	Statické kešovanie.....	107
B.5.3	Dynamická štruktúra karuselu	107
	Príloha C (normatívna): Všeobecné aplikovanie západoeurópskeho súboru znakov	109
	Príloha D (informatívna): Literatúra	110
	Príloha NA (informatívna): Doplnujúce skratky	110
	História	111

Práva duševného vlastníctva

Práva duševného vlastníctva, ktoré majú alebo môžu mať zásadný význam pre tento dokument, mohli byť oznámené organizácii ETSI. Informácie o týchto zásadných právach duševného vlastníctva, ak existujú, sú pre členov i nečlenov ETSI verejne dostupné a môžu ich nájsť v dokumente ETSI SR 000 314 s názvom Práva duševného vlastníctva (IPR), ktorý možno získať na sekretariáte ETSI. Najnovšie znenie je dostupné na serveri ETSI (<http://webapp.etsi.org/IPR/home.asp>).

V súlade so svojou politikou v oblasti práv duševného vlastníctva ETSI nevyhľadáva ani neskúma nijaké práva duševného vlastníctva. Neposkytuje ani záruku týkajúcu sa existencie iných IPR, ktoré nie sú uvedené v dokumente ETSI SR 000 314 (alebo v jeho aktualizovaných vydaniach na serveri ETSI), ktoré majú, môžu mať alebo môžu nadobudnúť zásadný význam pre predkladaný dokument.

Predhovor

Technickú špecifikáciu (TS) vytvorila spojená technická komisia (JTC) Vysielanie Európskej vysielacej únie (EBU), Európskeho výboru pre normalizáciu v elektrotechnike (CENELEC) a Európskeho inštitútu pre telekomunikačné normy (ETSI).

POZNÁMKA. – Spojená technická komisia EBU/ETSI Vysielanie sa zriadila v roku 1990 s cieľom koordinovať návrhy noriem na poli vysielania a v pridružených oblastiach. Od roku 1995 sa JTC Vysielanie zaradením CENELEC, zodpovedného za normalizáciu rozhlasových a televíznych prijímačov, do Memoranda o porozumení stala tripartitným orgánom. EBU je profesionálnym združením vysielateľov, ktorého práca zahŕňa koordináciu aktivít svojich členov v technickej a právnej oblasti a vo výrobe a výmene programov programov. EBU má aktívnych členov z približne 60 krajín európskeho vysielacieho priestoru; svoje sídlo má v Ženeve.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

Digital Video Broadcasting Project (DVB) je dôležitým priemyselným konzorciom vysielateľov, výrobcov, prevádzkovateľov sietí, vývojárov softvéru, regulačných orgánov, vlastníkov obsahu a ďalších zainteresovaných návrhárov globálnych noriem na poskytovanie digitálnej televízie a dátových služieb. DVB podporuje trhovo riadené riešenia, ktoré spĺňajú potreby a ekonomické možnosti investorov a spotrebiteľov vysielacieho priemyslu. Štandardy DVB pokrývajú všetky oblasti digitálnej televízie od prenosu cez rozhrania, podmienený prístup a interaktivitu digitálneho videa, audia a dát. Konzorcium sa zišlo v roku 1993 na prípravu globálnej štandardizácie, interoperability a budúcich testovacích špecifikácií.

Národný predhovor

Technická špecifikácia (TS) obsahuje Národnú prílohu NA (informatívnu) a Národnú poznámku v článku 3.2 Skratky.

1 Predmet

Dokument definuje rámec na signalizáciu a prenos interaktívnych aplikácií alebo služieb vysielaním a širokopásmovými sieťami, ktorý zahŕňa:

- signalizáciu interaktívnych aplikácií alebo služieb v tradičných vysielacích sieťach a širokopásmových sieťach;
- distribúciu súborov interaktívnych aplikácií alebo služieb cez klasické vysielacie siete alebo širokopásmové siete;
- synchronizáciu interaktívnych aplikácií a služieb s videom alebo obsahom audia distribuovaným cez klasické vysielacie siete alebo širokopásmové siete;
- odkazovanie na obsah videa audia alebo podtitulky distribuované cez klasické vysielacie siete alebo širokopásmové siete z interaktívnych aplikácií alebo služieb.

Dokument je nezávislý od ktorejkoľvek konkrétnej technológie na interaktívne aplikácie alebo služby. Organizácie, ktoré definujú využívanie interaktívnych aplikácií a služieb, sa môžu na tento dokument odvolávať ako na samostatný celok so všetkými príslušnými právami. Očakáva sa, že organizácie si z nich vyberú funkcionality vhodnú do ich trhu alebo na využitie, ktoré je definované v tomto dokumente. Výrazy "musí", "mal by" a im podobné sa používajú len v prípade, keď sa použije konkrétna funkcia a neznamená to, že táto funkcia sama osebe je povinná.

2 Referenčné dokumenty

Odkazy sú špecifikované (určené dátumom uverejnenia a/alebo číslom vydania, alebo číslom verzie) alebo nešpecifikované:

- na konkrétny odkaz sa následné revízie nevzťahujú;
- nešpecifický odkaz sa vzťahuje iba na celý dokument alebo jeho časť, a iba v týchto prípadoch:
- v prípade, že bude možné využiť všetky budúce zmeny odkazovaného dokumentu na účely odkazujúceho dokumentu;
- na informatívne odkazy.

Referenčné dokumenty, ktoré nie sú verejne dostupné na uvedenom mieste, možno nájsť na adrese <http://docbox.etsi.org/Reference>.

POZNÁMKA. – Aj keď v čase publikovania tohto dokumentu platili všetky hyperlinky uvedené v článku, ETSI nemôže zaručiť ich dlhodobú platnosť.

2.1 Normatívne referenčné dokumenty

Odkazované dokumenty sú nevyhnutné na aplikovanie dokumentu. Pri špecifikovaných odkazoch platí len citovaná verzia. Pri nešpecifikovaných odkazoch platí najnovšia verzia odkazovaného dokumentu (vrátane všetkých jeho doplnkov).

- [1] ETSI EN 300 468 (V1.9.1): "Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems".
- [2] ETSI EN 301 192 (V1.3.1): "Digital Video Broadcasting (DVB); DVB specification for data broadcasting".
- [3] ISO/IEC 13818-1: "Information technology - generic coding of moving pictures and associated audio information: Systems".
- [4] ISO/IEC 13818-6: "Information technology - generic coding of moving pictures and associated audio information: Part 6: Extensions for DSM-CC".
- [5] IETF RFC 2616: "Hypertext Transfer Protocol - HTTP/1.1".
- [6] ETSI TS 102 034: "Digital Video Broadcasting (DVB); Transport of MPEG-2 TS Based DVB Services over IP Based Networks".
- [7] ISO 639-2:1998: "Codes for the representation of names of languages - part 2: Alpha-3 code".
- [8] ISO/IEC 8859-1: "Information technology - 8-bit single-byte coded graphic character sets: Part 1: Latin alphabet No. 1".
- [9] Object Management Group: "The Common Object Request broker: Architecture and Specification".

POZNÁMKA. – Dostupný na <http://www.omg.org/cgi-bin/doc?formal/97-09-01.pdf>.

- [10] IETF RFC 2045: "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies".

- [11] IETF RFC 1950: "ZLIB Compressed Data Format Specification version 3.3".
- [12] IETF RFC 1951: "DEFLATE Compressed Data Format Specification version 1.3".
- [13] ETSI TS 102 823 (V1.1.1): "Digital Video Broadcasting (DVB); Specification for the carriage of synchronized auxiliary data in DVB transport streams".
- [14] IETF RFC 3986: "Uniform Resource Identifier (URI): Generic Syntax".
- [15] DVB Identifiers.
POZNÁMKA. – Dostupný na <http://www.dvb.org>.
- [16] ATSC A/100-5: "DTV application software environment level 1 (DASE-1); Part 5: ZIP archive resource format".
- [17] W3C: "XML Schema Part 0: Primer Second Edition".
POZNÁMKA. – Dostupný na <http://www.w3.org/TR/xmlschema-0/>
- [18] IETF RFC 1945: "Hypertext Transfer Protocol - HTTP/1.0".
- [19] IETF RFC 2818: "HTTP over TLS".
- [20] ETSI TS 102 851: "Digital Video Broadcasting (DVB); Uniform Resource Identifiers (URI) for DVB Systems".
POZNÁMKA. – Dokument DVB TM 3750R4.
- [21] IETF RFC 1035: "Domain names - implementation and specification".

2.2 Informatívne referenčné dokumenty

Odkazované dokumenty nie sú nevyhnutné na aplikovanie dokumentu, ale môžu byť nápomocné používateľovi s ohľadom na predmetnú oblasť. Pri nešpecifikovaných odkazoch platí najnovšia verzia odkazovaného dokumentu (vrátane všetkých jeho doplnkov).

- [i.1] ETSI TS 102 727: "Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.2.2".
- [i.2] ETSI TR 101 202 (V1.1.1): "Digital Video Broadcasting (DVB); Implementation guidelines for Data Broadcasting".

3 Definície a skratky

3.1 Definície

V dokumente sa používajú názvy a definície:

abstraktná služba (angl. **abstract service**): spôsob zoskupenia niekoľkých súvisiacich aplikácií, kde je určitý agregátor zodpovedný za to, že táto množina súvisiacich aplikácií spolupracuje

POZNÁMKA. – Pojem zovšeobecňuje vysielaciu službu na podporu aplikácií, ktoré nesúvisia so žiadnym televíznym vysielaním alebo rozhlasovou službou. Skupina trvalo prístupných (rezidentných) aplikácií, ktoré spoločne zoskupil prevádzkovateľ siete (napríklad čet, e-mail, prehliadač WWW), vytvára jednu abstraktnú službu.

aplikácia (angl. **application**): súbor prostriedkov a logiky, ktoré spoločne poskytujú ucelené interaktívne služby pre používateľov

životný cyklus aplikácie (angl. **application lifecycle**): rôzne stavy, v ktorých môže aplikácia existovať, a prechody medzi nimi, vrátane spustenia a zastavenia

aplikačné programovacie rozhranie [(angl. **application Programming Interface (API)**): rozhranie medzi aplikáciou a špecifickým prvkom, funkcionalitou alebo prostriedkom koncového zariadenia

tradičná vysielacia sieť (angl. **classical broadcast network**): sieť pozostávajúca z klasických vysielacích technológií, ktoré používajú prenosové toky MPEG-2 prenášané fyzickou vrstvou, ako je DVB-T, DVB-S alebo DVB-C

špecifikácia platformy (angl. **platform specification**): dokument, ktorý odkazuje na predkladaný dokument a definuje, ktoré časti dokumentu sa použijú na konkrétnom trhu alebo na rozvoj

vybraná služba (angl. **selected service**): služba televízneho alebo rozhlasového vysielania, ktorá je v danom čase reprodukováaná prijímačom a jej signalizácia je monitorovaná prijímačom

služba (angl. **service**): postupnosť programov riadená vysielateľom, ktorá sa môže vysielat' ako časť časového plánu

nesúvisiaca aplikácia (angl. **unbound application**): aplikácia, ktorá nie je pridružená k vysielacej službe

3.2 Skratky

V dokumente sa používajú skratky:

ADF	Application Description File	súbor opisu aplikácie
AIT	Application Information Table	tabuľka informácií o aplikáciách
API	Application Programming Interface	aplikačné programovacie rozhranie
AV	Audio/Video	audio/video
BNF	Backus-Naur Form	Backusova-Naurova forma
CRC	Cyclic Redundancy Check	kontrola cyklickým redundantným kódom
DAVIC	Digital Audio Video Council	Výbor pre digitálne audio – video
DSI	Download Server Initiate	inicializácia servera sťahovania
DSM-CC	Digital Storage Media Command and Control	príkazy na uloženie digitálneho pamäťového média a riadenie
DSM-CC-OC	Digital Storage Media Command and Control Object Carousel	príkazy na uloženie digitálneho pamäťového média a riadenie objektového karusela
DSM-CC-UU	Digital Storage Media Command and Control User to User	príkazy na uloženie digitálneho pamäťového média a riadenie spojenia účastník – účastník
DTD	Document Type Definition	definícia typu dokumentov
DVB	Digital Video Broadcasting	digitálne televízne vysielanie
ECMA	European Computer Manufacturers Association	Združenie európskych výrobcov počítačov
EPG	Electronic Programme Guide	elektronický sprievodca programom
ETSI	European Telecommunications Standards Institute	Európsky inštitút pre telekomunikačné normy
HTML	Hyper Text Mark-up Language	hypertextový značkový jazyk
HTTP	Hyper Text Transport Protocol	hypertextový prenosový protokol
IEC	International Electrotechnical Commission	Medzinárodná elektrotechnická komisia
IOR	Interoperable Object References	referencie prevádzkyschopných objektov
IP	Internet Protocol	internetový protokol
IPR	Intellectual Property Rights	práva duševného vlastníctva
IPTV	IP TeleVision	televízia s internetovým protokolom
ISO	International Organization for Standardization	Medzinárodná organizácia pre normalizáciu
ITU	International Telecommunication Union	Medzinárodná telekomunikačná únia
MHP	Multimedia Home Platform	domáca multimediálna platforma
MPEG	Moving Picture Expert Group	skupina expertov na pohyblivé obrazy

NIT	Network Information Table	tabuľka informácií o sieti
OC	Object Carousel	objektový karusel
PMT	Program Map Table	tabuľka programovej mapy
PNG	Portable Network Graphics	prenosná sieťová grafika
PSI	Program Specific Information	špecifická informácia o programe
SD&S	Service Discovery and Selection	zistenie služby a výber
SDT	Service Description Table	tabuľka opisu služby
SI	Service Information	informácia o službe
TCP	Transmission Control Protocol	protokol riadenia prenosu
TS	Transport Stream	transportný tok/prenosový tok
TV	TeleVision	televízia
UI	User Interface	používateľské rozhranie
URL	Uniform Resource Locator	jednotný vyhľadávač prostriedka
UTF	Unicode Transformation Format	transformačný formát s jednoznačným kódom
UU	User to User	používateľ – používateľ
WWW	World Wide Web	celosvetová internetová sieť
XML	eXtensible Markup Language	rozšíriteľný značkovací jazyk
POZNÁMKA. – V kapitole Príloha NA sú uvedené doplnujúce skratky: TAP, BIOP, ORB (platí v SR).		

4 Modely aplikácií

4.1 Úvod

Dokument umožňuje použiť široký výber rôznych aplikačných modelov v závislosti od toho, ktoré voliteľné funkcie sú vybrané. Niektoré z nich sú uvedené a zoradené od najjednoduchších k zložitejším:

- aplikácie súvisiace práve s jednou vysielacou službou, ktoré sú spustené výberom tejto služby a zastavené zrušením tohto výberu;
- aplikácie súvisiace viac ako s jednou vysielacou službou, ktoré sú spustené výberom ktorejkoľvek z týchto služieb, zastavené zrušením tohto výberu, aj keď zrušenie výberu je súčasťou prechodu na novú službu, ktorá s aplikáciou tiež súvisí;
- aplikácie, ktoré sú stále prístupné počas zmien služieb, sú aplikácie súvisiace viac ako s jednou vysielacou službou a môžu sa spustiť ktoroukoľvek vybranou službou, s ktorou súvisia, bežia bez prerušenia, zatiaľ čo ostatné súvisiace služby zostanú vybrané a skončia, keď už ďalej nesúvisia so žiadnou aktuálne vybranou službou;
- aplikácie súvisiace s obsahom, ktorý je časťou vysielacej služby, napríklad samostatný program alebo reklamy, sa spustia so spustením tohto obsahu, ak je v tom čase vybraná služba, a skončia, keď sa skončí obsah, ak v tom čase trvá vybraná služba;
- aplikácie súvisiace s obsahom na požiadanie sa spracujú zhodne s aplikáciami súvisiacimi s časťami vysielacej služby, vrátane možností dynamických zmien pri sťahovaní obsahu na požiadanie, alebo budú prístupné počas celého trvania obsahu;
- aplikácie, ktoré sú prístupné počas pripojenia koncového zariadenia na sieť prevádzkovateľa alebo platformy služby poskytovateľa; v niektorých využitíach sa môžu trvale sprístupniť.

4.2 Spustenie a zastavenie aplikácií

4.2.1 Aplikácie súvisiace s vysielacími službami

Ak je vybraná programová služba, platí nasledujúce:

- koncové zariadenie musí určiť, či existujú nejaké aplikácie signalizované ako súvisiace so službou podľa ustanovení článkov 5.3 alebo 5.4 tohto dokumentu;
- aplikácie, ktoré sú časťou tejto služby, sú signalizované riadiacim kódom AUTOŠTART (pozri tabuľku 3) a ešte nie sú spustené predtým zobrazenou službou, musia sa spustiť;
- aplikácie, ktoré sú časťou tejto služby, sú signalizované riadiacim kódom AUTOŠTART a sú už spustené predtým zobrazenou vysielacou službou, pôjdu aj naďalej bez prerušenia; aplikácia sa nesmie spustiť druhýkrát;
- aplikácie, ktoré sú časťou tejto vysielacej služby, sú signalizované riadiacim kódom PRESENT, musia byť naďalej prístupné, ak sú už spustené, ale nesmú sa spustiť, pokiaľ ešte neboli spustené;
- aplikácie spustené niektorou predtým zobrazenou vysielacou službou, ktoré nie sú súčasťou nového programu, sa musia prerušiť ako súčasť zmeny zobrazovanej služby.

Pokiaľ je vysielacia služba naďalej spustená, ako už bolo uvedené, platí nasledujúce:

- aplikácie pridané k službe s riadiacim kódom AUTOŠTART sa musia automaticky spustiť, keď ich pridanie zistilo koncové zariadenie; aplikácie pridané k službe s iným riadiacim kódom sa nesmú automaticky spustiť;
- aplikácie, ktoré sú časťou služby, ktorej sa zmenil riadiaci kód na AUTOŠTART z akejkoľvek inej hodnoty, musia sa automaticky spustiť, pokiaľ už neboli spustené;
- aplikácie, ktoré sú časťou služby, ktorej sa zmenil riadiaci kód z AUTOŠTART na PRESENT, a ktoré sú stále prístupné, musia byť prístupné naďalej;
- aplikácie, ktorým sa zmenil riadiaci kód na KILL alebo DESTROY, sa musia zastaviť podľa sémantiky, ktorá definuje špecifikáciu spôsobu aplikovania týchto riadiacich kódov.

Keď sa zastaví výber vysielacej služby, platí nasledujúce:

- aplikácie, ktoré signalizujú, že prvok serviceBound v deskriptore application_descriptor (pozri články 5.3.5.3 a 5.4.4.4) má správnu hodnotu, sa musia zastaviť;
- keď je aplikácia stále prístupná aj po zmene vysielacej služby, musí sa spustiť ako signalizovaná nová služba a nie predchádzajúca služba.

4.2.2 Aplikácie súvisiace s položkou obsahu na požiadanie

Aplikácie súvisiace s obsahom na požiadanie sa spracujú zhodne s aplikáciami súvisiacimi s časťou vysielacej služby, vrátane možnosti dynamických zmien počas obsahu na požiadanie, alebo budú prístupné počas celého času trvania obsahu. V druhom prípade aplikácie, ktorých riadiaci kód je AUTOŠTART, sa musia spustiť so začatím zobrazovania obsahu a musia sa zastaviť s ukončením zobrazovania obsahu. Riadiace kódy aplikácií nie je možné meniť.

4.2.3 Aplikácie súvisiace s prevádzkovateľom siete

Aplikácie viazané na prevádzkovateľa siete sa môžu spustiť kedykoľvek, keď je terminál pripojený k sieti tohto prevádzkovateľa. Aplikácie s riadiacim kódom AUTOŠTART sa musia spúšťať automaticky, keď sa koncové zariadenie pripojí prvýkrát na túto sieť, alebo keď je(sú) aplikácia(e) pridaná(é) do signalizácie. Keď koncové zariadenia môžu zmeniť prevádzkovateľa siete, tak v rámci tohto postupu sa musia ukončiť aplikácie predchádzajúceho prevádzkovateľa siete a spustia sa aplikácie nového prevádzkovateľa siete.

5 Signalizácia interaktívnych aplikácií a služieb

5.1 Sémantika

Toto ustanovenie sa vzťahuje na tieto témy:

- ako prijímač identifikuje aplikácie priradené službe a hľadá umiestnenia, z ktorých ich chce obnoviť;
- signalizácia, ktorá umožňuje vysielanie na riadenie životného cyklu aplikácií;
- ako prijímač identifikuje prostriedky vysielaných dát, ktoré sú požadované aplikáciami služby.

Väčšina signalizácie je všeobecne použiteľná. Napríklad deskriptor `application_descriptor` je nezávislý od vysielanej aplikácie. Ďalšia signalizácia sa môže definovať špecifikáciou platformy. Identifikátory URI používané v tejto signalizácii musia zodpovedať formátu a obmedzeniam definovaným v [20].

5.2 Metadáta aplikácie

5.2.1 Úvod

Aplikácia môže mať týchto niekoľko priradených položiek metadát:

- typ: identifikuje platformu potrebnú na vykonávanie alebo prezentáciu aplikácie;
- identifikátor: identifikuje aplikáciu;
- riadiaci kód: definuje stav životného cyklu aplikácie;
- profil: definuje minimálny profil koncového zariadenia potrebný na túto aplikáciu;
- viditeľnosť: označuje, či je aplikácia viditeľná pre používateľa, alebo na iné aplikácie v zozname aplikácií API (pokiaľ je podporované také API);
- priorita: definuje prednosť aplikácie proti iným signalizovaným aplikáciám;
- ikony: identifikuje umiestnenie ikon na túto aplikáciu;
- obmedzenie grafiky: identifikuje akékoľvek obmedzenia tejto aplikácie s ohľadom na zmeny v konfigurácii grafiky alebo prezentovanom videu;
- informácia o uložení: definuje, či sa má aplikácia uložiť, a ktoré súbory aplikácie sa majú uložiť.

V tomto článku každá časť definuje najprv sémantiku danej položky informácie, potom položky kódovania, založeného na MPEG-2, a položkou XML.

5.2.2 Typy aplikácií

5.2.2.1 Sémantika

Všetky aplikácie sa majú pridružiť, aby koncové zariadenie mohlo vyradiť tie typy aplikácií, ktoré nepodporuje.

POZNÁMKA. – Typ aplikácie nie je dostatočnou zárukou, že koncové zariadenie môže spustiť aplikáciu (pozri metadáta týkajúce sa profilov aplikácií v článku 5.2.5).

5.2.2.2 Kódovanie MPEG-2

Kódovanie typu aplikácií v MPEG-2 je v tvare pätnásťbitového čísla. Toto umožňuje prijímačom odfiltrovať signalizáciu nepodporovaných typov aplikácií. Definované typy aplikácií sú registrované v DVB [15]. Z historických dôvodov sú typy aplikácií registrované identifikátorom MHP_Application_Type_ID.

5.2.2.3 Kódovanie XML

Kódovanie typu aplikácií v XML tvorí reťazce, typicky typu MIME (pozri článok 5.4.4.11).

5.2.3 Identifikácia aplikácie

5.2.3.1 Sémantika

Každá aplikácia má pridružený identifikátor aplikácie. Ten sa skladá z dvoch častí `organisation_id` a `application_id` takto:

identifikátor organizácie (angl. **organisation_id**): pole je globálne jedinečná hodnota identifikujúca organizáciu, ktorá je zodpovedná za aplikáciu; hodnoty sú registrované v DVB [15]; hodnoty s nulami sa nesmú kódovať; na kompatibilitu s článkom B.2.10 musí byť nulových 8 najvýznamnejších bitov `organisation_id`

Aplikácie overené pomocou certifikátov X.509 majú toto pole uvedené v poli názvu subjektu `organisationName` v "liste" osvedčenia o overenej aplikácii.

POZNÁMKA. – Zahnutie tejto oblasti do listu osvedčenia poskytuje overenie hodnoty.

identifikátor aplikácie (angl. **application_id**): pole jednoznačne identifikuje aplikáciu; pole je pridelené registrovanou organizáciou s `organisation_id`, ktorá rozhoduje o postupe alokácie v rámci organizácie; hodnoty s nulami sa nesmú kódovať

Hodnoty `application_id` sú rozdelené z bezpečnostných dôvodov do troch častí: jedna na neoznačené aplikácie, druhá na označené aplikácie a tretia na špecificky prednostné aplikácie. Aplikácie prenášané ako neoznačené musia používať `application_id` z časti neoznačené aplikácie a aplikácie prenášané ako označené musia používať `application_id` z časti označené aplikácie. Aplikácie prenášané ako prednostné musia používať `application_id` z časti prednostné aplikácie.

Tabuľka 1 – Rozsah hodnôt – application_id

Hodnoty application_id	Použitie
0x0000	nesmie sa používať
0x0001 to 0x3fff	Application_ids na neoznačené aplikácie
0x4000 to 0x7fff	Application_ids na označené aplikácie
0x8000 to 0x9fff	Application_ids na prednostné aplikácie
0xa000 to 0xffff	Rezervované na budúce použitie DVB
0xfffe	špeciálne zástupné znaky na označené aplikácie organizácie
0xffff	špeciálne zástupné znaky na všetky aplikácie organizácie

Hodnoty application_id 0 x ffff a 0 x fffe sú zástupné znaky. Nesmú sa používať na identifikáciu aplikácie, ale môžu sa napríklad použiť v deskriptore external_application_authorization_descriptor (pozri článok 5.3.5.7). Hodnota 0 x ffff zodpovedá všetkým aplikáciám s rovnakou hodnotou organisation_id. Hodnota 0 x fffe zodpovedá všetkým označeným aplikáciám s rovnakou hodnotou organisation_id.

Rovnaký identifikátor aplikácie sa môže použiť na rôzne typy aplikácií, ktoré vykonávajú v podstate rovnakú funkciu.

5.2.3.2 Kódovanie MPEG-2

Toto je šesťbajtové pole s nasledujúcou štruktúrou.

Tabuľka 2 – Syntax identifikátora aplikácie

	Počet bitov
application_identifier {	
organisation_id	32
application_id	16
}	

Rovnaký identifikátor application_identifier() sa musí uviesť iba raz v rámci skupiny aplikácií signalizovaných na rovnaký typ aplikácie.

5.2.3.3 Kódovanie XML

Identifikátor aplikácie je definovaný nasledujúcim fragmentom XML, ktorého prvky sú definované skôr.

```
<xsd:complexType name="ApplicationIdentifier">
  <xsd:sequence>
    <xsd:element name="orgId" type="xsd:unsignedInt"/>
    <xsd:element name="appId" type="xsd:unsignedShort"/>
  </xsd:sequence>
</xsd:complexType>
```

5.2.4 Riadiace kódy aplikácie

Signalizácia vysielania poskytuje pre vysielateľov mechanizmus riadenia životného cyklu štandardných typov aplikácií.

5.2.4.1 Sémantika

Riadiaci kód umožňuje signalizovať prijímaču vysielateľa, čo má robiť s aplikáciou s ohľadom na jej životný cyklus. Skupiny kódov majú určité rozdiely medzi typmi aplikácií a presná sémantika je definovaná na každý typ aplikácie zvlášť.

Aplikácia sa nesmie zmeniť v prípade, ak prijímač prijme neznámy kód.

Tabuľka 3 – Hodnoty riadiaceho kódu aplikácie

Kódovanie MPEG-2	Identifikátor	Sémantika
0x00		reserved_future_use (rezervované na budúce používanie)
0x01	AUTOSTART (AUTOŠTART)	Keď je vybraná služba, aplikácia sa musí spustiť, pokiaľ nie je už aplikácia spustená.
0x02	PRESENT (PRÍTOMNÝ)	Aplikácia môže bežať počas výberu služby, nesmie sa spustiť automaticky, keď je služba už vybraná.
0x03	DESTROY (ZRUŠENÝ)	Aplikácia sa musí zastaviť, ale môže sa povoliť možnosť na jej riadne skončenie. Pokusy o spustenie aplikácie musia zlyhať.
0x04	KILL (UKONČENÝ)	Aplikácia sa musí zastaviť, len čo je to možné. Pokusy o spustenie aplikácie musia zlyhať.
0x05	PREFETCH (PREDPRIPRAVENÝ)	Súbory aplikácie sa majú kešovať prijímačom, ak je to možné. Aplikácia sa nesmie spustiť a pokusy o jej spustenie musia zlyhať.
0x06	REMOTE (VZDIALENÝ)	Aplikácia nie je k dispozícii v aktuálnom prenosovom toku, a preto bude k dispozícii až po nastavení na nový prenosový tok, alebo ak je uložená a signalizovaná ako plne spustiteľná z pamäte keš.
0x07	DISABLED (ZABLOKOVANÝ)	Aplikácia sa nesmie spustiť a pokusy o jej spustenie musia zlyhať.
0x08	PLAYBACK_AUTOSTART (PREHRÁVANIE_AUTOŠTART)	Aplikácia nesmie bežať ani priamo z vysielania, ani v režime časového posunu. Keď je nahrávka prehrávaná zo záznamu, aplikácia sa musí prezentovať, ako by to bol autoštart.
0x09 až 0xFF		reserved_future_use (rezervované na budúce použitie)

Špecifikácia platformy sa má definovať s informáciou, ktoré z týchto riadiacich kódov sa môžu použiť.

5.2.4.2 Kódovanie MPEG-2

Riadiaci kód aplikácie je signalizovaný prostredníctvom poľa `application_control_code` na použitie v AIT (pozri článok 5.3.4.6). Hodnoty sú uvedené v tabuľke 3.

5.2.4.3 Kódovanie XML

Kódovanie riadiaceho kódu aplikácie do XML je definované nasledujúcim zápisom a hodnoty sú definované v tabuľke 3:

```
<xsd:simpleType name="ApplicationControlCode">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="AUTOSTART"/>
    <xsd:enumeration value="PRESENT"/>
    <xsd:enumeration value="DESTROY"/>
    <xsd:enumeration value="KILL"/>
    <xsd:enumeration value="PREFETCH"/>
    <xsd:enumeration value="REMOTE"/>
    <xsd:enumeration value="DISABLED"/>
    <xsd:enumeration value="PLAYBACK_AUTOSTART"/>
  </xsd:restriction>
</xsd:simpleType>
```

5.2.5 Profily platformy

5.2.5.1 Sémantika

Niektoré špecifikácie platformy môžu definovať celý rad rôznych profilov a potenciálne rôzne verzie týchto profilov. Tieto polia definujú minimálny profil platformy a verzie tohto profilu požadovaného aplikáciou. Špecifikácie platformy, ktoré používajú túto informáciu, potrebujú definovať jej význam v ich kontexte.

profil aplikácie (angl. **application_profile**): pole je celočíselná hodnota, ktorá predstavuje profil platformy požadovaný aplikáciou; signalizuje, že prijímač, v ktorom je implementovaný jeden z profilov uvedených v tejto slučke, je schopný spracovať aplikáciu

hlavná verzia (angl. **version.major**): pole obsahuje číselnú hodnotu hlavného subpoľa čísla verzie profilu

vedľajšia verzia (angl. **version.minor**): pole obsahuje číselnú hodnotu vedľajšieho subpoľa čísla verzie profilu

mikroverzia (angl. **version.micro**): pole obsahuje číselnú hodnotu mikrosubpoľa čísla verzie profilu

Štyri uvedené polia predstavujú minimálny profil, ktorý zaistí použiteľnosť aplikácie. Aplikácie môžu testovať vlastnosti zistených vyšších profilov, spätne kompatibilných profilov, a využívať ich. Koncové zariadenie musí len spúšťať aplikácie v prípade, keď je nasledujúci výraz pravdivý aspoň v jednom zo signalizovaných profilov:

(application_profile Î terminal_profile_set)

Ú[application_version.major < terminal_version.major(application_profile)]

Ú[application_version.major = terminal_version.major(application_profile)]

Ú[application_version.minor < terminal_version.mirror(application_profile)]

Ú[application_version.minor = terminal_version.minor(application_profile)]

Ú[application_version.micro < terminal - version.micro(application_profile)],

kde:

Î znamená „patrí do súboru“;

Ú znamená „logickú hodnotu AND“;

Ú znamená „logickú hodnotu OR“.

POZNÁMKA. – Kódovanie týchto hodnôt sa môže meniť podľa typu aplikácií a je definované technickou špecifikáciou na interaktívne služby.

5.2.5.2 Kódovanie MPEG-2

Profily sú kódované ako súčasť deskriptora aplikácie.

Tabuľka 4 – Kódovanie profilu aplikácie

	Počet bitov	Identifikátor	Hodnota
application_profiles_length	8	uimsbf	
for(i=0; i<N; i++) {			
application_profile	16	uimsbf	
version.major	8	uimsbf	
version.minor	8	uimsbf	
version.micro	8	uimsbf	
}			

5.2.5.3 Kódovanie XML

Kódovanie XML profilov je nasledujúce:

```
<xsd:complexType name="MhpVersion">
  <xsd:sequence minOccurs="1">
```

```

<xsd:element name="profile" type="ipi:Hexadecimal16bit" />
<xsd:element name="versionMajor" type="ipi:Hexadecimal8bit" />
<xsd:element name="versionMinor" type="ipi:Hexadecimal8bit" />
<xsd:element name="versionMicro" type="ipi:Hexadecimal8bit" />
</xsd:sequence>
</xsd:complexType>

```

POZNÁMKA. – Názov typu je historický.

5.2.6 Viditeľnosť aplikácie

5.2.6.1 Sémantika

Pole viditeľnosti špecifikuje, či je aplikácia vhodná na ponuku pre koncového používateľa na jeho rozhodnutie, ak sa aplikácia má spustiť. Zoznam povolených hodnôt tohto poľa uvádza tabuľka 5.

POZNÁMKA. – To isté platí aj na akékoľvek všeobecné spustenie menu aplikácie prostredníctvom obsahu alebo služby poskytovateľa, alebo výrobcu koncového zariadenia.

Tabuľka 5 – Definícia stavov viditeľnosti na aplikácie

Kódovanie MPEG-2	Kódovanie XML	Opis
00	NOT_VISIBLE_ALL	Aplikácia nesmie byť viditeľná aplikáciám prostredníctvom zoznamu aplikácie API, ak je podporované také API koncovým zariadením alebo používateľom prostredníctvom navigátora s výnimkou chybového hlásenia alebo prihlásenia zariadenia atď.
01	NOT_VISIBLE_USERS	Aplikácia nesmie byť viditeľná používateľom, ale musí byť viditeľná aplikáciám prostredníctvom zoznamu aplikácie API, ak je podporované také API koncovým zariadením.
10		rezervované na budúce používanie
11	VISIBLE_ALL	Aplikácia môže byť viditeľná používateľom a musí byť viditeľná aplikáciám prostredníctvom zoznamu aplikácie API, ak je podporované také API koncovým zariadením.

Toto pole je voliteľné.

5.2.6.2 Kódovanie MPEG-2

Viditeľnosť aplikácie je zakódovaná v poli viditeľnosti deskriptora aplikácie v AIT (pozri článok 5.3.5.3). Možné hodnoty poľa sú uvedené v tabuľke 5.

5.2.6.3 Kódovanie XML

Kódovanie viditeľnosti XML je nasledujúce:

```

<xsd:simpleType name="VisibilityDescriptor">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="NOT_VISIBLE_ALL" />
    <xsd:enumeration value="NOT_VISIBLE_USERS" />
    <xsd:enumeration value="VISIBLE_ALL" />
  </xsd:restriction>
</xsd:simpleType>

```

Definovanie týchto hodnôt uvádza tabuľka 5 v článku 5.2.6.1.

5.2.7 Priorita aplikácie

5.2.7.1 Sémantika

Priorita aplikácie označuje relatívnu prioritu medzi aplikáciami signalizovanými v službe:

- keď je viac ako jedna aplikácia s rovnakým označením aplikácie v službe, musí táto priorita určiť poradie spúšťania aplikácií;

- keď existujú nedostatočné prostriedky na súčasný priebeh viacerých aplikácií, musí táto priorita určiť, ktoré aplikácie treba zastaviť alebo pozastaviť;
- väčšia celočíselná hodnota udáva vyššiu prioritu;
- keď majú dve aplikácie rovnaké označenie aplikácie a rovnakú prioritu, môže koncové zariadenie vykonať výber v závislosti od implementácie, ktorou začne.

POZNÁMKA. – Špecifikácie platformy môžu definovať špeciálnu sémantiku pri konkrétnych hodnotách priority.

5.2.7.2 Kódovanie MPEG-2

Priorita aplikácie je zakódovaná v poli `application_priority` deskriptora `application_descriptor` (pozri článok 5.3.5.3).

5.2.7.3 Kódovanie XML

Priorita aplikácie je zakódovaná v poli `priority` deskriptora aplikácie (pozri článok 5.4.4.4).

5.2.8 Ikony aplikácie

5.2.8.1 Sémantika

Aplikácia môže mať priradenú jednu ikonu alebo viac ikon. Formát obsahu týchto možných ikon musí byť PNG. Špecifikácie platformy môžu stanoviť ďalšie obmedzenia formátu obsahu ikon.

Každá ikona má lokátor ikony a súbor príznakov, ktoré identifikujú veľkosť ikony a pomer strán ikony.

Lokátor ikony je prvá časť reťazca, ktorá určuje umiestnenie súborov ikony. Umiestnenie je relatívne a závisí od typu aplikácie. Lokátor ikony nesmie končiť lomkou `/`.

Názvy súborov ikony sú kódované štandardným spôsobom:

```
filename      = icon_locator "/"dvb.icon." hex_string
hex_string    = 4*4hex
hex           = digit | "A" | "B" | "C" | "D" | "E" | "F" | "a" | "b" | "c" | "d" | "e" | "f"
digit        = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
```

Súbor ikony musí obsahovať len jednu ikonu. Ikona uvedená v súbore ikony má formát definovaný 4 hexadecimálnymi číslicami v dodatku názvu jej súboru. Hodnota tohto dodatku je daná zodpovedajúcim kódovaním príznakov ikony MPEG-2 (pozri tabuľku 7).

5.2.8.2 Kódovanie MPEG-2

Informácia týkajúca sa ikon aplikácie je zakódovaná v `application_icons_descriptor`.

Tabuľka 6 – Syntax deskriptora ikon aplikácie

	Počet bitov	Identifikátor	Hodnota
application_icons_descriptor() {			
descriptor_tag	8	uimsbf	0x0B
descriptor_length	8	uimsbf	
icon_locator_length	8	uimsbf	
for (i=0; i<N; i++) {			
icon_locator_byte	8	uimsbf	
}			
icon_flags	16	bslbf	
for (i=0; i<N; i++) {			
reserved_future_use	8	bslbf	
}			
}			

Možné hodnoty poľa icon_flags sú uvedené v tabuľke 7.

Tabuľka 7 – Definícia rôznych príznakov ikony

Bity príznaku ikony	Opis veľkosti ikony a pomer strán
0000 0000 0000 0001	32 x 32 na štvorcové zobrazenie pixelov
0000 0000 0000 0010	32 x 32 na vysielané pixely na displeji 4 : 3 (pozri poznámku)
0000 0000 0000 0100	24 x 32 na vysielané pixely na displeji 16 : 9
0000 0000 0000 1000	64 x 64 na štvorcové zobrazenie pixelov
0000 0000 0001 0000	64 x 64 na vysielané pixely na displeji 4 : 3 (pozri poznámku)
0000 0000 0010 0000	48 x 64 na vysielané pixely na displeji 16 : 9
0000 0000 0100 0000	128 x 128 na štvorcové zobrazenie pixelov
0000 0000 1000 0000	128 x 128 na vysielané pixely na displeji 4 : 3 (pozri poznámku)
0000 0001 0000 0000	96 x 128 na vysielané pixely na displeji 16 : 9
0000 0010 0000 0000	256 x 256 na štvorcové zobrazenie pixelov
0000 0100 0000 0000	256 x 256 na vysielané pixely na displeji 4 : 3 (pozri poznámku)
0000 1000 0000 0000	192 x 256 na vysielané pixely na displeji 16 : 9
xxxx 0000 0000 0000	reserved_future_use (rezervované na budúce využívanie)
POZNÁMKA. – Približne 15/16 pixelový pomer strán systému 50 Hz.	

Keď pole icon_flags deskriptora ikon aplikácie má hodnotu indikujúcu prítomnosť viacerých ikon, potom každá z uvedených ikon má mať svoj vlastný súbor ikony. Napríklad ak icon_flags má hodnotu 0 x 0005, potom určený adresár icon_locator má obsahovať dva súbory s názvami dvb.icon.0004 (24 x 32 pixelová plošná interpretácia) a dvb.icon.0001 (32 x 32 pixelová štvorcová interpretácia).

5.2.8.3 Kódovanie XML

Informácia o ikone je zakódovaná v jednom alebo viacerých prvkoch IconDescriptor:

```
<xsd:complexType name="IconDescriptor">
  <xsd:attribute name="filename" type="xsd:string" use="required"/>
  <xsd:attribute name="size" type="xsd:unsignedShort" use="optional"/>
  <xsd:attribute name="aspectRatio" type="mhp:AspectRatio" use="optional"/>
</xsd:complexType>
```

POZNÁMKA 1. – Kódovania MPEG-2 a XML sú zámerne odlišné. Kódovanie MPEG-2 obsahuje iba prefix icon_locator a zvyšok názvu súboru filename je vypočítaný. Kódovanie XML obsahuje úplnú adresu URL.

POZNÁMKA 2. – Typ mhp:AspectRatio je definovaný v článku 5.4.4.7 tohto dokumentu.

Veľkosť a atribúty aspectRatio sú definované ako voliteľné, pretože sú určené 4 hexadecimálnymi číslicami v dodatku názvu jeho súboru, ako je definované v tabuľke 7.

5.2.9 Obmedzenia grafiky

5.2.9.1 Sémantika

Aplikácie sa môžu obmedzovať v grafických rozlíšeníach, ktoré podporujú, alebo v ich schopnosti spracovať zmeny v grafike alebo konfigurácii videa.

Rozsah grafického obmedzenia aplikácie je určený pomocou niekoľkých polí, definovaných neskôr.

Pri aplikáciách, v ktorých sa táto informácia nesignalizuje, sa predpokladá, že majú nasledujúce grafické obmedzenia:

- podporovanie videa so štandardným rozlíšením;
- nie je možné spustenie bez viditeľného UI;
- nie je možné spracovať zmenené grafické konfigurácie;
- nie je možné nastaviť externe ovládané video.

Pri aplikáciách, ktoré nemajú signalizovaný súbor konfigurácií grafiky, sa predpokladá, že sa neberie do úvahy predvolená konfigurácia grafiky, nemôže sa použiť grafika, alebo v plnom rozsahu sa podporuje konfigurácia grafiky definovaná v tomto dokumente a je adekvátne odskúšaná.

5.2.9.1.1 Podporované konfigurácie grafiky

Podporované konfigurácie grafiky na aplikáciu sú dané zoznamom jednej alebo viacerých z podporovaných konfigurácií (uvedených v tabuľke 9). Konfigurácie celej obrazovky sú zoradené od najviac preferovanej do najmenej preferovanej.

5.2.9.1.2 Spustenie bez viditeľného UI

Príznak `can_run_without_visible_ui` indikuje, či aplikácia musí zobrazíť používateľské rozhranie. Ak je tento príznak nastavený, potom môže aplikácia úspešne bežať bez viditeľného používateľského rozhrania. Ak nie je tento príznak nastavený, potom aplikácia môže úspešne bežať len s viditeľným používateľským rozhraním. Aplikácie signalizované s týmto nastaveným príznakom zodpovedajú za určenie vhodného okamihu zobrazenia používateľského rozhrania podľa požiadaviek definovaných v špecifikácii platformy.

5.2.9.1.3 Nastavenie zmeny konfigurácií grafiky

Príznak `handles_configuration_changed` indikuje, či je aplikácia schopná spracovať zmeny v konfigurácii grafiky. Ak je tento príznak nastavený, potom môže aplikácia nastaviť zmeny v konfigurácii grafiky v rámci podporovaných konfigurácií grafiky na tieto aplikácie (pozri článok 5.2.9.1.1). Keď nie je nastavený tento príznak, potom sa v tomto prípade okamžite nastaví predvolená konfigurácia grafiky a zobrazenie bude správne iba v rámci tejto konfigurácie grafiky.

5.2.9.1.4 Nastavenie externe ovládaného videa

Príznak `handles_externally_controlled_video` indikuje, či aplikácia môže úspešne bežať, keď je prehľadanie videa riadené druhou externou aplikáciou na túto službu. Keď je tento príznak nastavený, potom sa aplikácia môže zobrazíť za týchto podmienok. Príklady predstavujú obraz v obraze a obraz mimo obrazu.

5.2.9.2 Kódovanie MPEG-2

Deskriptor deskriptora graphics_constraints_descriptor signalizuje obmedzenia, ktoré sa vzťahujú na aplikáciu.

Tabuľka 8 – Syntax deskriptora obmedzenia grafiky

	Počet bitov	Identifikátor	Hodnota
graphics_constraints_descriptor() {			
descriptor_tag	8	uimsbf	0x14
descriptor_length	8	c	
reserved_future_use	5	bslbf	
can_run_without_visible_ui	1	bslbf	
handles_configuration_changed	1	bslbf	
handles_externally_controlled_video	1	bslbf	
for(i=0;i<N;i++) {			
graphics_configuration_byte	8	uimsbf	
}			
}			

Podporované konfigurácie grafiky na aplikáciu sú dané zoznamom jednej alebo viacerých hodnôt uvedených v tabuľke 9.

Tabuľka 9 – Hodnoty bajtu konfigurácie grafiky

Hodnota	Význam
0	Rezervované
1	Štandardné rozlíšenie na celú obrazovku
2	Celá obrazovka 960x540
3	Celá obrazovka 1 280x720
4	Celá obrazovka 1 920x1080
5 až 31	Rezervované na budúce použitie projektu DVB
32 až 255	Rezervované na budúce používanie

5.2.9.3 Kódovanie XML

Kódovanie XML nie je definované v tomto dokumente.

5.2.10 Použitie aplikácie

5.2.10.1 Sémantika

Aplikácia poskytuje konkrétne, dobre známe služby; napríklad teletext, EPG alebo četovanie. Koncové zariadenia môžu poskytovať skrátené príkazy na spustenie týchto služieb, napríklad tlačidlo diaľkového ovládača. Koncové zariadenia môžu tiež poskytovať natívne používateľské rozhranie ponúkajúce prístup k týmto službám.

5.2.10.2 Kódovanie MPEG-2

Tabuľka 10 – Používanie deskriptora aplikácie

	Počet bitov	Identifikátor	Hodnota
application_usage_descriptor() {			
descriptor_tag	8	uimsbf	0x16
descriptor_length	8	uimsbf	
usage_type	8	uimsbf	
}			

príznak deskriptora (angl. **descriptor_tag**): osembitové pole s hodnotou 0 x 16 identifikuje deskriptor

typ použitia (angl. **usage_type**): osembitové pole označuje službu, ktorá je poskytovaná aplikáciou; musí sa označovať podľa tabuľky 11

Tabuľka 11 – Kódovanie typov používania aplikácie MPEG-2

Typy hodnôt	Opis
0x00	rezervované
0x01	aplikácia digitálneho textu
0x02 až 0x7F	rezervované na budúce používanie
0x80 až 0xFF	použiteľné pri špecifikáciách platformy (pozri poznámku)
POZNÁMKA. – Špecifikácia platformy má definovať doménu, v ktorej sú tieto hodnoty použiteľné, napr. pomocou špecifického vysielania dát ID.	

Špecifikácie platformy majú definovať, ktoré z týchto typov používania sú aplikovateľné.

Ak nie je príznak `application_usage_descriptor` prítomný, aplikácia neposkytuje konkrétnu dobre známu službu.

5.2.10.3 Kódovanie XML

Typ použitia aplikácie sa kóduje do XML prvkom `ApplicationUsageDescriptor`:

```
<xsd:complexType name="ApplicationUsageDescriptor">
  <xsd:sequence>
    <xsd:element name="ApplicationUsage" type="xsd:anyURI" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Prvok `ApplicationUsage` indikuje službu, ktorá je poskytovaná aplikáciou. Musí sa označovať podľa tabuľky 12.

Tabuľka 12 – Kódovanie typov používania aplikácie MPEG-2

Hodnota <code>ApplicationUsage</code>	Opis
<code>urn:dvb:mhp:2009:digitalText</code>	Aplikácia digitálneho textu

Špecifikácie platformy majú rozširovať typy použitia hodnotou predpony s príslušným identifikátorom namespace podľa špecifikácie platformy.

5.2.11 Uložené aplikácie

5.2.11.1 Sémantika

Uložiteľné aplikácie majú rovnaký model signalizácie ako normálna aplikácia definovaná v tomto dokumente. Koncové zariadenie, ktoré neposkytuje uloženie, alebo nerozpoznáva tieto prípony, vníma uložitelné aplikácie ako použiteľné aplikácie, ktoré sa môžu potenciálne spustiť v rámci vysielania. Signalizácia uvedená v tomto článku rozširuje možnosti signalizácie uvedené v iných článkoch tohto dokumentu.

Primárnym cieľom uložených aplikácií je zlepšenie správania sa aplikácií pri ich spustení poskytovaných prostredníctvom vysielania. Táto funkcia nie je až tak dôležitá na aplikácie poskytované prostredníctvom širokopásmového pripojenia.

5.2.11.1.1 Životný cyklus uložených aplikácií

Uložené aplikácie sa môžu vysielat' ako súvisiace s vysielaním alebo ako samostatné.

Ak sa aplikácia vysielala ako súvisiaca s vysielaním, potom je životný cyklus aplikácie riadený vysielacou službou. Takéto aplikácie sú vhodnejšie na uloženie, čo ale nie je vhodné na samostatnú aplikáciu. Na uložené aplikácie súvisiace s vysielaním sa musí používať signalizácia vysielaná pri spustení uloženej aplikácie. Preto je na signalizáciu takejto aplikácie potrebné uložiť minimum informácií.

Aplikácia vysielaná ako samostatná sa môže úspešne spúšťať používateľom nezávisle od vysielacej služby. Aplikácie s touto vlastnosťou sa môžu tiež spúšťať ako aplikácie súvisiace s vysielaním v prípade, keď aktuálne vybraná služba ich má začlenené v rámci svojej signalizácie. Na spustenie uložených samostatných aplikácií musí signalizácia prichádzať z údajov uložených v tabuľke AIT.

5.2.11.1.2 Vytváranie verzií aplikácie

Pole verzie obsahuje číslo verzie aplikácie. Toto číslo začína nulou a zvyšuje sa o jeden zakaždým, keď sa nejaký zo súborov začlení do súboru opisu aplikácie (pozri článok 5.2.12) zmenou alebo zmenou samotného obsahu súboru opisu aplikácie. Použité hodnoty sa nesmú nikdy opakovať. V prípade, že sa číselný rozsah vyčerpá, musí sa použiť nový `application_id`.

Príznak `is_launchable_with_older_version` indikuje, či kešovaná staršia verzia aplikácie sa môže spustiť, aj keď je vo vysielaní signalizovaná vyššia verzia. Ak áno, koncové zariadenie spustí kešovanú aplikáciu, pri ktorej je číslo verzie kešovanej aplikácie nižšie alebo rovnajúce sa číslu verzie vysielanej aplikácie. Ak je číslo verzie kešovanej aplikácie vyššie ako číslo verzie vysielanej aplikácie, kešovaná aplikácia sa nesmie spustiť. Ak príznak nie je nastavený, aplikácia sa nesmie spustiť.

POZNÁMKA. – Ak je nastavený príznak, uložená aplikácia zodpovedá za zvládnutie konfliktu verzií medzi kódom uloženej aplikácie a vysielanými dátami aplikácie.

5.2.11.1.3 Spúšťanie aplikácií z pamäte keš

Príznak `launchable_completely_from_cache` indikuje, či je požadované prepojenie na prenosový protokol. Ak je nastavený príznak, znamená to, že táto aplikácia sa môže spustiť iba z pamäte keš, bez prepojenia na prenosový protokol signalizovaný v signalizácii aplikácie, za predpokladu, že všetky dôležité súbory boli kešované. Ak nie je nastavený príznak, prepojenie na prenosový protokol sa požaduje z dôvodu spustenia tejto aplikácie ako aplikácie súvisiacej s vysielaním. Tento príznak sa použije len v prípade, keď je aplikácia spustená ako aplikácia súvisiaca s vysielaním; to je ignorované pri uložení aplikácie do uloženej služby, pretože všetky aplikácie signalizované ako samostatné aplikácie môžu bežať bez prepojenia na prenosový protokol pri spustení ako súčasť uloženej služby.

Tento príznak sa musí nastaviť len vtedy, keď je príznak `not_launchable_from_broadcast` tiež nastavený.

POZNÁMKA. – Tento príznak sa má nastaviť len v aplikáciách, ktoré vôbec nepoužívajú objektový karusel.

Príznak `not_launchable_from_broadcast` indikuje, či aplikácia sa môže úspešne spustiť predtým, ako je úplne kešovaná. Ak je príznak nastavený, znamená to, že doručovacie parametre tejto aplikácie sú také, že nie sú vhodné na spustenie aplikácie, pokiaľ nebola úplne kešovaná/uložená. Ak nie je nastavený príznak, potom kešovanie poskytuje niektoré výhody, ale nie je nevyhnutné.

Aplikácie v uložených službách a aplikácie, v ktorých je príznak `not_launchable_from_broadcast` nastavený, musia sa spustiť len ak koncové zariadenie uložilo všetky súbory, ktoré sú uvedené v súbore opisu aplikácie ako kritické.

Tabuľka 13 – Kombinácie príznaku deskriptora uloženia

<code>not_launchable_from_broadcast</code>	<code>launchable_completely_from_cache</code>	<code>is_launchable_with_older_version</code>	Opis
0	0	0	Normálny prípad.
0	0	1	Nesmie sa signalizovať.
0	1	0	Nesmie sa signalizovať.
0	1	1	Nesmie sa signalizovať.
1	0	0	Spustí sa, keď je signalizovaná uložená verzia.
1	0	1	Prebieha, keď je signalizovaná alebo uložená staršia verzia.
1	1	0	Prebieha kompletne z pamäte keš, pokiaľ je signalizovaná verzia uložená. Aplikáciu nie je možné uložiť z dôvodu nedostupnosti objektového karuselu na aktuálnu službu.
1	1	1	Prebieha, keď je signalizovaná alebo uložená staršia verzia. Aplikáciu nie je možné uložiť z dôvodu nedostupnosti objektového karuselu na aktuálnu službu.
Keď je nastavený, príznak označuje, že súbory sú prítomné, ale bitová rýchlosť je príliš nízka.	Keď je nastavený, príznak označuje, že súbory nie sú prítomné v aktuálnom vysielaní.		

5.2.11.1.4 Priorita uloženia

Priorita uloženia aplikácie označuje dôležitosť tejto aplikácie na uloženie v porovnaní s ostatnými signalizovanými aplikáciami tejto služby. To má význam iba na aplikácie, ktoré boli aktívne kešované koncovým zariadením, pretože inak sa neberú do úvahy.

Vyššie hodnoty znamenajú väčšiu dôležitosť aplikácie z hľadiska uloženia. Správanie sa aplikácií v prípade rovnakej priority závisí od implementácie.

5.2.11.2 Kódovanie MPEG-2

Informácia o možnostiach uloženia aplikácie sa nachádza v deskriptore uloženia aplikácie.

Tabuľka 14 – Syntax deskriptora uloženia aplikácie

	Počet bitov	Identifikátor	
<code>application_storage_descriptor() {</code>			
<code> descriptor_tag</code>	8	<code>uimsbf</code>	<code>0x10</code>
<code> descriptor_length</code>	8	<code>uimsbf</code>	
<code> storage_property</code>	8	<code>uimsbf</code>	
<code> not_launchable_from_broadcast</code>	1	<code>bslbf</code>	
<code> launchable_completely_from_cache</code>	1	<code>bslbf</code>	
<code> is_launchable_with_older_version</code>	1	<code>bslbf</code>	
<code> Reserved</code>	5	<code>bslbf</code>	
<code> Reserved</code>	1	<code>bslbf</code>	
<code> Version</code>	31	<code>uimsbf</code>	
<code> Priority</code>	8	<code>uimsbf</code>	
<code>}</code>			

Pole `storage_property` je nasledujúco kódované:

Tabuľka 15 – Sémantika hodnôt vlastností uloženia

<code>storage_property</code>	Vlastnosti
0	súvisiaca s vysielaním
1	samostatná
2 až 255	rezervované

5.2.11.3 Kódovanie XML

Kódovanie možností uloženia aplikácie do XML je daná prvkom `StorageCapabilities`:

```
<xsd:complexType name="StorageCapabilities">
  <xsd:sequence minOccurs="0">
    <xsd:element name="storageProperty" type="mhp:StorageType"/>
    <xsd:element name="isStorable" type="xsd:boolean"/>
    <xsd:element name="canCache" type="xsd:boolean"/>
  </xsd:sequence>
  <xsd:attribute name="launchableFromBroadcast" type="xsd:boolean" use="required"/>
  <xsd:attribute name="launchableCompletelyFromCache" type="xsd:boolean" use="required"/>
  <xsd:attribute name="launchableWithOlderVersion" type="xsd:boolean" use="required"/>
</xsd:complexType>
```

POZNÁMKA 1. – Typ `mhp:StorageType` je definovaný v článku 5.4.4.10 tohto dokumentu.

POZNÁMKA 1. – Priorita uloženia nie je definovaná v kódovaní XML.

5.2.12 Súbor opisu aplikácie

5.2.12.1 Opis

Súbor opisu aplikácie (ADF) poskytuje zoznam súborov, ktoré je potrebné uložiť pri aplikácii, rovnako ako aj ďalšie potrebné súvisiace informácie. Na zápis sa používa syntax XML.

V aplikáciách, ktoré sa môžu uložiť, sa musí súbor opisu aplikácie umiestniť v tom istom karuseli ako aplikácia.

POZNÁMKA. – Súbor opisu aplikácie nemá obsahovať duplikáty všetkých potrebných informácií na bežanie aplikácie. Koncové zariadenie musí tiež použiť informácie nachádzajúce sa v signalizácii aplikácie pri inštalácii aplikácie.

Tam, kde je súbor uvedený v súbore opisu aplikácie vo viac ako jednej aplikácii a je uložený, koncové zariadenie musí zabezpečiť, aby každá aplikácia našla správnu verziu súboru v danej aplikácii. Verzia súboru dostupná jednej aplikácii sa nesmie zmeniť žiadnou zmenou verzie súboru dostupnou v inej aplikácii, ktorá môže mať rovnaký súbor.

5.2.12.2 Názov súboru opisu aplikácie a jeho umiestnenie

Umiestnenie ADF sa má definovať v špecifikácii platformy. Podľa konvencie je názov ADF v tvare:

```
'dwb.storage.00000000.aaaa'
```

kde:

00000000 is the `organisation_id` of the application as a 8 character hexadecimal string

aaaa is the `application_id` as a 4 character hexadecimal string

Identifikátory `organisation_id` a `application_id` sa musia doplniť nulami na začiatku reťazca na určenú dĺžku.

Na zakódovanie organisation_id a application_id sa musia použiť malé hexadecimálne znaky.

5.2.12.3 Syntax

Syntax súboru opisu aplikácie je definovaná týmto XML DTD.

PublicLiteral používaný na určenie tohto DTD v opise typu dokumentu – súbory XML:

```
"-//DVB//DTD Application Description File 1.0//EN"
```

a URL pre SystemLiteral:

and the URL for the SystemLiteral is:

```
"http://www.dvb.org/mhp/dtd/applicationdescriptionfile-1-0.dtd"
<!ENTITY % object "(dir|file)">
<!-- the main element for the application description -->
<!ELEMENT applicationdescription (%object;)+>
<!ATTLIST applicationdescription version NMTOKEN #REQUIRED>
<!ELEMENT dir (%object;)*>
<!ATTLIST dir
  name CDATA #REQUIRED
  priority NMTOKEN #IMPLIED
>
<!ELEMENT file EMPTY>
<!ATTLIST file
  name CDATA #REQUIRED
  priority NMTOKEN #IMPLIED
  size NMTOKEN #REQUIRED
>
```

5.2.12.4 Sémantika

Verzia: Celé desiatkové číslo označuje číslo verzie tejto aplikácie.

Hodnota tohto atribútu nesmie obsahovať nuly na začiatku (pokiaľ nie je "0") a musí tiež zodpovedať číslu signalizovanej verzie v poli verzie deskriptora uloženej aplikácie vo vstupnej AIT tejto aplikácie; ak tomu tak nie je, súbor opisu aplikácie je neplatný. Toto pole umožňuje autorom aplikácie zaistiť správnu signalizáciu čísla verzie v AIT. Ak tomu tak nie je, tak nebudú uložené žiadne súbory.

Názov: Tento atribút obsahuje názov objektu v rámci systémového súboru (adresár alebo súbor), ktorý je uložený. Je to názov objektu v jeho zaslanom adresári a z toho dôvodu neobsahuje žiadnu informáciu o ceste k jeho adresáru. Len na atribút názvu súboru prvku môže posledný znak názvu obsahovať zástupný znak "*". Tento znak vyhovuje ľubovoľnému reťazcu, vrátane prázdneho reťazca.

Ak je názov "." alebo "..", cesta obsahuje oddeľovací znak "/", alebo obsahuje znak NUL (U+0000), potom prijímač musí zamietnuť ADF ako neplatný.

POZNÁMKA 1. – Na pomenovanie typov objektu nie sú poskytnuté žiadne prvky, ako sú Stream alebo StreamEvent, ktoré sa nachádzajú v objektovom karuseli, a preto nie je určený žiadny mechanizmus na požiadavku, aby boli uložené objekty Stream a StreamEvent.

POZNÁMKA 2. – Výpis adresára objektu v súbore neobsahuje nič o tých obsahoch adresára, ktoré nie sú uvedené v súbore.

POZNÁMKA 3. – Špecifikácie odvolávajúce sa na aktuálny dokument môžu obsahovať obmedzenia na názvy súborov, ktoré sa môžu používať.

Cesty sa vzťahujú na adresár, ktorý obsahuje súbor opisu aplikácie, t. j. prvky <file> a <dir> priamo vnútri prvku <applicationdescription> s odvolaním sa na súbory a adresáre v rovnakom adresári ako súbor opisu aplikácie.

Priorita: Tento atribút opisuje dôležitosť uloženia objektu. Hodnota musí byť v rozsahu od 0 do 255, vrátane. Ak je hodnota mimo tento rozsah, potom je súbor opisu aplikácie neplatný. Hodnota nula znamená, že uloženie objektu je kritické (t. j. z uloženia objektov nevyplýva žiadna výhoda, pokiaľ sú tieto časti uložené). Vyššie hodnoty znamenajú nižšiu prioritu uloženia.

Predvolená hodnota atribútu priority je nulová (t. j. kritická).

Priorita objektu sa prenáša priamo zo zaslaného adresára.

Veľkosť: Tento atribút definuje veľkosť súboru v bajtoch alebo súborov, kde atribút názvu obsahuje zástupný znak.

5.3 Tabuľka MPEG-2 a syntax sekcie

5.3.1 Prehľad

5.3.1.1 Prehľad spoločnej signalizácie

Minimálne požiadavky na signalizáciu pri všetkých aplikáciách sú zhrnuté týmto spôsobom:

- PMT so signalizáciou deskriptora aplikácie na identifikáciu zložky služby nesúca tabuľku informácií o aplikácii;
- tabuľka informácií o aplikácii s nasledujúcimi informáciami v jej spoločnej slučke deskriptora;
 - transport_protocol_descriptor: všetky opisy aplikácií musia byť v rozsahu aspoň jedného transport_protocol_descriptor; tie sa môžu umiestniť v jednej alebo oboch slučkách deskriptora;
- tabuľka informácií o aplikácii s nasledujúcimi informáciami v jej slučke deskriptora informácií o aplikácii:
 - application_descriptor;
 - application_name_descriptor.

5.3.1.2 Prehľad doplnkovej signalizácie na aplikácie prenášané prostredníctvom OC

V prvej slučke deskriptora "common" alebo vnútornej slučke deskriptora "application":

- transport_protocol_descriptor so selektorom bajtov obsahujúcim konkrétne informácie OC, ako je definované v tabuľke 31.

5.3.1.3 Ako pridať nový návrh (informatívne)

Signalizácia návrhu vychádza z predpokladu jeho rozširovania s ohľadom na pridávanie ďalších aplikácií a podporovaných prenosových protokolov. Oblasti, ktoré je potrebné riešiť pri týchto aktivitách, sú sumarizované neskôr.

Na pridanie ďalších prenosových protokolov je potrebné:

- rozšíriť tabuľku 30 "Sémantika selektora bajtov";
- definovať možnosti ďalších špeciálnych deskriptorov, napríklad IP_signalling_descriptor.

Na pridanie ďalších aplikácií je potrebné:

- v prípade potreby definovať ďalšie špeciálne deskriptory (pozri článok 10.9 "DVB-J špecifické deskriptory" v špecifikácii MHP [i.1] ako príklady);
- definovať špecifické riadiace kódy životného cyklu typu aplikácie v článku 5.2.4 "Kódy riadenia aplikácie".

Ak sú konštantné hodnoty registrované v aktuálnom dokumente, je potrebné rozšíriť tabuľku 38 "Register konštantných hodnôt".

5.3.2 Špecifická informácia o programe

Elementárny tok vnútornej slučky PMT do služby DVB podporujúci jednu alebo viac aplikácií musí odkazovať toky na nasledujúce:

- umiestnenie toku prenášajúceho tabuľku informácií o aplikácii;
- umiestnenie toku(ov) prenášajúceho kód a dáta aplikácie.

5.3.2.1 Tok signalizácie aplikácie

Základný tok informácií do vstupnej PMT, ktorá opisuje základný tok obsahujúci tabuľku informácií o aplikácii, má nasledujúce charakteristiky:

- stream_type je nastavený na 0 x 05 (ISO/IEC 13818-1 [3], neverejná sekcia);
- application_signalling_descriptor (pozri článok 5.3.5.1).

V ňom môže byť viacej základných tokov, ktoré prenášajú informácie o signalizácii aplikácie na službu.

5.3.2.2 Toky vysielaných dát

Minimálna signalizácia v PMT súvisiaca s vysielanými zložkami dát je hodnota poľa PMT stream_type požadovaná špecifikáciou na vysielanie dát DVB (EN 301 192 [2]) s prenosovým protokolom. Úplné detaily o protokole vysielania dát umiestnené v jeho "hlavnej" zložke atď. sú uvedené v AIT (pozri článok 5.3.4 "Tabuľka informácií o aplikácii").

Nepovinne – PMT môže voliteľne obsahovať deskriptory data_broadcast_id_descriptor.

POZNÁMKA. – Prítomnosť deskriptorov data_broadcast_id_descriptor umožňuje prijímačom začať zostavovať systémový súbor, ktorý poskytuje aplikácie súčasne so získaním AIT, pričom identifikuje, ktoré aplikácie sú dôležité. Povolenie tejto súbežnej prevádzky umožňuje prijímačom urýchliť aktiváciu interaktívnej aplikácie (pozri článok B.2.8 "Zostavenie objektového karuselu").

Data_broadcast_id_descriptor identifikuje "hlavnú" zložku vysielaných dát. Podrobnú sémantiku tejto voliteľnej signalizácie poskytuje prenosový protokol. Napríklad v prípade objektového karuselu DVB identifikuje zložku prenášajúcu DSI.

V PMT sa môže nachádzať tiež určitý protokol so špecifickými deskriptormi. Napríklad objektový karusel vyžaduje prítomnosť `carousel_identifier_descriptor` (pozri článok B.2.8 "Zostavenie objektového karuselu").

Vysielanie dát id deskriptora v najjednoduchšom tvare (bez selektora informácií) identifikuje len "hlavnú" zložku. Táto sa môže voliteľne rozšíriť o selektor informácií, ktorý identifikuje typy aplikácií z vysielaných aplikácií s autoštartom (pozri článok 5.3.5.2 "Vysielanie dát id deskriptora").

5.3.3 Poznámky

5.3.3.1 Rezervované

V článku použitý termín **rezervovaný** (angl. **reserved**) definuje kódovaný bitový tok, ktorý vyjadruje, že hodnota sa môže použiť v budúcnosti na rozšírenia definované ISO. Ak nie je v článku uvedené inak, tak všetky bity označené ako "reserved", musia sa nastaviť na „1“.

5.3.3.2 Rezervované na budúce použitie

V článku použitý termín **rezervované na budúce použitie** (angl. **reserved_future_use**) definuje kódovaný bitový tok, ktorého hodnota sa môže použiť v budúcnosti na rozšírenie definované ETSI. Ak nie je v článku uvedené inak, všetky bity označené ako "reserved_future_use" musia sa nastaviť na „1“.

5.3.4 Tabuľka informácií o aplikácii

Tabuľka informácií o aplikácii (AIT) poskytuje úplné informácie o vysielaní dát, požadovanom stave aktivácie aplikácií nachádzajúcich sa v nej atď. AIT pozostáva zo skupiny subtabuliek (pozri článok 5.3.4.5) v rámci vybranej služby, ktorá obsahuje `application_type`, na základe ktorého ich môže prijímač dekódovať.

Vysielateľ môže prostredníctvom dát v AIT zmeniť v prijímači aktivačný stav aplikácie.

5.3.4.1 Chybné dáta

Tabuľky AIT, v ktorých sa nachádzajú chyby, sa musia spracovať takto:

- chyba v deskriptore má za následok zámerné ignorovanie deskriptora; spracovanie tejto slučky deskriptora musí pokračovať nasledujúcim deskriptorom (ak existuje); rozsah detekcie chýb deskriptora sa má obmedziť na sekciu informácií o aplikácii, v ktorej sa chyby prenášajú;
- chyba v slučke aplikácie mimo deskriptora má za následok, že záznam slučky aplikácie je zámerné ignorovaný; spracovanie tejto slučky aplikácie bude pokračovať nasledujúcim záznamom (ak existuje);

POZNÁMKA. – Dôsledkom uvedeného je, že chyba v povinnom deskriptore má za následok jeho zámerné ignorovanie, čo môže potom spôsobiť, že slučka aplikácie takto stratí povahu povinného deskriptora. Preto je táto slučka aplikácie zámerné ignorovaná.

- chyba v sekcii informácií o aplikácii mimo slučky aplikácie má za následok, že celá sekcia informácií o aplikácii je zámerné ignorovaná; spracovanie AIT musí pokračovať nasledujúcou sekciiu informácií o aplikácii (ak existuje).

5.3.4.2 Prenos AIT a monitorovanie

Koncové zariadenia musia sledovať PMT z dôvodu zmien počtu prítomných elementárnych tokov AIT. Od typu aplikácie závisí čas detegovania týchto zmien. Koncové zariadenia musia sledovať všetky elementárne toky AIT v rámci vybranej služby, ktoré sú podrobnejšie uvedené neskôr.

Špecifikácia platformy má definovať minimálnu opakovaciu frekvenciu do každej subtabuľky AIT.

Za predpokladu, že tabuľky AIT na vybranú službu sú poskytované tromi alebo menším počtom elementárnych tokov, potom maximálny časový interval medzi okamihom aktualizácie AIT a okamihom novej verzie detekovanej koncovým zariadením má definovať špecifikácia platformy.

POZNÁMKA. – Ak sa na prenos tabuliek AIT použije vysielanie viac ako s tromi elementárnymi tokmi, potom sa čas odozvy prijímača môže nepredvídateľne zhoršiť.

Koncové zariadenie požaduje monitorovať len typy aplikácií v sekciách AIT, ktoré môže dekódovať. V tomto prípade sa môže signalizácia aplikácie odovzdať len podmnožine typov aplikácií, ktoré sa vysielajú, ak kapacita vysielania pokrýva celkovú kapacitu koncového zariadenia.

Ak subtabuľka AIT, z ktorej boli odstránené ňou signalizované aplikácie, zostáva prítomná v sieti, aplikácie sa musia ukončiť, akoby boli signalizované riadiacim kódom DESTROY.

Ak subtabuľka AIT signalizuje úplné zmiznutie aplikácie zo siete, aplikácia musí naďalej bežať. Koncové zariadenie musí sledovať nové obnovenie subtabuľky AIT definované vznikom uvedených nových subtabuliek AIT, zatiaľ čo táto služba zostáva vybraná.

5.3.4.3 Optimalizovaná signalizácia AIT

Voliteľné `AIT_version_number` obsahuje `application_signalling_descriptor`, ktorý umožňuje optimalizáciu zariadenia prijímača, pretože umožňuje prijímaču získať AIT až potom, keď nastanú zmeny vo verzii AIT uvedené v PMT (pozri článok 5.3.5.1 "Deskriptor signalizácie aplikácie").

5.3.4.4 Viditeľnosť AIT

Pokiaľ sa aplikácia prenáša inde ako v prenosovom toku, v ktorom sa prenáša jej signalizácia bez výberu novej služby, bude naďalej bežať, aj keď AIT nie je viditeľná.

V koncových zariadeniach s viacerými sieťovými rozhraniami, ak je tabuľka AIT vybranej služby viditeľná prostredníctvom ktoréhokoľvek z nich, použije sa bežný spôsob signalizácie tabuľky AIT.

5.3.4.5 Definícia subtabuľky do AIT

Všetky sekcie s rovnakým PID s `table_id` AIT a rovnakou hodnotou `application_type` sú prvkami tej istej subtabuľky.

5.3.4.6 Syntax AIT

Sekcia informácií o aplikácii opisuje aplikácie a ním pridané informácie. Každá sekcia informácií o aplikácii obsahuje jednu "common" (spoločnú) slučku deskriptora na najvyššej úrovni s deskriptormi, ktoré sú spoločné medzi aplikáciami tejto subtabuľky a slučkou aplikácií. Každá aplikácia v slučke aplikácie má slučku deskriptora "application" obsahujúcu deskriptory súvisiace s touto aplikáciou.

Predmetom spoločnej slučky deskriptorov je subtabuľka, podobne ako pri tabuľkách DVB-SI. Všetky deskriptory prítomné v spoločnej slučke deskriptora sa vzťahujú na všetky sekcie

subtabuľky. Typicky, spoločné deskriptory by mali byť bežne prítomné len v sekcii 0 subtabuľky, s výnimkou keď v nej nebude dostatok priestoru.

Ktorékoľvek reťazce obsiahnuté v tabuľkách nesmú mať ukončenie nulami, podobne ako ostatné tabuľky SI DVB.

Tabuľka 16 – Syntax sekcie informácií o aplikácii

	Počet bitov	Identifikátor
application_information_section() {		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
reserved_future_use	1	bslbf
reserved	2	bslbf
section_length	12	uimsbf
test_application_flag	1	bslbf
application_type	15	uimsbf
reserved	2	bslbf
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
reserved_future_use	4	bslbf
common_descriptors_length	12	uimsbf
for(i=0;i<N;i++){		
descriptor()		
}		
reserved_future_use	4	bslbf
application_loop_length	12	uimsbf
for(i=0;i<N;i++){		
application_identifier()		
application_control_code	8	uimsbf
reserved_future_use	4	bslbf
application_descriptors_loop_length	12	uimsbf
for(j=0;j<N;j++){		
descriptor()		
}		
}		
CRC_32	32	rpchof
}		

identifikátor tabuľky (angl. **table_id**): osembitové celé číslo s hodnotou 0 x 74 identifikuje túto tabuľku

syntax identifikátora sekcie (angl. **section_syntax_indicator**): jednobitové pole, ktoré sa musí nastaviť na "1"

dĺžka sekcie (angl. **section_length**): pole sa skladá z dvanástich bitov, prvé dva bity musia byť "00"; zvyšných 10 bitov určuje počet bajtov sekcie počínajúc nasledujúcim poľom section_length, a to vrátane CRC_32; hodnota v tomto poli nesmie prekročiť 1 021 (0 x 3FD)

príznak testovacej aplikácie (angl. **test_application_flag**): jednobitové pole, ak je nastavené, indikuje aplikáciu, ktorá sa prenáša na účely testovania prijímača; testovanie sa nesmie začať alebo zaznamenať v žiadnej API, alebo zobrazit' v žiadnom používateľskom rozhraní prijímačov počas bežných prevádzkových podmienok; spôsob (ak existuje), ktorým prijímač prejde do módu, v ktorom aplikácie signalizované s takýmto nastavením bitu sú spracované, ako keby toto pole bolo nastavené na nulu, je závislý od využitia a typickí koncoví používatelia nemajú mať naň prístup

typ aplikácie (angl. **application_type**): pätnásťbitové pole označuje typ aplikácií opísaných v príslušnej subtabuľke AIT; .pozri článok 5.2.2.2

číslo verzie (angl. **version_number**): päťbitové pole predstavuje číslo verzie sub_table; číslo version_number sa zvýši o 1, keď dochádza k zmene údajov v sub_table; keď dosiahne hodnotu "31", zmení sa na "0"

indikátor aktuálny – nasledujúci (angl. **current_next_indicator**): jednobitový ukazovateľ sa musí nastaviť na "1"

číslo sekcie (angl. **section_number**): osembitové pole označuje číslo sekcie; číslo section_number prvej sekcie v subtabuľke musí byť "0 x 00"; Section_number sa zvýši o 1 s každou ďalšou sekciou s rovnakým table_id a application_type

číslo poslednej sekcie (angl. **last_section_number**): osembitové pole špecifikuje číslo poslednej sekcie, t. j. sekcie s najvyšším číslom section_number, subtabuľky, do ktorej táto sekcia patrí

dĺžka spoločných deskriptorov (angl. **common_descriptors_length**): dvanásťbitové pole nasledujúcich deskriptorov predstavuje celkovú dĺžku v bajtoch; deskriptory v tejto slučke deskriptora sa používajú na všetky aplikácie nachádzajúce sa v tejto subtabuľke AIT

riadiaci kód aplikácie (angl. **application_control_code**): osembitové pole riadi stav aplikácie; sémantika tohto poľa závisí od typu aplikácie; pozri článok 5.2.4 "Riadiace kódy aplikácie"

dĺžka slučky aplikácie (angl. **application_loop_length**): dvanásťbitové pole predstavuje celkovú dĺžku v bajtoch nasledujúcej slučky, ktorá obsahuje informácie o aplikácii

identifikátor aplikácie (angl. **application_identifier()**): štyridsaťosembitové pole identifikuje aplikáciu; štruktúra tohto poľa je definovaná v článku 5.2.3 "Identifikácia aplikácie"

dĺžka slučky deskriptora aplikácie (angl. **application_descriptors_loop_length**): dvanásťbitové pole predstavuje celkovú dĺžku v bajtoch nasledujúcich deskriptorov; deskriptory v tejto slučke sa vzťahujú na konkrétnu aplikáciu

CRC_32: tridsaťdvabitové pole obsahuje hodnotu CRC, ktorá po spracovaní celej sekcie dáva nulový výstup registrov v dekódovači definovanom v prílohe B normy EN 300 468 [1]

5.3.4.7 Používanie neverejných deskriptorov v AIT

Neverejné deskriptory sa môžu začleniť do AIT za predpokladu, že sú v predmete DVB-SI EN 300 468 [1] špecifikátora deskriptora neverejných dát. Rozsah pôsobnosti pravidiel špecifikátora deskriptora neverejných dát je takýto:

- ak sa tento deskriptor nachádza v niektorej slučke deskriptora AIT, potom každý identifikovaný špecifikátor v tejto slučke deskriptora sa vzťahuje na všetky nasledujúce deskriptory a používateľom definované hodnoty v konkrétnej slučke deskriptora až po koniec slučky deskriptora, alebo až do ďalšieho výskytu private_data_specifier_descriptor;
- použitie deskriptora v spoločnej (prvej) slučke deskriptora sa nevzťahuje na deskriptory alebo používateľom definované hodnoty v (druhej) slučke deskriptora aplikácie.

5.3.4.8 Kódovanie textu v AIT

Pokiaľ nie je uvedené inak, všetky polia interpretované ako textové reťazce v AIT sa musia kódovať v UTF8, ale nesmú obsahovať nulový znak.

5.3.4.9 Prístup na AIT vo formáte MPEG-2 prostredníctvom širokopásmového pripojenia

Súbor AIT obsahuje kódovanie MPEG-2 AIT v tvare, ktorý sa môže načítať cez HTTP a slúži na zoskupenie aplikácií, ktoré nesúvisia s vysielacou službou.

POZNÁMKA. – Špecifikácie platformy, ktoré obsahujú podporu súboru AIT, majú definovať, ako sa táto podpora používa, napríklad prechodom HTTP URL, ktorý odkazuje na tento súbor volaním na platformou definované API.

V špecifikácii platformy sa majú definovať všetky požiadavky na monitorovanie alebo dotazovanie na zmeny súboru AIT.

5.3.4.9.1 Syntax

Kódovanie interaktívneho kanála AIT do súboru AIT je nasledovné:

- jeden súbor musí obsahovať všetky údaje;
- súbor musí obsahovať zreteľné sekcie informácií o aplikácii, uvedené v článku 5.3 tohto dokumentu;
- v prípade viacerých sekcií sa tieto musia usporiadať takto:
 - vzostupné poradie `application_type`;
 - v rámci jednej hodnoty `application_type` vo vzostupnom poradí `section_number`;
- všetky sekcie musia mať `current_next_indicator` nastavený na "1";
- vhodné sú len riadiace kódy aplikácie AUTOSTART a PRESENT (pozri tabuľku 3).

5.3.4.9.2 Syntaktické obmedzenia

5.3.4.9.2.1 Prenosové protokoly

Povolený je len `protocol_id`, ktorý má hodnotu 0 x 0003 (pozri tabuľku 29 "Protocol_id").

5.3.4.9.3 Typ dát MIME

V súbore AIT musí byť typ MIME "application/vnd.dvb.ait". Prípona súboru musí byť ".ait". V rámci implementácie sa v súbore AIT môže použiť aj typ MIME na spätnú kompatibilitu "application/dvb.ai". Použitie tohto typu MIME sa neodporúča v nových aplikáciách a využití služieb.

5.3.5 Generické deskripty

5.3.5.1 Deskriptor signalizácie aplikácie

Deskriptor `Application_signalling_descriptor` je definovaný na použitie v slučke elementárneho toku PMT, kde `stream_type` elementárneho toku je 0 x 05. To znamená, že elementárny tok prenáša tabuľku informácií o aplikácii.

Deskriptor `Application_signalling_descriptor` voliteľne obsahuje slučku `application_type` a páry `AIT_version_number`. Tie umožňujú deskriptoru voliteľne reprodukovať aktuálny stav čísla verzie súvisiacej tabuľky informácií o aplikácii. To umožňuje, aby prijímač bol informovaný o verzii AIT,

ako vedľajší účinok sledovania PMT, ktorá sa má detailne sledovať počas bežných podmienok (pozri článok 5.3.4.3 "Optimalizovaná signalizácia AIT").

Keď prijímač zistí zmenu obsahu `application_signalling_descriptor`, tak musí získať novú verziu AIT a zodpovedajúcim spôsobom reagovať.

Prítomnosť subpolí `application_type` a `AIT_version_number` je voliteľná. Ak nie sú prítomné, potom platia na prenos a monitorovanie AIT, podmienky podľa článku 5.3.4.2 "Prenos AIT a monitorovanie".

Tabuľka 17 – Syntax deskriptora signalizácie aplikácie

	Počet bitov	Identifikátor
<code>application_signalling_descriptor() {</code>		
<code>descriptor_tag</code>	8	uimsbf
<code>descriptor_length</code>	8	uimsbf
<code>for(i=0; i<N; i++){</code>		
<code>reserved_future_use</code>	1	
<code>application_type</code>	15	uimsbf
<code>reserved_future_use</code>	3	bslbf
<code>AIT_version_number</code>	5	uimsbf
<code>}</code>		
<code>}</code>		

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 6F identifikuje tento deskriptor

dĺžka deskriptora (angl. **descriptor_length**): osembitové pole špecifikuje celkový počet bajtov dátovej časti deskriptora nasledujúcich za bajtmi, ktoré definujú hodnotu tohto poľa

typ aplikácie (angl. **application_type**): pätnásťbitové pole identifikuje typ aplikácie zo subtabuľky informácií o aplikácii, ktorá je v tomto elementárnom toku

číslo verzie AIT (angl. **AIT_version_number**): päťbitové pole poskytuje "aktuálne" číslo verzie subtabuľky informácií o aplikácii identifikovanej poľom `application_type`

5.3.5.2 Deskriptor identifikátora id vysielania dát

Deskriptor `data_broadcast_id_descriptor` je definovaný na používanie informácií PMT v elementárnom toku. Deskriptor identifikuje:

- formát prenosu vysielania dát elementárneho toku, ktorého je "hlavnou zložkou";
- sémantika "hlavnej zložky" je špecifický prenosový protokol;
- skupinu typov aplikácií do všetkých aplikácií s autoštartom poskytovaných vysielaním dát.

Jeden elementárny tok môže mať jeden a viac `data_broadcast_id_descriptor` na indikáciu zhody s viacerými špecifikáciami vysielaných dát. Okrem toho, sa môže použiť aj viac `data_broadcast_id_descriptor` na vyhľadanie ďalších typov aplikácií v rámci vysielania dát určitého id.

Deskriptor `data_broadcast_id_descriptor` môže indikovať aj viac elementárnych tokov, pretože aplikácie s autoštartom sú prenášané viacerými doručovacími mechanizmami, napríklad jedna služba môže mať aj viac objektových karuselov na poskytovanie aplikácií s autoštartom.

5.3.5.2.1 Generické deskripty

Deskriptor `data_broadcast_id_descriptor` je všeobecne definovaný podľa EN 300 468 [1] (znázornený v tabuľke 18). Tam, kde nie je k dispozícii "konkrétny údaj id", deskriptor identifikuje len "hlavnú" zložku vysielať dát.

Tabuľka 18 – Syntax generického deskriptora vysielať dát

	Počet bitov	Identifikátor	Hodnota
<code>data_broadcast_id_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	0x66
<code>descriptor_length</code>	8	uimsbf	
<code>data_broadcast_id</code>	16	uimsbf	
<code>for (i=0; i<N; i++) {</code>			
<code>id specific data</code>	8	bslbf	
<code>}</code>			
<code>}</code>			

5.3.5.2.2 Identifikátor id deskriptora vysielať dát do interaktívnej aplikácie

Ak identifikátor `data_broadcast_id` je 0 x 00F0 alebo 0 x 00F1 (pozri tabuľku 38), potom syntax deskriptora `data_broadcast_id_descriptor` je uvedená v tabuľke 19. Tá rozširuje generický deskriptor s voliteľným zoznamom typov aplikácií s autoštartom v rámci vysielať dát. Tento zoznam obsahuje pokyn, ktorý umožní koncovému zariadeniu uprednostniť pripojenie vysielať dát, ak sú niektoré poskytované službou. Ak nie je k dispozícii žiadny zoznam, potom `data_broadcast_id_descriptor` zámerne neposkytne typy aplikácií s autoštartom, ktoré sa môžu preniesť prostredníctvom vysielať dát. Ak zoznam aplikácie nie je prázdny, potom vysielať dát nesmie obsahovať aplikácie s autoštartom iných typov aplikácií, ako sú uvedené v zozname. Nepožaduje sa, aby vysielať dát vždy obsahovalo všetky typy aplikácií s autoštartom uvedené v zozname.

Tabuľka 19 – Syntax deskriptora `data_broadcast_id_descriptor` do interaktívnych aplikácií

	Počet bitov	Identifikátor	Hodnota
<code>data_broadcast_id_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	0x66
<code>descriptor_length</code>	8	uimsbf	
<code>data_broadcast_id</code>	16	uimsbf	
<code>for (i=0; i<N; i++) {</code>			
<code>reserved_future_use</code>	1		
<code>application_type</code>	15	uimsbf	
<code>}</code>			
<code>}</code>			

príznak deskriptora (angl. `descriptor_tag`): osembitové celé číslo s hodnotou 0 x 66 identifikuje tento deskriptor

identifikátor vysielať dát (angl. `data_broadcast_id`): šesťnásťbitové pole identifikuje formát prenosového protokolu vysielať dát; tieto hodnoty sú uvedené na <http://www.dvb.org>

typ aplikácie (angl. `application_type`): pätnásťbitové pole identifikuje typ aplikácie; pozri článok 5.2.2 tohto dokumentu

5.3.5.3 Deskriptor aplikácie

V každej vnútornej slučke deskriptora AIT *application* sa môže deskriptor *application_descriptor* nachádzať len raz.

Tabuľka 20 – Syntax deskriptora aplikácie

	Počet bitov	Identifikátor	Hodnota
<code>application_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	0x00
<code>descriptor_length</code>	8	uimsbf	
<code>application_profiles_length</code>	8	uimsbf	
<code>for(i=0; i<N; i++) {</code>			
<code>application_profile</code>	16	uimsbf	
<code>version.major</code>	8	uimsbf	
<code>version.minor</code>	8	uimsbf	
<code>version.micro</code>	8	uimsbf	
<code>}</code>			
<code>service_bound_flag</code>	1	bslbf	
<code>visibility</code>	2	bslbf	
<code>reserved_future_use</code>	5	bslbf	
<code>application_priority</code>	8	uimsbf	
<code>for(i=0; i<N; i++) {</code>			
<code>transport_protocol_label</code>	8	uimsbf	
<code>}</code>			
<code>}</code>			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 00 identifikuje každý deskriptor

dĺžka profilov aplikácie (angl. **application_profiles_length**): osembitové pole indikuje dĺžku slučky *application_profile* v bajtoch

profil aplikácie (angl. **application_profile**): šestnásťbitové pole identifikuje, ktorý špecifický typ profilu aplikácie je požadovaný aplikáciou; pozri článok 5.2.5 "Profily platformy"

hlavná verzia (angl. **version.major**): osembitové pole indikuje číslo hlavnej verzie profilu; pozri článok 5.2.5 "Profily platformy"

vedľajšia verzia (angl. **version.minor**): osembitové pole indikuje číslo vedľajšej verzie profilu; pozri článok 5.2.5 "Profily platformy".

mikroverzia (angl. **version.micro**): osembitové pole indikuje číslo mikroverzie profilu; pozri článok 5.2.5 "Profily platformy"

návesť súvisiacej služby (angl. **service_bound_flag**): ak je tento jednobitový príznak nastavený na "1", aplikácia je pridružená len k aktuálnej službe a tak proces zrušenia aplikácie musí začať spustením zmeny služby od začiatku bez ohľadu na obsah cieľovej AIT

viditeľnosť (angl. **visibility**): dvojbitové pole označuje, či aplikácia je viditeľná vzhľadom na ostatné aplikácie prostredníctvom zoznamu aplikácie API (ak to platforma podporuje) alebo pre používateľov; pozri článok 5.2.6 "Viditeľnosť aplikácie"

priorita aplikácie (angl. **application_priority**): osembitové pole určuje dôležitosť aplikácie vzhľadom na ostatné signalizované aplikácie; pozri článok 5.2.7 "Priorita aplikácie"

identifikátor prenosového protokolu (angl. **transport_protocol_label**): osembitové pole identifikuje prenosový protokol, ktorý poskytuje aplikáciu; pozri transport_protocol_label v článku 5.3.6 "Deskriptory prenosového protokolu"

Ak je signalizovaných viac protokolov, potom každý protokol používa iný spôsob doručovacieho mechanizmu. Poradie sa určuje z pohľadu vysielateľa, ktorého prenosové spojenie poskytuje najlepší zážitok používateľovi (prvé je najlepšie). Toto sa môže použiť ako návod poskytnutý koncovému zariadeniu na implementáciu. Hodnotí sa len raz počas životného cyklu aplikácie.

Výber protokolu koncovým zariadením závisí od rôznych faktorov, vrátane preferencií používateľa a výkonnosti spojenia, na ktoré je pripojené koncové zariadenie.

5.3.5.4 Deskriptor záznamu aplikácie

Deskriptor application_recording_descriptor sa môže signalizovať v slučke deskriptora aplikácie AIT. Tento deskriptor obsahuje ďalšie informácie o životnom cykle aplikácie najmä s uvedením, či je aplikácia vhodná na použitie prehrávania s trikovým módom. Ten signalizuje, či sa aplikácia má alebo nemá zaznamenať, keď program, v priebehu ktorého je táto aplikácia signalizovaná, sa zaznamená. To znamená určiť miesto dátových prostriedkov, ktoré sa zaznamenajú spolu s aplikáciou, ako aj návesti modulov aplikácií objektových karuselov, ktoré sa musia, majú sa alebo sa nemajú zaznamenať.

Tabuľka 21 – Syntax deskriptora záznamu aplikácie

Syntax	Počet bitov	Identifikátor	Komentáre/hodnota
application_recording_descriptor (){			
descriptor_tag	8	uimsbf	0x06
descriptor_length	8	uimsbf	
scheduled_recording_flag	1	bslbf	
trick_mode_aware_flag	1	bslbf	
time_shift_flag	1	bslbf	
dynamic_flag	1	bslbf	
av_synced_flag	1	bslbf	
initiating_replay_flag	1	bslbf	
reserved	2	bslbf	
label_count	8	uimsbf	N0
for(i=0;i<N0;i++){			
label_length	8	uimsbf	N1
for(j=0;j<N1;j++){			
label_char	8	uimsbf	
}			
storage_properties	2	uimsbf	
reserved	6		
}			
component_tag_list_length	8	uimsbf	N2
for(i=0;i<N2;i++){			
component_tag	8	uimsbf	
}			
private_length	8	uimsbf	N3
for(i=0;i<N3;i++){			
private	8	uimsbf	
}			
for(i=0;i<N4;i++){			
reserved_future_use	8	uimsbf	
}			
}			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 06 identifikuje tento deskriptor

návesť časového plánu záznamu (angl. **scheduled_recording_flag**): jednobitový príznak nastavený na "1" signalizuje aplikáciu vhodnú na záznam, kedy služba, v ktorej je to signalizované, sa zaznamenáva podľa časového plánu záznamu; ak je príznak nastavený na "0", znamená to, že aplikácia nie je vhodná na záznam pomocou časového plánu záznamu; príklady aplikácií nevhodných na záznam: aplikácia neoverená v prostredí PVR, alebo aplikácia priamo súvisiaca s časom prenosu a pre koncového používateľa nemá význam ju prehrávať zo záznamu; napríklad aplikácia viazaná na živú udalosť

návesť informácie o trikovom móde (angl. **trick_mode_aware_flag**): jednobitový príznak nastavený na "1" signalizuje aplikáciu, ktorá má informáciu o trikovom móde; ak je príznak nastavený na "0", aplikácia nemá informáciu o trikovom móde

návesť časového posunu (angl. **time_shift_flag**): jednobitový príznak nastavený na "1" signalizuje aplikáciu vhodnú na záznam; ak je príznak signalizovaný v službe, tá je zaznamenaná v móde záznam s časovým posunom; ak je nastavený na "0", znamená to, že aplikácia nie je vhodná na záznam s časovým posunom

návesť dynamiky (angl. **dynamic_flag**): jednobitový príznak signalizuje, či aplikácia závisí od používania dynamických dát vo vysielaní počas jej trvania; ak je príznak nastavený na "1", znamená to, že aplikácia závisí od prítomnosti súborov (kódu alebo dát) alebo signalizácie aplikácie (napríklad riadiaci kód aplikácie), ktorá zmení v priebehu životnosti časť obsahu; ak je nastavená na "0", znamená to, že aplikácia nezávisí od vysielania dynamických dát

POZNÁMKA. – Dokument nedefinuje správanie koncových zariadení, ktoré je podmienené hodnotou tohto príznaku. Špecifikácie platformy môžu použiť tento príznak na označenie, či je alebo nie je možné aplikáciu zaznamenať.

návesť synchronizácie av (angl. **av_synced_flag**): jednobitový príznak signalizuje aplikáciu, ktorá môže požadovať tok udalostí; ak je nastavený na "1", tento tok sa požaduje

POZNÁMKA. – Dokument nedefinuje správanie koncových zariadení, ktoré je podmienené hodnotou tohto príznaku. Špecifikácie platformy môžu použiť tento príznak na označenie, či je alebo nie je možné aplikáciu zaznamenať.

návesť inicializácie prehrávania (angl. **initiating_replay_flag**): jednobitový príznak nastavený na "1" signalizuje, že koncové zariadenie nesmie začať prehrávanie tokov nachádzajúcich sa v rovnakom zázname ako aplikácia; aplikácia zodpovedá za spustenie tohto prehrávania; ak je príznak nastavený na "0", prehrávanie sa musí spustiť súbežne so spustením aplikácie ako v bežnom prípade; s týmto príznakom sa počíta len pri prvom spustení prehrávania záznamu, po uplynutí tohto času sa hodnota príznaku ignoruje

počet návěstí (angl. **label_count**): osembitové pole určuje počet používaných návěstí

dĺžka návěstí (angl. **label_count**): osembitové pole určuje počet bajtov návěstí

vlastnosti návěstí (angl. **label_char**): osembitové pole určuje usporiadanie bajtov časti návěstí aplikácie v prenosovom protokole

POZNÁMKA. – Dokument nedefinuje, ktoré časti aplikácií sa môžu označiť návěstou alebo formou návěstí (ak existuje). Špecifikácie platformy, ktoré chcú používať tento mechanizmus, musia definovať formát návěstí.

vlastnosti uloženia (angl. **storage_properties**): pole signalizuje dôležitosť uloženia označenej časti aplikácie; hodnoty tohto poľa sú definované v tabuľke 22

Tabuľka 22 – Hodnoty poľa storage_properties

Hodnota storage_properties	Opis
0	nemajú sa uložiť
1	rozhodujúce na uloženie
2	voliteľné na uloženie
3	rezervované

príznak dĺžky zložky zoznamu (angl. **component_tag_list_length**): celé číslo špecifikuje príznak dĺžky zložky zoznamu v bajtoch

príznak zložky (angl. **component_tag**): pole určuje zložku služby poskytnutú dátami, ktoré požaduje aplikácia počas prehrávania, a ktoré sa musia zaznamenať spolu s aplikáciou a tokmi zvuku, tokmi obrazu a titulkami

neverejné (angl. **private**): bajty sa môžu použiť na neverejné rozšírenie

rezervované na budúce používanie (angl. **reserved_future_use**): vyhradené bajty sa môžu použiť na budúce rozšírenie DVB

5.3.5.5 Deskriptor používania aplikácie

Deskriptor `application_usage_descriptor` signalizuje aplikáciu, ktorá poskytuje konkrétnu, dobre známu službu, napríklad teletext, EPG alebo čítanie. Koncové zariadenia môžu obsahovať voliteľné klávesy na spustenie týchto služieb, napríklad tlačidlo diaľkového ovládača. Koncové zariadenia môžu obsahovať tiež natívne UI ponúkajúce prístup k týmto službám. Ak `application_usage_descriptor` nie je prítomný, potom aplikácia neposkytuje konkrétnu dobre známu službu.

Tabuľka 23 – Deskriptor používania aplikácie

	Počet bitov	Identifikátor	Hodnota
<code>application_usage_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	0x16
<code>descriptor_length</code>	8	uimsbf	
<code>usage_type</code>	8	uimsbf	
<code>}</code>			

príznak deskriptora (angl. **descriptor_tag**): osembitové pole s hodnotou 0 x 16 identifikuje tento deskriptor

typ použitia (angl. **usage_type**): osembitové pole označuje službu, ktorú poskytuje aplikácia; pole sa musí označovať podľa tabuľky 11

5.3.5.6 Deskriptory informácií pre používateľa

Deskriptory informácií pre používateľa dopĺňujú `application_descriptor` tým, že poskytujú informácie vhodné na prezentáciu používateľovi. Deskriptor `application_descriptor` poskytuje technické informácie na automatické používanie v prijímači.

Tieto deskriptory sú definované na používanie vo vnútornej `application` slučke deskriptora AIT.

5.3.5.6.1 Deskriptor názvu aplikácie

V každej vnútornej slučke deskriptora *application* AIT sa môže *application_descriptor* nachádzať len raz. Názov aplikácie musí rozpoznať aplikáciu a musí poskytovať informáciu pre používateľa.

Tabuľka 24 – Syntax deskriptora názvu aplikácie

	Počet bitov	Identifikátor	Hodnota
<code>application_name_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	0x01
<code>descriptor_length</code>	8	uimsbf	
<code>for (i=0; i<N; i++) {</code>			
<code>ISO_639_language_code</code>	24	bslbf	
<code>application_name_length</code>	8	uimsbf	
<code>for (i=0; i<N; i++) {</code>			
<code>application_name_char</code>	8	uimsbf	
<code>}</code>			
<code>}</code>			
<code>}</code>			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 01 identifikuje tento deskriptor

kód jazyka ISO 639 (angl. **ISO_639_language_code**): dvadsaťštyribitové pole obsahuje v názve aplikácie trojpísmenový kód v jazyku ISO 639-2 [7]; používať sa môžu obidve normy ISO 639.2/B a ISO 639.2/T

Každý znak je kódovaný do ôsmich bitov podľa normy ISO 8859-1 [8] a vkladá sa do postupnosti dvadsaťštyribitového poľa.

dĺžka názvu aplikácie (angl. **application_name_length**): osembitové celé číslo bez znamienka určuje počet bajtov v názve aplikácie

charakteristika názvu aplikácie (angl. **application_name_char**): pole obsahuje jeden znak reťazca znakov (nie je ukončený nulami) kódovaný podľa prílohy A normy EN 300 468 [1]; takto určený názov reťazca aplikácie je informatívny pre používateľa; špecifické typy aplikácií môžu stanoviť ďalšie obmedzenia na kódovanie tejto hodnoty

5.3.5.6.2 Deskriptor ikon aplikácie

V každej vnútornej slučke deskriptora *application* AIT sa môže tento deskriptor nachádzať len raz alebo sa nemusí nachádzať ani raz; z tohto dôvodu sú ikony pridružené k aplikácii.

Tabuľka 25 – Syntax deskriptora ikon aplikácie

	Počet bitov	Identifikátor	Hodnota
application_icons_descriptor() {			
descriptor_tag	8	uimsbf	0x0B
descriptor_length	8	uimsbf	
icon_locator_length	8	uimsbf	
for (i=0; i<N; i++) {			
icon_locator_byte	8	uimsbf	
}			
icon_flags	16	bslbf	
for (i=0; i<N; i++) {			
reserved_future_use	8	bslbf	
}			
}			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 0B identifikuje tento deskriptor

dĺžka lokátora ikony (angl. **icon_locator_length**): osembitové celé číslo špecifikuje počet bajtov v reťazci predvolby štandardného názvu súboru ikony

bajt lokátora ikony (angl. **icon_locator_byte**): osembitová hodnota predstavuje jednobajtový reťazec lokátora ikony; pozri článok 5.2.8 "Ikony aplikácie"

príznak ikony (angl. **icon flags**): šestnásťbitové pole udáva veľkosť a pomer strán dostupných ikon na danú aplikáciu; pozri článok 5.2.8 "Ikony aplikácie"

5.3.5.7 Deskriptor autorizácie externej aplikácie

Prvá slučka deskriptora tabuľky informácií o aplikácii *common* môže obsahovať viac deskriptorov `external_application_authorization_descriptor` alebo nemusí obsahovať žiadny deskriptor. Každý deskriptor obsahuje informácie o externých aplikáciách, ktoré dovoľujú pokračovať v priebehu s aplikáciami uvedenými v subtabuľke tabuľky informácií o aplikácii, ale nemôžu sa spustiť z tejto služby. Externá autorizácia sa vzťahuje na aplikácie s identifikovanými `application_identifier()`, ktoré sú v `application_type` identifikované subtabuľkou AIT, kde sa nachádza tento deskriptor.

Tabuľka 26 – Syntax deskriptora autorizácie externej aplikácie

	Počet bitov	Identifikátor	Hodnota
external_application_authorisation_descriptor() {			
descriptor_tag	8	uimsbf	0x05
descriptor_length	8	uimsbf	
for(i=0; i<N; i++) {			
application_identifier()			
application_priority	8	uimsbf	
}			
}			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 05 identifikuje tento deskriptor

identifikátor aplikácie() [angl. **application_identifier()**] štyridsaťosembitové pole identifikuje aplikáciu; štruktúra tohto poľa je definovaná v článku 5.2.3 "Identifikácia aplikácie"

priorita aplikácie (angl. **application_priority**): osembitové celé číslo určuje prioritu aplikácie, ktorá predpokladá súvislosť s aktuálnou službou

Ak sa používajú zástupné znaky 0 x ffff alebo 0 x fffe v `application_id` v `application_identifier()` a existujú aplikácie z rovnakým `organisation_id`, ktorý je explicitne signalizovaný v slučke aplikácie AIT, potom sa prioritou týchto aplikácií musí signalizovať v `application_descriptor` (pozri článok 5.3.5.3 a `application_priority` v článku 5.3.5.3 "Deskriptor aplikácie").

5.3.5.8 Deskriptor obmedzenia grafiky

Deskriptor obmedzenia grafiky definuje situácie, počas ktorých môže aplikácia pracovať alebo sa môže testovať počas činnosti. Tieto situácie sú:

- keď aplikácia podporuje grafické rozlíšenie na celú obrazovku;
- fungujú s aplikáciou umožnenou v prípade riadeného videa, napríklad zmenšená veľkosť obrazu, nie celá obrazovka; v inej aplikácii nie je signalizovaná ako súčasť aktuálnej služby, napríklad EPG, navigátor, alebo nesúvisiaca aplikácia, ktorá beží v rámci abstraktnej služby.

Tento deskriptor môže byť prítomný vo vnútornej slučke `application` AIT, v tomto prípade to platí len v tejto aplikácii, alebo vo vonkajšej slučke `common` AIT. V takomto prípade to platí na všetky signalizované aplikácie v subtabuľke AIT.

Tabuľka 27 – Syntax deskriptora obmedzenia grafiky

	Počet bitov	Identifikátor	Hodnota
<code>graphics_constraints_descriptor() {</code>			
<code>descriptor_tag</code>	8	uimsbf	0x14
<code>descriptor_length</code>	8	uimsbf	
<code>reserved_future_use</code>	5	bslbf	
<code>can_run_without_visible_ui</code>	1	bslbf	
<code>handles_configuration_changed</code>	1	bslbf	
<code>handles_externally_controlled_video</code>	1	bslbf	
<code>for(i=0;i<N;i++) {</code>			
<code>graphics_configuration_byte</code>	8	uimsbf	
<code>}</code>			
<code>}</code>			

Polia majú nasledujúce významy:

príznak deskriptora (angl. `descriptor_tag`): osembitové celé číslo s hodnotou 0 x 14 identifikuje tento deskriptor

dĺžka deskriptora (angl. `descriptor_length`): osembitové pole určuje celkový počet bajtov dátovej časti deskriptora nasledujúcich za bajtmi, ktoré definujú hodnotu poľa

možnosť priebehu bez viditeľného ui (angl. `can_run_without_visible_ui`): jednobitový príznak signalizuje aplikáciu, ktorá môže bežať bez viditeľného UI; pozri článok 5.2.9.1.2 "Spustenie bez viditeľného UI"

zmena konfigurácie grafiky (angl. `handles_configuration_changed`): jednobitový príznak signalizuje aplikáciu, ktorá môže podporovať zmeny v konfigurácii grafiky koncového zariadenia; pozri článok 5.2.9.1.3 "Nastavenie zmeny konfigurácií grafiky"

externé nastavenie riadenia videa (angl. `handles_externally_controlled_video`): jednobitový príznak signalizuje aplikáciu s možnosťou ovládania prezentácie videa v tej istej službe; pozri článok 5.2.9.1.4 "Nastavenie externe ovládaného videa"

bajt konfigurácie grafiky (angl. **graphics_configuration_byte**): osembitové polia obsahujú hodnotu uvedenú v článku 5.2.9.1.1 "Podporované konfigurácie grafiky"

5.3.6 Deskriptory prenosového protokolu

Transport_protocol_descriptor identifikuje prenosový protokol priradený zložke služby a prípadne poskytuje informácie závislé od protokolu.

Deskriptor sa môže použiť vo vonkajšej slučke deskriptora *common* alebo vo vnútornej slučke deskriptora *application*. Ak sa použije v slučke *common*, platí to na všetky aplikácie v subtabuľke. Všetky ostatné deskriptory v slučke *application* opísané ďalšími prenosovými protokolmi sú k dispozícii konkrétnej aplikácii.

Každá aplikácia musí obsahovať aspoň jeden transport_protocol_descriptor.

Tabuľka 28 – Syntax deskriptora prenosového protokolu

	Počet bitov	Identifikátor	Hodnota
transport_protocol_descriptor() {			
descriptor_tag	8	uimsbf	0x02
descriptor_length	8	uimsbf	
protocol_id	16	uimsbf	
transport_protocol_label	8	uimsbf	
for(i=0; i<N; i++) {			
selector_byte	8	uimsbf	N1
}			
}			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 02 identifikuje príslušný deskriptor

identifikátor protokolu (angl. **protocol_id**): identifikátor protokolu sa používa na prenos aplikácií; hodnoty protocol_id sú uvedené v tomto dokumente a na <http://www.dvb.org>.

Tabuľka 29 – Identifikátor protokolu protocol_id

protocol_id	Opis
0x0000	reserved_future_use
0x0001	Object Carousel as defined in annex B of the present document.
0x0002	reserved
0x0003	Transport via HTTP over the interaction channel as defined in clause 7.2.
0x0004 to 0x00FF	Reserved for use by DVB
0x0100 to 0xFFFF	Subject to registration at http://www.dvb.org .

návesť prenosového protokolu (angl. **transport_protocol_label**): osembitové pole jednoznačne identifikuje prenosový protokol v danej sekcii AIT; application_descriptor odkazuje na hodnotu, ktorá určuje prenosové spojenie, ktorým sa prenáša aplikácia

selektor bajtov (angl. **selector_byte**): dodatkový protokol špecifickej informácie

Tabuľka 30 – Sémantika selektora bajtov

protocol_id	Selektor dátových bajtov
0x0000	reserved_future_use
0x0001	See clause 5.3.6.1, "Syntax of selector bytes for OC transport".
0x0002	reserved
0x0003	See clause 5.3.6.2, "Syntax of selector bytes for interaction channel transport"
0x0004 to 0xFFFF	Not defined in this version of the present document

5.3.6.1 Syntax selektora bajtov na prenos OC

Ak identifikátor protokolu je 0 x 0001, potom sa selektor bajtov v transport_protocol_descriptor musí uviesť podľa tabuľky 31.

Tabuľka 31 – Syntax selektora bajtov na prenos OC

Syntax	Bitsy	Identifikátor
remote_connection	1	bslbf
reserved_future_use	7	bslbf
if(remote_connection == "1"){		
original_network_id	16	uimbsf
transport_stream_id	16	uimbsf
service_id	16	uimbsf
}		
component_tag	8	uimbsf

diaľkové spojenie (angl. **remote_connection**): jednobitový príznak nastavený na "1" signalizuje, že prenosové spojenie je poskytnuté vysielacou službou, ktorá je odlišná od prenášanej AIT; takéto aplikácie nesmú prijímače automaticky spustiť, ale sú viditeľné (podmienka viditeľnosti poľa deskriptora aplikácie a dostupnosti API na zistenie signalizovaných aplikácií) na možné spustenie výberom služby (ale nie prostredníctvom aplikácie spustenej API); ak je tento bit nastavený, potom sa nasledujúce 3 polia (original_network_id, transport_stream_id a service_id) nachádzajú v selektore bajtov; v prípade poskytnutia prenosového spojenia aktuálnou službou sa tento príznak musí nastaviť na "0"

Aplikácie s takto nastaveným príznakom musia mať riadiaci kód aplikácie nastavený na REMOTE (pozri tabuľku 3) alebo musia mať application_storage_descriptor s "launchable_completely_from_cache" nastavené na "1" (pozri článok 5.2.11.1.3).

Aplikácie, ktoré majú remote_connection nastavené na "1" a tiež application_storage_descriptor s "launchable_completely_from_cache" nastavené na "1" (pozri článok 5.2.11.1.3), predstavujú zvláštny prípad. Ak je takáto aplikácia kešovaná koncovým zariadením, môže sa spustiť obvyklým spôsobom. Na aplikáciu signalizovanú týmto spôsobom neexistujú žiadne zvláštne obmedzenia riadiaceho kódu, napríklad môže to byť PRESENT alebo dokonca AUTOŠTART. Ak aplikácia nie je kešovaná koncovým zariadením, nemôže sa spustiť a signalizovaný riadiaci kód sa bude ignorovať. Aplikácia bude vždy spracovaná tak, akoby sa použil REMOTE.

Diaľkové aplikácie sa môžu kešovať a môžu sa uložiť obvyklým spôsobom, keď aplikácia najprv nalaď sieťové rozhranie na príslušný prenosový tok.

identifikátor východiskovej siete (angl. **original_network_id**): šestnásťbitové pole označuje identifikátor východiskovej siete prenosového toku DVB SI, ktorý poskytuje prenosové spojenie

identifikátor prenosového toku (angl. **transport_stream_id**): šestnásťbitové pole určuje identifikátor prenosového toku MPEG v rámci prenosového toku, ktorý poskytuje prenosové spojenie

identifikátor služby (angl. **service_id**): šestnásťbitové pole identifikuje identifikátor služby DVB-SI v rámci služby, ktorá poskytuje prenosové spojenie

príznak zložky (angl. **component_tag**): identifikuje hlavnú zložku služby, ktorá poskytuje aplikáciu; identifikovaná zložka predstavuje elementárny tok, ktorý prenáša DSI objektového karuselu

5.3.6.2 Syntax selektora bajtov na prenos interaktívnym kanálom

Ak má identifikátor protokolu hodnotu 0 x 0003, potom sa selektor bajtov v deskriptore `transport_protocol_descriptor` musí uviesť podľa tabuľky 32 "Syntax selektora bajtov na prenos interaktivity". To umožňuje kódovanie niekoľkých adries URL. Deskriptor sa môže použiť aj v zjednodušenej forme, kde je kódovaná iba jedna adresa URL.

Účinné kódovanie umožňuje rozdeliť veľa podobných adries URL na adresu spoločnej základnej časti URL a na skupinu rozšírení URL. Skupina URL môže identifikovať súbory ZIP [16] alebo základné adresy URL končiacie lomkou "/", ktorá zapuzdruje časti systémového súboru.

Viaceré deskriptory prenosového protokolu s hodnotou protokolu ID 0 x 0003 a rovnakou návěstou prenosového protokolu umožňujú definovať väčšiu skupinu adries URL na opis systémového súboru.

Tabuľka 32 – Syntax selektora bajtov na prenos interaktivity

Syntax	Bity	Identifikátor
<code>for(i=0; i<N; i++){</code>		
URL_base_length	8	uimsbf
for(j=0; j<N; j++){		
URL_base_byte	8	uimsbf
}		
URL_extension_count	8	uimsbf
for(j=0; j<URL_extension_count; j++){		
URL_extension_length	8	uimsbf
for(k=0; k<URL_length; k++){		
URL_extension_byte	8	uimsbf
}		
}		
}		

dĺžka základnej URL (angl. **URL_base_length**): osembitové pole určuje počet bajtov základnej časti URL

bajt základnej URL (angl. **URL_base_byte**): bajty tvoria prvú časť adresy HTTP URL zodpovedajúcej HTTP 1.0 (pozri RFC 1945 [18]) alebo prvú časť HTTPS URL podľa RFC 2818 [19], alebo prvú časť inej adresy URL podľa RFC 3986 [14]

počet rozšírení URL (angl. **URL_extension_count**): osembitové pole určuje počet adries rozšírenia URL sprostredkovaných týmto deskriptorom

dĺžka rozšírenia URL (angl. **URL_extension_length**): osembitové pole určuje počet bajtov rozšírenej časti adresy URL

bajt rozšírenia URL (angl. **URL_extension_byte**): tieto bajty tvoria druhú časť HTTP URL podľa HTTP 1.0 (pozri RFC 1945 [18]) alebo druhú časť HTTPS URL podľa RFC 2818 [19], alebo inú URL, ktorých schéma je podporovaná registrovaným prenosovým interaktívnym kanálom implementovaným poskytovateľom služby

URL sú tvorené spájaním rozšírenej URL s predchádzajúcou základnou URL. URL tak tvorí určený adresár systémového súboru alebo konkrétny súbor ZIP.

V zjednodušenej podobe platí nasledujúce:

- kódovať sa musí práve jedna základná URL;
- URL tvorená URL_base_byte musí končiť znakom lomky ("/"); odkazy na súbory ZIP nie sú povolené;
- URL_extension_count musí byť nulová;
- v obsahu aplikácie sa musí nachádzať len jeden transport_protocol_descriptor s protocol_id 0 x 0003.

5.3.7 Deskriptor umiestnenia jednoduchej aplikácie

V každej vnútornej slučke deskriptora *application* AIT sa môže tento deskriptor nachádzať len raz.

Tabuľka 33 – Syntax deskriptora umiestnenia jednoduchej aplikácie

	Počet bitov	Identifikátor	Hodnota
simple_application_location_descriptor () {			
descriptor_tag	8	uimsbf	0x15
descriptor_length	8	uimsbf	
for(i=0; i<N; i++) {			
initial_path_bytes	8	uimsbf	
}			
}			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 15 identifikuje tento deskriptor

bajty cesty inicializácie (angl. **initial_path_bytes**): bajty obsahujú reťazec, ktorý určuje cestu zložky URL vstupného bodu dokumentu

5.3.7.1 Príklad

Nasledujúci príklad opisuje použitie deskriptora simple_application_location_descriptor.

Autor aplikácie navrhuje aplikáciu nasledujúcim spôsobom:

- dáta aplikácie sa distribuujú do viacerých adresárov, napríklad do adresára "image" a adresára "main";
- vstupný bod aplikácie je dokument nazvaný "index.foo" a je uložený v adresári "main".

Z pohľadu autora aplikácie je vstupný bod aplikácie určený cestou "main/index.foo". Táto cesta je uložená v reťazci initial_path_bytes deskriptora umiestnenia.

Tabuľka 34 – Príklady znázorňujúce vstupný bod signalizácie aplikácie s rôznymi hodnotami identifikátora protocol_id

Hodnota protocol_id	Selektor	Vstupný bod výslednej aplikácie
0x0001	Tag zložky, napr. 0xb4	dvb://1.2.3.b4/main/index.foo
0x0003	Základná URL napr. "http://www.example.com/apps"	http://www.example.com/apps/main/index.foo

Ak sa vysielateľ rozhodne uložiť túto aplikáciu v systémovom súbore subadresára s názvom *application*, potom *initial_path_bytes* musí mať prefix s reťazcom "application/", t. j. *initial_path_bytes* musí mať hodnotu "application/main/index.foo".

5.3.8 Deskriptor ohraničenia jednoduchej aplikácie

Deskriptor je definovaný na použitie v slučke aplikácie AIT. Poskytuje nastavenie prefixov, ktoré opisujú dátové prvky a tie vytvárajú aplikáciu.

Deskriptor je voliteľný. V prípade jeho neprítomnosti predvolené ohraničenia aplikácie kompletne nastaví celý prichádzajúci obsah zo signalizovaného prenosu v *transport_protocol_descriptor* súvisiaceho s aplikáciou. Tento sa môže zrušiť špecifikáciou platformy.

Na tú istú aplikáciu sa môžu použiť viaceré deskriptory ohraničenia. V tomto prípade je aplikovateľné nastavenie rozšírení spojením nastavených rozšírení definovaných deskriptormi.

Tabuľka 35 – Syntax deskriptora ohraničenia jednoduchej aplikácie

	Počet bitov	Identifikátor	Hodnota
<code>simple_application_boundary_descriptor {</code>			
<code>descriptor_tag</code>	8	uimsbf	0x17
<code>descriptor_length</code>	8	uimsbf	
<code>boundary_extension_count</code>	8	uimsbf	
<code>for(j=0; j<boundary_extension_count; j++){</code>			
<code>boundary_extension_length</code>	8	uimsbf	
<code>for(k=0; k<boundary_extension_length; k++){</code>			
<code>boundary_extension_byte</code>	8	uimsbf	
<code>}</code>			
<code>}</code>			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 17 identifikuje tento deskriptor

počet rozšírení s ohraničením (angl. **boundary_extension_count**): osembitové pole určuje počet rozšírení s ohraničením sprostredkovaných týmto deskriptorom

dĺžka rozšírenia s ohraničením (angl. **boundary_extension_length**): osembitové pole udáva počet bajtov ohraničeného rozšírenia

bajt rozšírenia s ohraničením (angl. **boundary_extension_byte**): bajty tvoria prefix URL; akékoľvek adresy URL, ktoré obsahujú tento prefix, sú považované za ohraničené aplikácie; prefix URL je presne vymedzený (napríklad "http://www.example.com" namiesto "www.example.com") a môže obsahovať zložky cesty (napríklad "http://www.example.com/epg/"). Špecifikácie platformy môžu definovať minimálnu úroveň nespojitosti danú prefixom.

5.3.9 Informácie o službe

5.3.9.1 Deskriptor vysielania dát na oznámenie interaktívnej aplikácie

Generický deskriptor *data_broadcast_descriptor* je definovaný v EN 300 468 [1]. Tento článok definuje syntax a sémantiku selektora bajtov, keď identifikátor vysielania dát má hodnotu 0 x 00F2 (pozri tabuľku 36). V tomto prípade selektor bajtov poskytuje zoznam interaktívnych aplikácií a informácie o každej aplikácii. Na identifikáciu interaktívnych aplikácií súvisiacich s touto službou, alebo udalosťou, sa môže tento deskriptor nachádzať v tabuľkách SDT alebo EIT viackrát alebo sa nemusí nachádzať ani raz. Deskriptor indikuje len súvislosť medzi službou alebo udalosťou a aplikáciami. Umiestnenie každej uvedenej aplikácie sa musí riešiť prostredníctvom AIT. Deskriptor

nesmie zobrazit' zoznam aplikácií, ktorých návesť `test_application_flag` je nastavená v zodpovedajúcej pološke, alebo sa nastaví do zodpovedajúcej položky v AIT.

Tabuľka 36 – Syntax deskriptora rozšíreného vysielania dát – vysielanie id 0 x F2

	Počet bitov	Identifikátor	Hodnota
<code>data_broadcast_descriptor(){</code>			
<code>descriptor_tag</code>	8	uimsbf	
<code>descriptor_length</code>	8	uimsbf	
<code>data_broadcast_id</code>	16	uimsbf	
<code>component_tag</code>	8	uimsbf	
<code>selector_length</code>	8	uimsbf	
<code>for(i=0; i<selector_length; i++){</code>			
<code>organization_id</code>	32	uimsbf	
<code>application_id</code>	16	uimsbf	
<code>reserved_future_use</code>	1	bslbf	
<code>application_type</code>	15	uimsbf	
<code>application_profile_length</code>	8	uimsbf	
<code>for (j=0; j<N; j++){</code>			
<code>application_profile</code>	16	uimsbf	
<code>version.major</code>	8	uimsbf	
<code>version.minor</code>	8	uimsbf	
<code>version.micro</code>	8	uimsbf	
<code>}</code>			
<code>application_names_length</code>	8	uimsbf	
<code>for(j=0; j<N2;j++){</code>			
<code>ISO_639_language_code</code>	24	bslbf	
<code>application_name_length</code>	8	uimsbf	
<code>for(l=0; l<N3; l++){</code>			
<code>application_name_char</code>	8	bslbf	
<code>}</code>			
<code>}</code>			
<code>reserved_length</code>	8	uimsbf	
<code>for(j=0; i<N4; i++){</code>			
<code>reserved_future_use</code>	8	bslbf	
<code>}</code>			
<code>private_data_length</code>	8	uimsbf	
<code>for(j=0; j<N5; j++){</code>			
<code>private_data_byte</code>	8	bslbf	
<code>}</code>			
<code>}</code>			
<code>ISO_639_language_code</code>	24	bslbf	
<code>text_length</code>	8	uimsbf	
<code>for (i=0; i<text_length; i++){</code>			
<code>text_char</code>	8	uimsbf	
<code>}</code>			
<code>}</code>			

Sémantika deskriptora vysielania dát

Sémantika syntaxe nasledujúcich prvkov je definovaná v EN 300 468 [1]:

príznak deskriptora (angl. **descriptor_tag**): toto osembitové pole pozri v EN 300 468 [1]

dĺžka deskriptora (angl. **descriptor_length**): toto osembitové pole pozri v EN 300 468 [1]

identifikátor vysielania dát: (angl. **data_broadcast_id**): toto šestnásťbitové pole pozri v EN 300 468 [1]; pole má hodnotu 0 x 00 F2 (pozri tabuľku 38) pri oznámení interaktívnych aplikácií, bez ohľadu na spôsob(y) prenosu použitý(é) na interaktívnu aplikáciu a dáta

príznak zložky (angl. **component_tag**): toto osembitové pole pozri v EN 300 468 [1]

dĺžka selektora (angl. **selector_tag**): toto osembitové pole pozri v EN 300 468 [1]

Sémantiky syntaxe nasledujúcich prvkov sú definované v tomto dokumente:

identifikátor organizácie (angl. **organization_id**): tridsaťdvabitové pole kóduje identifikátor aplikácie `organisation_id`; pozri článok 5.2.3 "Identifikácia aplikácie"

identifikátor aplikácie (angl. **application_id**): šesťnásťbitové pole kóduje identifikátor aplikácie; pozri článok 5.2.3 "Identifikácia aplikácie"

typ aplikácie (angl. **application_type**): pätnásťbitové pole kóduje typ aplikácie; pozri článok 5.2.2 "Typy aplikácií"

dĺžka profilu aplikácie (angl. **application_profile_length**): osembitové pole špecifikuje dĺžku slučky profilu aplikácie v bajtoch

profil aplikácie (angl. **application_profile**): šesťnásťbitové pole identifikuje, ktorý špecifický typ profilu aplikácie požaduje daná aplikácia; pozri článok 5.2.5 "Profily platformy"

hlavná verzia (angl. **version.major**): osembitové pole špecifikuje číslo hlavnej verzie profilu; pozri článok 5.2.5 "Profily platformy"

vedľajšia verzia (angl. **version.minor**): osembitové pole špecifikuje číslo vedľajšej verzie profilu; pozri článok 5.2.5 "Profily platformy"

mikroverzia (angl. **version.micro**): osembitové pole špecifikuje číslo mikroverzie profilu; pozri článok 5.2.5 "Profily platformy"

dĺžka názvov aplikácie (angl. **application_names_length**): osembitové celé číslo bez znamienka špecifikuje počet bajtov v nasledujúcich viacjazyčných názvoch aplikácie

kód jazyka ISO 639 (angl. **ISO_639_language_code**): dvadsaťštyribitové pole kóduje `ISO_639_language_code` názvu aplikácie; pozri článok 5.3.5.6.1 "Deskriptor názvu aplikácie"

dĺžka názvu aplikácie (angl. **application_name_length**): osembitové pole kóduje dĺžku názvu aplikácie; pozri článok 5.3.5.6.1 "Deskriptor názvu aplikácie"

charakteristika názvu aplikácie (angl. **application_name_char**): pozri `application_name_char` v článku 5.3.5.6.1 "Deskriptor názvu aplikácie"

dĺžka rezervovania (angl. **reserved_length**): osembitové celé číslo bez znamienka špecifikuje počet nasledujúcich bajtov rezervovania

rezervované na budúce používanie (angl. **reserved_future_use**): osembitové pole

dĺžka neverejných dát (angl. **private_data_use**): osembitové celé číslo bez znamienka špecifikuje počet bajtov neverejných dát

bajt neverejných dát (angl. **private_data_byte**): osembitové pole

Sémantiky syntaxe nasledujúcich prvkov sú definované v EN 300 468 [1]:

kód jazyka ISO 639 (angl. **ISO_639_language_code**): toto dvadsaťštyribitové pole pozri v EN 300 468 [1].

dĺžka textu (angl. **text_length**): toto osembitové pole pozri v EN 300 468 [1]

charakteristika textu (angl. **text_char**): toto osembitové pole pozri v EN 300 468 [1]

5.3.10 Uložené aplikácie

5.3.10.1 Deskriptor uloženia aplikácie

Deskriptor `application_storage_descriptor` oznamuje, že aplikácia sa môže uložiť a poskytuje určité informácie o jej vlastnostiach. Prítomnosť tohto deskriptora naznačuje, že súbor opisu aplikácie je k dispozícii (pozri článok 5.2.11). Do uložiteľnej aplikácie sa musí umiestniť samostatný `application_storage_descriptor` do vonkajšej slučky deskriptora *common* alebo do vnútornej slučky deskriptora *application* tabuľky AIT.

Deskriptor a spomenutý súbor opisu aplikácie podporujú aj prijímače s implementáciou špekulatívneho kešovania.

Tabuľka 37 – Syntax deskriptora uloženia aplikácie

	Počet bitov	Identifikátor	Hodnota
<code>application_storage_descriptor() {</code>			
<code> descriptor_tag</code>	8	uimsbf	0x10
<code> descriptor_length</code>	8	uimsbf	
<code> Storage_property</code>	8	uimsbf	
<code> not_launchable_from_broadcast</code>	1	bslbf	
<code> launchable_completely_from_cache</code>	1	bslbf	
<code> is_launchable_with_older_version</code>	1	bslbf	
<code> reserved</code>	5	bslbf	
<code> reserved</code>	1	bslbf	
<code> version</code>	31	uimsbf	
<code> priority</code>	8	uimsbf	
<code>}</code>			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 10 identifikuje tento deskriptor

dĺžka deskriptora (angl. **descriptor_length**): osembitové pole indikuje počet bajtov, ktoré nasledujú po deskriptore dĺžky poľa

vlastnosti uloženia (angl. **storage_property**): osembitové pole indikuje aplikáciu súvisiacu s vysielaním alebo samostatnú aplikáciu

úplne spustiteľná z pamäte keš (angl. **launchable_completely_from_cache**): pozri článok 5.2.11.1.3 "Spúšťanie aplikácií z pamäte keš"

spustiteľná staršou verziou (angl. **is_launchable_with_older_version**): pozri článok 5.2.11.1.2 "Vytváranie verzií aplikácie"

nepustiteľná z vysielania (angl. **not_launchable_from_broadcast**): pozri článok 5.2.11.1.3 "Spúšťanie aplikácií z pamäte keš"

verzia (angl. **version**): pozri článok 5.2.11.1.2 "Vytváranie verzií aplikácie"

priorita (angl. **priority**): pozri článok 5.2.11.1.4 "Priorita uloženia"

5.4 Syntax založená na XML

Tento článok definuje kódovanie XML v AIT, okrem tabuľky MPEG-2 a okrem sekcie založenej na kódovaní definovanom v článku 5.3 tohto dokumentu. Odkedy sa používa kódovanie XML v

súvislosti so SD&S, ktoré je definované v TS 102 034 [6], toto kódovanie vychádza z rovnakého formátu a opakovane používa už definované prvky a typy.

Z vlastností kódovania v MPEG-2 vyplýva, že kódovanie v XML nepodporuje signalizáciu obmedzenia grafiky.

Sémantiky polí definované v tomto článku sa musia zhodovať s príslušnými poľami v existujúcej tabuľke MPEG-2 a kódovanie sekcie je definované v tomto dokumente.

Sledovanie zmien v tabuľke AIT kódovanej v XML sa musí vykonávať podľa článku 5.4.3 TS 102 034 [6].

Typ MIME použitý v AIT kódovanej v XML musí byť application/vnd.dvb.ait+xml. Prípona súboru musí byť ".aitx".

5.4.1 Signalizácia aplikácie súvisiacej so službou

Aplikácie súvisiace so službou sa musia signalizovať vrátane prvku ApplicationList v IPService alebo v balíčku prvkov SD&S (pozri TS 102 034 [6]). Toto je v plnom rozsahu špecifikované v článkoch 5.4.3.2 a 5.4.3.1 tohto dokumentu.

Aplikácie sú definované v rade s prvkom ApplicationList alebo v zázname ApplicationDiscovery, ako je to definované v článku 5.4.5. V druhom prípade sa používa ApplicationIdentifier v ApplicationList ako odkaz na aplikáciu v zázname ApplicationDiscovery.

Alternatívne, služba môže obsahovať AIT vo formáte MPEG-2 v rámci toku. Oznamenie, že prenosový tok obsahuje túto AIT, signalizuje prvok IPService. Začlenenie AIT vo formáte MPEG-2 v rámci toku zakazuje použitie AIT s kódovaním v XML.

5.4.2 Signalizácia nesúvisiacich aplikácií

Nesúvisiace aplikácie, t. j. aplikácie, ktoré nie sú priradené ku konkrétnej službe, sú signalizované jedným alebo viacerými prvkami AbstractIPService v službe SD&S poskytovateľa sprístupnenia záznamu (pozri TS 102 034 [6]). Toto je v plnom rozsahu špecifikované v článku 5.4.3.3.

Aplikácie sú definované postupne s prvkom ApplicationList v AbstractIPService alebo v zázname ApplicationDiscovery, ako je to definované v článku 5.4.5. V druhom prípade sa ApplicationIdentifier používa v prvku ApplicationList v AbstractIPService ako odkaz na aplikáciu v zázname ApplicationDiscovery.

5.4.3 Rozšírenia definované prvkami SD&S

5.4.3.1 Balíček

Zoznam aplikácie signalizovaný v balíčku obsahuje množinu aplikácií, ktoré sú k dispozícii pri všetkých signalizovaných IPServices v balíčku. ApplicationList je pridaný ako rozšírenie typu balíčka definované v TS 102 034 [6].

```
<xsd:complexType name="PackageType">
  <xsd:complexContent>
    <xsd:extension base="ipi:Package">
      <xsd:sequence>
        <xsd:element name="ApplicationList" type="mhp:ApplicationList" minOccurs="0" />
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

POZNÁMKA. – Pridanie aplikácie do balíčka je významovo rovnaké ako pridanie aplikácie na všetky služby tohto balíčka.

5.4.3.2 Služba IP

Aplikácie súvisiace so službou na jednotlivé služby sú signalizované rozšírením typu IPService definovaného v TS 102 034 [6].

```
<xsd:complexType name="IPServiceType">
  <xsd:complexContent>
    <xsd:extension base="ipi:IPService">
      <xsd:choice>
        <xsd:element name="ApplicationList" type="mhp:ApplicationList" minOccurs="0"/>
        <xsd:element name="AITDescriptor" type="mhp:AITDescriptorType" minOccurs="0"/>
      </xsd:choice>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="AITDescriptorType" />
```

Toto rozšírenie pridáva prvok ApplicationList na koniec prvku IPService. Prvok ApplicationList je konkretizáciou zápisu ApplicationList, ktorý je definovaný v článku 5.4.4.1 tohto dokumentu.

Tabuľka AIT služby sa môže vkladať do prenosového toku pomocou syntaxe MPEG2. Toto sa môže signalizovať prvkom AITDescriptor. Ak je signalizované vkladanie AIT, prvok ApplicationList nesmie byť prítomný.

5.4.3.3 Prevádzkovateľ služby

Nesúvisiace aplikácie sú signalizované rozšírením typu ServiceProviderType definovaným v TS 102 034 [6].

```
<xsd:complexType name="ServiceProviderType">
  <xsd:complexContent>
    <xsd:extension base="ipi:ServiceProviderType">
      <xsd:sequence>
        <xsd:element name="AbstractService" type="mhp:AbstractIPService"
          maxOccurs="unbounded" minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Rozšírenie pridáva prvok AbstractService na koniec poskytovateľom definovanej služby. Prvok AbstractService je konkretizáciou zápisu AbstractIPService, ktorý je definovaný v článku 5.4.4.15 tohto dokumentu.

5.4.4 Definície nových prvkov XML

5.4.4.1 Zoznam aplikácie

```
<xsd:complexType name="ApplicationList">
  <xsd:sequence minOccurs="0" maxOccurs="unbounded">
    <xsd:element name="application" type="mhp:Application" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="ApplicationReference" type="mhp:ApplicationIdentifier" minOccurs="0"
      maxOccurs="unbounded">
    </xsd:sequence>
  </xsd:complexType>
```

ApplicationList je zoznam aplikácie alebo prvkov ApplicationReference. ApplicationReference je konkretizáciou zápisu ApplicationIdentifier, ktorý je definovaný v článku 5.4.4.3 tohto dokumentu. Odkaz sa rieši hľadaním ApplicationIdentifier v záznamoch ApplicationDiscovery toho istého poskytovateľa služieb.

5.4.4.2 Aplikácia

```

<xsd:complexType name="Application">
  <xsd:sequence>
    <xsd:element name="appName" type="ipi:MultilingualType" maxOccurs="unbounded"/>
    <xsd:element name="applicationIdentifier" type="mhp:ApplicationIdentifier"/>
    <xsd:element name="applicationDescriptor" type="mhp:ApplicationDescriptor"/>
    <xsd:element name="applicationSpecificDescriptor"
      type="mhp:ApplicationSpecificDescriptor" minOccurs="0"/>
    <xsd:element name="applicationUsageDescriptor"
      type="mhp:ApplicationUsageDescriptor" minOccurs="0"/>
    <xsd:element name="applicationBoundary"
      type="mhp:SimpleApplicationBoundaryDescriptorType"
      minOccurs="0"
      maxOccurs="1" />
    <xsd:element name="applicationTransport"
      type="mhp:TransportProtocolDescriptorType"
      minOccurs="1"
      maxOccurs="unbounded" />
    <xsd:element name="applicationLocation"
      type="mhp:SimpleApplicationLocationDescriptorType"
      minOccurs="1"
      maxOccurs="1" />
  </xsd:sequence>
</xsd:complexType>

```

Aplikácia sa môže kompletne opísať:

- názvom aplikácie, ktorý môže byť viacjazyčný (appName);
- jednoznačnou identifikáciou (applicationIdentifier);
- generickým deskriptorom, ktorý je spoločný a záväzný vo všetkých typoch aplikácií (applicationDescriptor);
- konkrétnym deskriptorom aplikácie, ktorý závisí od typu signalizovanej aplikácie;
- deskriptorom používania aplikácie, ktorý je voliteľný;
- deskriptorom ohraničenia aplikácie, ktorý je voliteľný;
- jedným alebo viacerými deskriptormi prenosu aplikácie;
- deskriptorom umiestnenia jednoduchej aplikácie.

5.4.4.2.1 Špecifická informácia o aplikácii (informatívne)

Niektoré špecifikácie platformy môžu zvoliť opis alebo požadovať, aby sa oznamovali koncovému zariadeniu ďalšie informácie, ktoré sú mimo rozsahu aktuálne prenášaných informácií v prvku aplikácie, napríklad dodatočný protokol prenosu informácie alebo informácie súvisiace s bezpečnosťou. Aby to bolo možné, typ aplikácie definovaný v článku 5.4.4.2 sa môže rozšíriť podľa odporúčania rozšíriteľnosti XML definovaného v [17].

5.4.4.3 Identifikátor aplikácie

```

<xsd:complexType name="ApplicationIdentifier">
  <xsd:sequence>
    <xsd:element name="orgId" type="xsd:unsignedInt" />
    <xsd:element name="appId" type="xsd:unsignedShort" />
  </xsd:sequence>
</xsd:complexType>

```

Ako je definované v článku 5.2.3 "Identifikácia aplikácie" a aplikácia jednoznačne identifikovaná:

- globálne jedinečným identifikátorom OrgId, ktorý identifikuje organizáciu zodpovednú za aplikáciu;
- identifikátorom aplikácie Appld prideleným registrovanou organizáciou s identifikátorom organizácie, ktorá rozhoduje o politike pridelovania v rámci organizácie.

5.4.4.4 Deskriptor aplikácie

```
<xsd:complexType name="ApplicationDescriptor">
  <xsd:sequence>
    <xsd:element name="type" type="mhp:ApplicationType"/>
    <xsd:element name="controlCode" type="mhp:ApplicationControlCode"/>
    <xsd:element name="visibility" type="mhp:VisibilityDescriptor" minOccurs="0"/>
    <xsd:element name="serviceBound" type="xsd:boolean" default="true" minOccurs="0"/>
    <xsd:element name="priority" type="ipi:Hexadecimal8bit"/>
    <xsd:element name="version" type="ipi:Version"/>
    <xsd:element name="mhpVersion" type="mhp:MhpVersion" minOccurs="0"/>
    <xsd:element name="icon" type="mhp:IconDescriptor" minOccurs="0"/>
    <xsd:element name="storageCapabilities" type="mhp:StorageCapabilities" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Obsahy ich úplného znenia sú väčšinou definované v článku 5.3.5.3 "Deskriptor aplikácie".

Jednotlivé typy prvkov sú definované takto:

viditeľnosť (angl. **visibility**): voliteľný prvok určuje, či je aplikácia vhodná ako ponuka koncovému používateľovi, aby rozhodol, či sa má aplikácia spustiť; pozri článok 5.2.6 "Viditeľnosť aplikácie"

súvisiaca služba (angl. **serviceBound**): určuje, či aplikácia súvisí so službou alebo nie, podľa definovania `service_bound_flag`; pozri článok 5.3.5.3 "Deskriptor aplikácie"

priorita (angl. **priority**): pole určuje relatívnu prednosť medzi aplikáciami signalizovanými v tejto službe; pozri článok 5.2.7 "Priorita aplikácie"

verzia (angl. **version**): pozri článok 5.2.11.1.2 "Vytváranie verzií aplikácie"

verzia MHP (angl. **mhpVersion**): pozri článok 5.4.4.8

ikona (angl. **ikon**): signalizuje prítomnosť ikony, ktorá reprezentuje aplikáciu

kapacita pamätí (angl. **storageCapabilities**): voliteľný prvok, ktorý sa musí pridať s cieľom poskytnúť požadovanú informáciu na uloženie/kešovanie

5.4.4.5 Deskriptor viditeľnosti

```
<xsd:simpleType name="VisibilityDescriptor">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="NOT_VISIBLE_ALL"/>
    <xsd:enumeration value="NOT_VISIBLE_USERS"/>
    <xsd:enumeration value="VISIBLE_ALL"/>
  </xsd:restriction>
</xsd:simpleType>
```

Tieto hodnoty sú definované v tabuľke 5 v článku 5.2.6.1 "Sémantika".

5.4.4.6 Deskriptor ikony

```
<xsd:complexType name="IconDescriptor">
  <xsd:attribute name="filename" type="xsd:string" use="required"/>
  <xsd:attribute name="size" type="xsd:unsignedShort" use="optional"/>
  <xsd:attribute name="aspectRatio" type="mhp:AspectRatio" use="optional"/>
</xsd:complexType>
```

Podľa článku 5.2.8 "Ikony aplikácie" slúži prvok IconDescriptor na indikáciu prítomnosti ikony, ktorá predstavuje aplikáciu. Atribúty veľkosť a pomer strán ikony sú voliteľné a definované v tabuľke 7.

Napríklad:

```
<icon filename="dvb.icon.1"/> , size = 32x32 pixel square
```

5.4.4.7 Pomer strán

```
<xsd:simpleType name="AspectRatio">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="4_3"/>
    <xsd:enumeration value="16_9"/>
    <xsd:enumeration value="1_1"/>
  </xsd:restriction>
</xsd:simpleType>
```

Pomery strán sú množinou pomerov strán použitých v tabuľke 7 v článku 5.2.8 "Ikony aplikácie".

5.4.4.8 Verzia Mhp

```
<xsd:complexType name="MhpVersion">
  <xsd:sequence minOccurs="1">
    <xsd:element name="profile" type="ipi:Hexadecimal16bit"/>
    <xsd:element name="versionMajor" type="ipi:Hexadecimal8bit"/>
    <xsd:element name="versionMinor" type="ipi:Hexadecimal8bit"/>
    <xsd:element name="versionMicro" type="ipi:Hexadecimal8bit"/>
  </xsd:sequence>
</xsd:complexType>
```

Prvky sú definované týmto spôsobom:

profil (angl. **profile**): pozri application_profile v článku 5.2.5 "Profily platformy"

hlavná verzia (angl. **versionMajor**): pozri version.major v článku 5.2.5 "Profily platformy"

vedľajšia verzia (angl. **versionMinor**): pozri version.minor v článku 5.2.5 "Profily platformy"

mikroverzia (angl. **versionMicro**): pozri version.micro v článku 5.2.5 "Profily platformy"

POZNÁMKA. – Tento typ je pomenovaný z historických dôvodov.

5.4.4.9 Kapacita pamäti

```
<xsd:complexType name="StorageCapabilities">
  <xsd:sequence minOccurs="0">
    <xsd:element name="storageProperty" type="mhp:StorageType"/>
  </xsd:sequence>
  <xsd:attribute name="launchableFromBroadcast" type="xsd:boolean" use="required"/>
  <xsd:attribute name="launchableCompletelyFromCache" type="xsd:boolean" use="required"/>
  <xsd:attribute name="launchableWithOlderVersion" type="xsd:boolean" use="required"/>
</xsd:complexType>
```

Tento deskriptor, ak je použitý, vyjadruje, či sa aplikácia môže uložiť alebo kešovať, ako je to definované v bode 5.2.11 "Uložené aplikácie".

Atribúty launchableFromBroadcast, launchableCompletelyFromCache, launchableWithOlderVersion majú presne rovnaký význam ako príznaky definované v článkoch 5.2.11.1.2 a 5.2.11.1.3.

5.4.4.10 Typ pamäte

```
<xsd:simpleType name="StorageType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="BROADCAST-RELATED"/>
  </xsd:restriction>
</xsd:simpleType>
```

```

    <xsd:enumeration value="STANDALONE" />
  </xsd:restriction>
</xsd:simpleType>

```

Pozri článok 5.2.11.1.1 "Životný cyklus uložených aplikácií".

5.4.4.11 Typ aplikácie

```

<xsd:complexType name="ApplicationType">
  <xsd:choice>
    <xsd:element name="DvbApp" type="mhp:DvbApplicationType" />
    <xsd:element name="OtherApp" type="mpeg7:mimeType" />
  </xsd:choice>
</xsd:complexType>

```

Pozri článok 5.2.2 "Typy aplikácií".

5.4.4.12 Typ aplikácie DVB

```

<xsd:simpleType name="DvbApplicationType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="DVB-J" />
    <xsd:enumeration value="DVB-HTML" />
  </xsd:restriction>
</xsd:simpleType>

```

5.4.4.13 Riadiaci kód aplikácie

```

<xsd:simpleType name="ApplicationControlCode">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="AUTOSTART" />
    <xsd:enumeration value="PRESENT" />
    <xsd:enumeration value="DESTROY" />
    <xsd:enumeration value="KILL" />
    <xsd:enumeration value="PREFETCH" />
    <xsd:enumeration value="REMOTE" />
    <xsd:enumeration value="DISABLED" />
    <xsd:enumeration value="PLAYBACK_AUTOSTART" />
  </xsd:restriction>
</xsd:simpleType>

```

Tento deskriptor slúži na dynamické riadenie životného cyklu aplikácie. Význam každého jedného z vymenovaných prvkov, rovnako ako očakávané správanie v prijímači, je v plnom rozsahu definované v článku 5.2.4 "Riadiace kódy aplikácie".

5.4.4.14 Deskriptor špecifickej aplikácie

```

<xsd:complexType name="ApplicationSpecificDescriptor">
  <xsd:choice>
    <xsd:element name="dvbjDescriptor" type="mhp:DVBjDescriptor" />
    <xsd:element name="htmlDescriptor" type="mhp:DVBhtmlDescriptor" />
    <xsd:element name="otherDescriptor" type="mhp:OtherDescriptor" />
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="OtherDescriptor" abstract="true" />

```

Tento deskriptor obsahuje špecifický deskriptor, ktorý závisí od typu aplikácie. Rovnako deskriptory definované v tomto dokumente môžu obsahovať aj externe definované deskriptory.

POZNÁMKA. – DVBJDescriptor a DVBHTMLDescriptor nie sú predmetom tohto dokumentu.

5.4.4.15 Abstraktná IP služba

```

<xsd:complexType name="AbstractIPService">
  <xsd:sequence>
    <xsd:element name="svcName" type="ipi:MultilingualType" maxOccurs="unbounded" />
    <xsd:element name="svcId" type="mhp:Hexadecimal24bit" />
  </xsd:sequence>
</xsd:complexType>

```

```

    <xsd:element name="isAutoSelect" type="xsd:boolean"/>
    <xsd:element name="ApplicationList" type="mhp:ApplicationList" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="Hexadecimal24bit">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9a-fA-F]{1}[1-9a-fA-F]{1}[0-9a-fA-F]{4}"/>
  </xsd:restriction>
</xsd:simpleType>

```

Prvky sú definované týmto spôsobom:

názov svc (angl. **svcName**): názov abstraktnej služby

identifikátor svc (angl. **svclId**): identifikátor abstraktnej služby; ten musí byť jedinečný v rámci abstraktných služieb signalizovaných pre poskytovateľa služby

je automaticky vybraná (angl. **isAutoSelect**): príznak indikujúci službu, ktorá sa má automaticky spustiť; ak je hodnota tohto prvku pravdivá, potom sa služba musí automaticky spustiť, ak je vybraný poskytovateľ služby; ak je nepravdivá, nesmie sa spustiť

5.4.4.16 Typ ponúkanej služby

```

<xsd:complexType name="ApplicationOfferingType">
  <xsd:complexContent>
    <xsd:extension base="ipi:OfferingBase">
      <xsd:sequence>
        <xsd:element name="ApplicationList" maxOccurs="unbounded">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="Application" type="mhp:Application"
                maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

```

Tento prvok sa môže použiť poskytovateľom služby ako zoznam ponúkaných aplikácií.

5.4.4.17 Zistenie aplikácie

```

<xsd:element name="ServiceDiscovery">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element name="ApplicationDiscovery" type="mhp:ApplicationOfferingType"
        maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>

```

POZNÁMKA. – Tento prvok tvorí hlavný prvok dokumentu SD&S XML, ktorý definuje jeden alebo viac záznamov zistenej aplikácie. Ďalšie ponúkané záznamy SD&S definované v TS 102 034 [6] sú obsiahnuté v hlavnom prvku ServiceDiscovery, ktorý je definovaný v TS 102 034 [6]. Vhodný prvok ServiceDiscovery je identifikovaný v dokumente XML prostredníctvom jeho prefixu namespace.

5.4.4.18 Deskriptor používania aplikácie

```

<xsd:complexType name="ApplicationUsageDescriptor">
  <xsd:sequence>
    <xsd:element name="ApplicationUsage" type="xsd:anyURI" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

```

Pozri článok 5.2.10.3 – definícia sémantiky tohto prvku.

5.4.4.19 Typ deskriptora prenosového protokolu

```
<xsd:complexType name="TransportProtocolDescriptorType" abstract="true" />
```

Tento typ definuje základnú triedu deskriptora prenosového protokolu. Ide o abstraktný typ; podtriedy tohto typu sú definované na podporu špecifických prenosových protokolov (pozri článok 5.3.6 na širšiu diskusiu o deskriptoroch prenosového protokolu).

5.4.4.20 Typ prenosu HTTP

```
<xsd:complexType name="HTTPTransportType">
  <xsd:complexContent>
    <xsd:extension base="mhp:TransportProtocolDescriptorType">
      <xsd:sequence>
        <xsd:element name="URLBase" type="xsd:anyURI"
          minOccurs="1"
          maxOccurs="1" />
        <xsd:element name="URLExtension" type="xsd:anyURI"
          minOccurs="0"
          maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Ak je vstupný bod do aplikácie prístupný prostredníctvom protokolu http, musí sa použiť deskriptor prenosového protokolu HTTP. Aktuálna adresa URL vstupného bodu sa skladá zo základnej adresy URL zlúčenej s cestou poskytnutou deskriptorom umiestnenia jednoduchej aplikácie.

V URLBase a URLExtension platí rovnaká sémantika ako aj v URL_base_byte a URL_extension_byte, v tomto poradí v článku 5.3.6.2 tohto dokumentu.

5.4.4.21 Typ prenosu OC

```
<xsd:complexType name="OCTransportType">
  <xsd:complexContent>
    <xsd:extension base="mhp:TransportProtocolDescriptorType">
      <xsd:sequence>
        <xsd:choice maxOccurs="1" minOccurs="0">
          <xsd:element name="DvbTriplet" type="ipi:DVBTriplet">
            </xsd:element>
          <xsd:element name="TextualId" type="ipi:TextualIdentifier">
            </xsd:element>
        </xsd:choice>
        <xsd:element name="ComponentTag" minOccurs="1" maxOccurs="1"
          type="mhp:ComponentTagType">
            </xsd:element>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

Prvok ApplicationType typu OTransportType musí byť prítomný, ak je aplikácia doručená prostredníctvom objektového karuselu DSMCC. Ak je táto aplikácia spojená so službou a karusel je súčasťou tejto služby, potom sa môže identifikátor služby vynechať.

V aplikáciách, ktoré nie sú spojené so službou, sa musí použiť identifikátor, napríklad aplikácie poskytovateľa služby, DvbTriplet alebo TextualId služby.

5.4.4.22 Príznak typu zložky

```
<xsd:complexType name="ComponentTagType">
  <xsd:attribute name="ComponentTag" type="ipi:Hexadecimal8bit">
    </xsd:attribute>
</xsd:complexType>
```

Tento typ definuje zastúpenie príznaku zložky DVB.

5.4.4.23 Typ deskriptora umiestnenia jednoduchej aplikácie

```
<xsd:simpleType name="SimpleApplicationLocationDescriptorType">
  <xsd:restriction base="xsd:anyURI" /> </xsd:simpleType>
```

Tento deskriptor je definovaný v článku 5.3.7.

5.4.4.24 Typ deskriptora ohraničenia jednoduchej aplikácie

```
<xsd:complexType name="SimpleApplicationBoundaryDescriptorType">
  <xsd:sequence>
    <xsd:element name="BoundaryExtension" type="xsd:anyURI"
      minOccurs="1"
      maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>
```

Tento deskriptor poskytuje množinu prefixov, ktoré opisujú prvky dát, ktoré tvoria aplikáciu (pozri článok 5.3.8 – sémantika tohto deskriptora).

5.4.5 Záznam sprístupnenia aplikácie

Záznam ApplicationDiscovery je konkretizácia ApplicationOfferingType prenášaného prvkom ServiceDiscovery, ako je uvedené v článkoch 5.4.4.16 a 5.4.4.17.

Prítomnosť a získanie umiestnenia záznamu ApplicationDiscovery sú signalizované zo záznamu ServiceProviderDiscovery pomocou OfferingListType (pozri článok 5.2.5 v TS 102 034 [6]). PayloadId záznam ApplicationDiscovery musí byť 0 x C1.

5.5 Konštantné hodnoty

Tabuľka 38 – Register konštantných hodnôt

Kde sa používajú	Typ	Hodnota	Kde sú definované	Obsah
Deskriptor špecifikátora neverejných dát	tag deskriptora	0x5F	Tabuľky PSI a SI	SI
Deskriptor id vysielača dát		0x66	PMT	
Deskriptor signalizácie aplikácie		0x6F	PMT	
Deskriptor identifikátora služby		0x71	SDT	
Deskriptor priority kešovania	tag deskriptora	0x71	DII moduleInfo userInfo	SI-DAT
Deskriptor typu obsahu		0x72	BIOP objectInfo (poznámka 1)	
Rezervované na DVB pre budúce deskriptory OC		0x73 až 0x7F	OC	
Rezervované na DVB na budúce používanie	tabuľka ID do AIT PID	0x00 až 0x73		Tento dokument
Tabuľka informácií o aplikácii		0x74		
Rezervované na DVB na budúce používanie		0x75 až 0x7F		
Rezervované na neverejné používanie		0x80 až 0xFF		
Deskriptor aplikácie	tag deskriptora	0x00	AIT	Tento dokument
Deskriptor názvu aplikácie		0x01		
Deskriptor transportného protokolu		0x02		

Kde sa používajú	Typ	Hodnota	Kde sú definované	Obsah
Rezervované na DVB na budúce používanie		0x03, 0x04		
Deskriptor autorizácie externej aplikácie		0x05		
Deskriptor záznamu aplikácie		0x06		
Rezervované na DVB na budúce používanie		0x07 – 0x0A		
Deskriptor ikon aplikácie		0x0B		
Rezervované na DVB na budúce používanie		0x0C – 0x0F		
Deskriptor uloženia aplikácie		0x10		
Rezervované na DVB na budúce používanie		0x11 až 0x13		
Deskriptor obmedzenia grafiky (pozri článok 5.3.5.8 "Deskriptor obmedzenia grafiky")		0x14	5.2.5.8	
Deskriptor umiestnenia jednoduchej aplikácie		0x15		
Deskriptor používania aplikácie		0x16		
Deskriptor ohraničenia jednoduchej aplikácie		0x17		
Rezervované na DVB na budúce používanie		0x18 až 0x5E		
Deskriptor špecifikátora neverejných dát (poznámka 2)		0x5F		
Predmet registrácie na http://www.dvb.org		0x60 až 0x7F		
Používateľom definované (poznámka 3)	0x80 až 0xFE			
Objektový karusel DVB	id vysielania dát	0x00F0	PMT, AIT	SI
Rezervované		0x00F1		
Prítomnosť aplikácie DVB		0x00F2	EIT, SDT	SI
Rezervované na DVB na budúce používanie		0x00F3 - 0x00FE	PMT, AIT	SI
<p>POZNÁMKA 1. – Presne MessageSubHeader::ObjectInfo v súbore správy a súvisiaci info objekt v súbore povinného adresára alebo v správe služby sieťového prechodu.</p> <p>POZNÁMKA 2. – Deskriptor špecifikátora neverejných dát DVB SI je definovaný na používanie v tabuľke informácií o aplikácii na zavedenie neverejných deskriptorov.</p> <p>POZNÁMKA 3. – Všetky používateľom definované deskriptory musia byť v rámci deskriptora špecifikátora neverejných dát (pozri článok 5.3.4.7 "Používanie neverejných deskriptorov v AIT").</p>				

6 Odkazovanie na služby DVB

6.1 Syntax a sémantika URL DVB

Syntax a sémantika schémy URL "dVB:" sú definované v [20].

6.2 Rozlíšenie URL DVB

6.2.1 Deskriptor identifikátora služby

Opis služby SDT môže obsahovať viac deskriptorov `service_identifier_descriptor` alebo nemusí obsahovať žiadny deskriptor. Každý takýto deskriptor definuje v službe jediný identifikátor textu. Syntax identifikátora textovej služby je:

```
<service_name> "." <service_provider_domain_name> ,
```

kde:

```
<service_name>
```

je jedinečný názov služby v rámci domény poskytovateľa služby.

```
<service_provider_domain_name>
```

je názov internetovej domény DNS, ktorú má poskytovateľ služieb právo riadiť. Organizácia spravujúca názvy internetových domén DNS používa globálne jedinečný registračný mechanizmus, ktorý umožňuje pridelovať týmto identifikátorom textovej služby globálne jedinečné názvy.

Pole `<service_name>` musí spĺňať pravidlá definované na internetové názvy DNS tak, že celý identifikátor textovej služby je platný názov hostiteľa, ktorý sa používa na internetovú DNS, ako je definované v RFC 1035 [21].

Príklad identifikátora textovej služby je:

```
movie-channel-1.broadcaster-b.com,
```

kde "broadcaster-b.com" je internetová doména DNS vo vlastníctve vysielateľa a "movie-channel-1" je jedinečný názov služby pridelený poskytovateľom služby.

POZNÁMKA. – Identifikátor textovej služby má rovnakú syntax ako internetový názov hostiteľa a tá sa má priradiť v doméne, ktorú poskytovateľ služby riadi. Názov textovej služby nie je nutné riešiť s akoukoľvek adresou IP použitím internetovej služby DNS, a ak áno, tak táto verzia existujúceho dokumentu nestanovuje žiadne špecifické služby, ktoré má hostiteľ s adresou IP poskytnúť na pripojenie prostredníctvom protokolov IP.

Jediný identifikátor služby sa môže priradiť službám v rôznych fyzických sieťach, aj keď majú rôzne `original_network_id` a `service_id`. Daný identifikátor služby sa musí priradiť len službám, ktoré sa považujú za rovnaké služby.

POZNÁMKA. – Je na poskytovateľovi služieb, aby rozhodol, ktoré služby sú "rovnaké", a ktoré nie. Napríklad dve služby v dvoch rôznych sieťach sa môžu všeobecne považovať za "rovnaké" služby, ak služba má rovnaký programový obsah, ale rôzne regionálne inzeráty. Toto rozhodnutie je v plnej kompetencii poskytovateľa služieb.

V prípade služby sa môže prideliť viac identifikátorov služby.

Tabuľka 39 – Deskriptor identifikátora služby

	Počet bitov	Identifikátor	Hodnota
service_identifier_descriptor () {			
descriptor_tag	8	uimsbf	0x71
descriptor_length	8	uimsbf	
for (i = 0; i < descriptor_length; i++) {			
textual_service_identifier_bytes	8	uimsbf	
}			
}			

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 71 identifikuje tento deskriptor

identifikátor bajtov textovej služby (angl. **textual_service_identifier_bytes**): bajty obsahujúce jedinečný identifikátor služby kódovanej pomocou bežného kódovania textových reťazcov v DVB SI

7 Prenos aplikácie

7.1 Objektový karusel

Tento článok opisuje protokol, ktorý sa používa v prípade, keď sa vysielané aplikácie prenášajú pomocou objektových karuselov DSM-CC používateľ – používateľ.

Dokument vychádza z nasledujúcich špecifikácií:

- ISO/IEC 13818-1 [3] – systémy MPEG 2;
- ISO/IEC 13818-6 [4] – DSM-CC;
- EN 301 192 [2] – špecifikácia DVB na vysielanie dát;
- TR 101 202 [i.2] – implementačné pokyny na vysielanie dát, s obmedzeniami a rozšíreniami opísanými v prílohe B.

7.2 HTTP

Keď sa aplikácie sťahujú pomocou protokolu HTTP, musí sa protokol HTTP 1.1 podporovať tak, ako je definované v RFC 2616 [5].

8 Synchronizácia

8.1 Úvod

Tento dokument podporuje synchronizáciu tokov videa alebo tokov audia v službe pomocou tokov udalostí DSM-CC tak, ako je definované v článku B.2.4.

Tie môžu byť:

- udalosti "urob to teraz" sú definované v článku B.2.4.2.2; tieto udalosti sa posielajú do aplikácie, len čo ich prijme koncové zariadenie;
- udalosti synchronizované časovou základňou DVB, definované v článku B.2.4.2.2; tieto udalosti sa posielajú do aplikácie, keď časová základňa dosiahne čas signalizovaný udalosťou.

Špecifikácie platformy majú definovať, ktoré z nich sú podporované pri ich využití v prípade potrebnej synchronizácie.

8.2 Odkazovanie

Definované sú dva spôsoby odkazovania na prostriedky toku udalostí z aplikácií:

- odkazovaním sa na tok objektu udalosti DSM-CC v objektovom karuseli; toto vyžaduje služba, ktorá obsahuje objektový karusel, ako aj základný tok prenášajúci správy toku udalosti;
- odkazovaním sa na súbor XML, ktorý obsahuje rovnocenné informácie ako tok objektu udalosti DSM-CC, ktorý je definovaný v nasledujúcom príklade; toto umožňuje synchronizáciu služieb, ktoré prenášajú tok správ udalosti, ale neobsahujú objektový karusel.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!--W3C Schema generated by XMLSpy v2006 sp2 U (http://www.altova.com)-->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:dsmcc="urn:dvb:mis:dsmcc:2009"
  targetNamespace="urn:dvb:mis:dsmcc:2009" elementFormDefault="qualified"
  attributeFormDefault="qualified">
  <xs:complexType name="DsmccType">
    <xs:sequence>
      <xs:element name="dsmcc_object" type="dsmcc:DsmccObjectType" minOccurs="0"
        maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="dsmcc" type="dsmcc:DsmccType"/>
  <xs:complexType name="DsmccObjectType">
    <xs:sequence>
      <xs:element name="stream_event" type="dsmcc:StreamEventType" minOccurs="0"
        maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="component_tag" type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="StreamEventType">
    <xs:attribute name="stream_event_id" type="xs:string" use="required"/>
    <xs:attribute name="stream_event_name" type="xs:string" use="required"/>
  </xs:complexType>
</xs:schema>
```

Príloha A (informatívna): Prvky definované špecifikáciou platformy

A.1 Úvod

Príloha opisuje prvky, ktorých sémantika je definovaná špecifikáciou platformy.

A.2 Prvky definované špecifikáciou platformy

Tabuľka A.1 – Prvky definované špecifikáciou platformy

Článok	Prvok
5.2.5	Profilovanie aplikácie a vytváranie verzií
5.2.8	Umiestnenie ikony aplikácie (keď je uvedený lokátor ikony app)
5.2.8	Kódovanie lokátora ikon aplikácie
5.2.12	Umiestnenie súboru opisu aplikácie
5.3.4.2	Čas, v ktorom budú detegované zmeny AIT
5.3.4.2	Minimálna opakovacia frekvencia do každú subtabuľku AIT
5.3.4.2	Čas, v ktorom bude aktualizácia AIT detekovaná
5.3.5.6.1	Kódovanie názvu aplikácie vo formáte MPEG-2 deskriptora názvu aplikácie
B.2.4.6.1	Čas, v ktorom budú detegované aktualizácie balíka časových základní
B.2.4.6.2	Čas, v ktorom sú detegované zmeny časov spustenia udalostí

Príloha B (normatívna): Objektový karusel

B.1 Úvod

Príloha opisuje obmedzenia a rozšírenia špecifikácií uvedených v článku 7.1 pri používaní objektového karuselu DSM-CC používateľ – používateľ na prenášanie vysielaných aplikácií.

B.1.1 Kľúč k poznámkam

V tabuľkách v stĺpcoch "hodnota" sa používajú tieto syntaktické poznámky.

Tabuľka B.1 – Legenda k poznámkam

Symbol	
+	Hodnota, ktorá je "pridelená", napríklad konfiguračný parameter servera objektového karuselu.
*	Hodnota, ktorá sa "počíta", napríklad pole, ktorého hodnotu vypočíta server karuselu podľa počtu bajtov v iných poliach.

B.2 Profil objektového karuselu

V nasledujúcom článku sú uvedené štruktúry správy objektového karuselu súvisiace s ďalšími obmedzeniami. Každá sekcia obsahuje tabuľku s uvedením obmedzení na používanie polí. Tabuľka tiež uvádza zdroj týchto obmedzení: štandard DSM-CC, pokyny DVB alebo osobitné obmedzenia na tento dokument.

V správach objektového karuselu sa nachádza aj syntax správy. V tabuľkách je syntax označená šedým tieňovaním, ktoré vysielateľ môže vložiť, ale koncové zariadenie v súlade s touto normou to môže ignorovať.

B.2.1 Sekcie DSM-CC

Všetky správy objektového karuselu sú prenášané pomocou formátu sekcie DSM-CC, ktorý je definovaný v článku 9.2 špecifikácie DSM-CC [4].

Norma DSM-CC poskytuje možnosť použiť CRC32 alebo kontrolný súčet na detegovanie chybných bitov. V rámci tohto dokumentu sú nasledujúce obmedzenia.

Tabuľka B.2 – Obmedzenia formátu sekcie DSM-CC

Pole	Obmedzenia	Zdroj
section_syntax_indicator	1 (s uvedením použitia CRC32)	Tento dokument
last_section_number	Sekcie prenášajúce fragmenty DownloadDataBlock: - všetky určené moduly, ktoré sa majú vyhľadať, musia mať číslo poslednej sekcie 0xFE; - správanie prijímača nie je definované, ak číslo poslednej sekcie = 0xFF.	

Maximálna dĺžka sekcie je 4 096 bajtov pri všetkých používaných typoch sekcií v objektových karuseloch. Záhlavie sekcie má 12 bajtov, takže v sekcii sa nachádza maximálne 4 084 bajtov užitočných dát.

B.2.1.1 Sekcie v rámci paketu TS

V jednom pakete TS sa môžu doručovať maximálne štyri sekcie.

B.2.2 Dátový karusel

Článok definuje obsah správ dátového karuselu v prípade používania objektového karuselu.

Používanie deskriptorov dátového karuselu, ktoré nie sú uvedené v objektovom karuseli DVB, nie je definované v tomto dokumente.

B.2.2.1 Všeobecne

Definície v tabuľke B.3 používajú obidve záhlavia dsmccDownloadDataHeader a rovnako aj dsmccMessageHeader.

Tabuľka B.3 – Obmedzenia DSM-CC DownloadData a hlavičky správ

Pole	Obmedzenia	Zdroj
TransactionId	Pozri článok B.2.5. "Priradovanie a používanie hodnôt transactionId"	Tento dokument
AdaptationLength	Koncové zariadenie môže ignorovať prípadné obsahy poľa dsmccAdaptationHeader.	

B.2.2.2 Zobrazenie informácií o sťahovaní

DownloadInfoIndication je správa, ktorá opisuje skupinu modulov a poskytuje potrebné parametre na umiestnenie modulu a znovu ho načíta.

Tabuľka B.4 – Obmedzenia DII

Pole	Obmedzenia	Zdroj
blockSize	maximálny počet 4 066 (max. užitočných dát sekcie – veľkosť záhlavia DDB (18)) Odporúčaná veľkosť blockSize je 4 066.	DSM-CC (definícia BlockSize), tento dokument (hodnota)
windowSize	0 (nepoužíva sa v objektových karuseloch)	DSM-CC
ackPeriod	0 (nepoužíva sa v objektových karuseloch)	DSM-CC
tCDownloadWindow	0 (nepoužíva sa v objektových karuseloch)	DSM-CC
tCDownloadScenario	0 (nepoužíva sa v objektových karuseloch)	DSM-CC
compatibilityDescriptor(): compatibilityDescriptorLength	0 (deskriptor nie je kompatibilný s objektovými karuselmi)	DSM-CC
PrivateDataLength	Koncové zariadenie môže ignorovať prípadné obsahy poľa privateData.	DVB

B.2.2.3 Inicializácia servera sťahovania

DownloadServerInitiate sa používa v prípade objektových karuselov na poskytnutie odkazu na objekt v ServiceGateway (t. j. hlavný adresár) objektového karuselu.

Tabuľka B.5 – Obmedzenia DSI

Pole	Obmedzenia	Zdroj
compatibilityDescriptor(): compatibilityDescriptorLength	0 (deskriptor nie je kompatibilný s objektovými karuselmi)	DSM-CC

privateData	Obsahuje štruktúru ServiceGatewayInfo	DSM-CC
serverId	Každých 20 bajtov sa musí nastaviť na hodnotu 0xFF	DVB/tento dokument

B.2.2.4 Modul informácií

Štruktúra moduleInfo sa nachádza v poli moduleInfo v DownloadInfoIndication dátového karuselu. Obsahuje informácie potrebné na vyhľadanie modulu.

Tabuľka B.6 – Obmedzenia poľa DII moduleInfo

Pole	Obmedzenia	Zdroj
BIOP::ModuleInfo::Taps	Prvý TAP musí "použiť" hodnotu 0x0017 (BIOP_OBJECT_USE). Polia id a selektora sa nepoužívajú a koncové zariadenie ich môže ignorovať. Koncové zariadenie môže ignorovať prípadné ďalšie TAP v zozname.	DVB
BIOP::ModuleInfo::UserInfo	Pole UserInfo obsahuje slučku deskriptorov. Tieto sú uvedené v norme Vysielanie DVB dát a/alebo v tomto dokumente. Koncové zariadenie musí podporovať compressed_module_descriptor (tag 0x09) používaný na signalizáciu, pretože modul je prenášaný v komprimovanej podobe. Pole UserInfo môže tiež obsahovať caching_priority_descriptor a jeden alebo viac label_descriptor.	DVB/tento dokument
moduleTimeOut blockTimeOut minBlockTime	Tieto polia sú definované v jednotkách µs. Príslušná hodnota sa musí výhradne kódovať karuselovým generovacím zariadením. Neexistuje žiadna predvolená hodnota, ktorá sa môže kódovať, t. j. 0xFFFFFFFF nemá žiadny zvláštny význam. Prijímače nesmú namiesto toho používať prednastavenú signalizovanú hodnotu, pretože neexistuje žiadny spôsob, ako ich definovať bez znalosti zostavy konkrétneho karuselu.	Tento dokument

Tabuľka B.7 – Syntax ModuleInfo BioP

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
BIOP::ModuleInfo() {				
moduleTimeOut	32	uimsbf	+	
blockTimeOut	32	uimsbf	+	
minBlockTime	32	uimsbf	+	
taps_count	8	uimsbf	N1	≥1
{				
id	16	uimsbf	0x0000	neverejný používateľ
use	16	uimsbf	0x0017	BIOP_OBJECT_USE
assocTag	16	uimsbf	+	
selector_length	8	uimsbf	0x00	
}				
for (j=1; j<N1; j++) {				Možné ďalšie TAP, ktoré sa môžu ignorovať koncovými zariadeniami.
id	16	uimsbf	+	
use	16	uimsbf	+	
assocTag	16	uimsbf	+	
selector_length	8	uimsbf	N2	
for (j=0; j<N2; j++) {				
selector_data	8	uimsbf	+	
}				
userInfoLength	8	uimsbf	N3	
for (k=0; k<N3; j++) {				
userInfo_data	8	uimsbf	+	
}				
}				

B.2.2.4.1 Deskriptor náveste

Tento článok v dokumente neobsahuje žiadny text.

B.2.2.4.2 Deskriptor priority kešovania

Označenie priorít objektov caching_priority_descriptor sa môže zahrnúť do poľa userInfo modulu moduleInfo v správe DownloadInfoIndication.

Tento deskriptor poskytuje hodnotu priority na kešovanie. Rovnaká priorita platí v každom objekte v module. Priorita uvedená v deskriptore je len pokyn koncovým zariadeniam a implementácie sa môžu používať v kombinácii s inými stratégiami kešovania.

Deskriptor obsahuje tiež transparentnú úroveň (pozri článok B.5.2 "Úrovně transparentného kešovania"), ktorú musí použiť koncové zariadenie pri kešovaní objektov v tomto module.

Tabuľka B.8 – Syntax deskriptora priority kešovania

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
caching_priority_descriptor() {				
descriptor_tag	8	uimsbf	0x71	
descriptor_length	8	uimsbf		
priority_value	8	uimsbf		
transparency_level	8	uimsbf		
}				

príznak deskriptora (angl. **descriptor_tag**): osembitová celočíselná hodnota s 0 x 71 identifikuje tento deskriptor

hodnota priority (angl. **priority_value**): označuje prioritu kešovania v objektoch v rámci tohto modulu; vyššia hodnota znamená väčší význam kešovania

úroveň transparentnosti (angl. **transparency_level**): koncové zariadenie musí používať transparentnú úroveň v prípade, keď sa v tomto module nachádzajú kešované objekty; možné hodnoty sú uvedené v tabuľke B.9; sémantiky postupov sú definované v článku B.5.2 "Úrovne transparentného kešovania"

Tabuľka B.9 – Hodnoty úrovni transparentnosti

Hodnota	Opis
0	rezervované
1	transparentné kešovanie
2	semitransparentné kešovanie
3	statické kešovanie
4 až 255	rezervované na budúce používanie

Ak nie je tento deskriptor zahrnutý v poli userInfo v moduleInfo v module, predpokladajú sa takéto predvolené hodnoty:

- priority_value: 128;
- transparency_level: 1 (transparentné kešovanie).

B.2.2.5 Informácie o službe sieťového prechodu

Štruktúra ServiceGatewayInfo sa prenáša v správe DownloadServerInitiate a poskytuje predmet odkazu na objekt ServiceGateway.

Tabuľka B.10 – Obmedzenia ServiceGatewayInfo

Pole	Obmedzenia	Zdroj
BIOP::ServiceGatewayInfo::downloadTaps	Koncové zariadenie môže ignorovať zoznam stiahnutých TAP.	Tento dokument
BIOP::ServiceGatewayInfo::serviceContextList	Koncové zariadenie môže ignorovať kontext zoznamu služby stiahnutých TAP.	
BIOP::ServiceGatewayInfo::userInfo	Koncové zariadenie môže ignorovať informácie používateľa.	

Tabuľka B.11 – Syntax ServiceGatewayInfo()

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
ServiceGatewayInfo(){				
IOP::IOR()			+	Pozri tabuľku B.21 "IOP::IOR syntax"
downloadTaps_count	8	uimsbf	N1	stiahnutý protokol TAP
for (i=0; i<N1; i++) {				
DSM::Tap()				
}				
serviceContextList_count	8	uimsbf	N2	serviceContextList
for (i=0; i<N2; i++) {				
context_id	32	uimsbf		
context_data_length	16	uimsbf	N3	
for (j=0; j<N3; j++) {				
context_data_byte	8	uimsbf	+	
}				
}				
userInfoLength	16	uimsbf	N5	informácia pre používateľa
for (i=0; i<N5; i++) {				
userInfo_data	8	uimsbf	+	
}				
}				

B.2.2.6 Zrušenie sťahovania

Neexistuje žiadna sémantika tejto správy v tomto profile. Prijímače ich môžu ignorovať.

B.2.2.7 Blok sťahovania dát**Tabuľka B.12 – Obmedzenia DDB**

Pole	Obmedzenia	Zdroj
moduleId	Id modulov sú jedinečné v rámci objektového karuselu; pozri ISO/IEC 13818-6 [4], článok 11.2.3.	DSM-CC

B.2.3 Objektový karusel**B.2.3.1 Správa generického objektu BIOP**

Správa generického objektu BIOP je spoločná štruktúra, ktorú používajú všetky správy BIOP.

Tabuľka B.13 – Obmedzenia správy generického objektu BIOP

Pole	Obmedzenia	Zdroj
MessageHeader::byte_order	0 (označuje konečné poradie bajtov)	DVB
MessageSubHeader::objectKey	Maximálna dĺžka kľúča je štyri bajty.	DVB
MessageSubHeader::objectKind	Musia sa používať krátke trojpísmenové iné názvy a ukončenie nulami.	DVB
Access attributes Prístupové atribúty	Prístupové atribúty nie sú prenášané v objektovom karuseli.	DSM-CC

B.2.3.2 Reťazce CORBA

V mnohých prípadoch správy objektového karuselu obsahujú textové reťazce. Tie sú formátované v súlade s článkom 12.3.2 CORBA/IIOP [9] a pomocou tzv. kódovania "CDR-Lite", ako je opísané v ISO/IEC 13818-6 [4], článok 5.6.3.4. To znamená, že text je uvádzaný ako celé číslo určujúce dĺžku reťazca s následným zakončením nulami. Veľkosť celého čísla závisí od príslušného reťazca, čo jasne vidieť v nasledujúcich syntaktických tabuľkách. Reťazce formátu CORBA a veľkosť dĺžky ich polí sú uvedené v tabuľke B.14.

Tabuľka B.14 – Umiestnenie reťazcov formátu CORBA

Reťazec	Veľkosť dĺžky poľa (bity)	Umiestnenie
objectKind_data	32	Tabuľka B.16 "BIOP::FileMessage syntax"
objectKind_data	32	Tabuľka B.19 "BIOP::DirectoryMessage syntax"
id_data	8	
kind_data	8	
objectKind_data	32	Tabuľka B.28 "BIOP::StreamMessage syntax"
objectKind_data	32	Tabuľka B. 30 "BIOP::StreamEventMessage syntax"
eventName_data	8	
type_id_byte	32	Tabuľka B.21 "IOP::IOR syntax"
id_data	32	Tabuľka B.25 "Syntax Lite Options Profile Body so zložkou ServiceLocation"
kind_data	32	

B.2.3.3 Súbor správy BIOP

BIOP FileMessage sa používa na prenos súboru objektov.

Tabuľka B.15 – Obmedzenia súboru správy BIOP

Pole	Obmedzenia	Zdroj
MessageSubHeader::ObjectInfo	ObjectInfo môže byť prázdna (má nulovú dĺžku). Ak nie je prázdna, potom prvých 8 bajtov ObjectInfo musí obsahovať atribút DSM::File::ContentSize. Voliteľne môže nasledovať slučka deskriptorov. Deskriptor definovaný na možné použitie v tejto časti je: deskriptor typu obsahu.	Tento dokument
MessageSubHeader::ServiceContextList	Koncové zariadenie môže preskočiť prípadné štruktúry serviceContextList.	

Tabuľka B.16 – Syntax FileMessage BIOP

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
BIOP::FileMessage() {				
magic	4 x 8	uimsbf	0x42494F50	"BIOP"
biop_version.major	8	uimsbf	0x01	hlavná verzia 1 BIOP
biop_version.minor	8	uimsbf	0x00	vedľajšia verzia 0 BIOP
byte_order	8	uimsbf	0x00	poradie bajtov big-endian
message_type	8	uimsbf	0x00	
message_size	32	uimsbf	*	
objectKey_length	8	uimsbf	N1	1 to 4
for (i=0; i<N1; i++) {				
objectKey_data	8	uimsbf	+	
}				
objectKind_length	32	uimsbf	0x00000004	
objectKind_data	4 x 8	uimsbf	0x66696C00	"fil" type_id alias
objectInfo_length	16	uimsbf	N2	
DSM::File::ContentSize	64	uimsbf	+	objectInfo (poznámka)
for (i=0; i<N2 - 8; i++) {				

descriptor()	8	uimsbf	+	
}				
serviceContextList count	8	uimsbf	N3	serviceContextList
for (i=0; i<N3; i++) {				
context_id	32	uimsbf		
context_data length	16	uimsbf	N4	
for (j=0; j<N4; j++) {				
context_data byte	8	uimsbf	+	
}				
}				
messageBody_length	32	uimsbf	*	
content_length	32	uimsbf	N5	
for (i=0; i<N5; i++) {				
content_byte	8	uimsbf	+	aktuálny obsah súboru
}				
}				
POZNÁMKA. – Ak je prítomný a nenulový, tak musí byť rovnaký ako content_length z odkazovaného FileMessage.				

B.2.3.4 Deskriptor typu obsahu

V súbore MessageSubHeader:ObjectInfo alebo BioP::Binding: ObjectInfo sa môže prenášať jeden deskriptor content_type_descriptor alebo žiadny deskriptor. V prípade, keď sa používa viac ako jeden deskriptor content_type_descriptor, deskriptory musia mať rovnaký formát obsahu. Taktiež typ obsahu, ak existuje, je signalizovaný v povinnom adresári a musí sa zhodovať so signalizovanou hlavičkou súvisiaceho súboru. Tento voliteľný deskriptor identifikuje typ média súboru.

Signalizácia tohto typu obsahu sa vzťahuje iba na objekty typu súboru a nie je vhodná na iné typy objektov.

Formát deskriptora content_type_descriptor je uvedený v tabuľke B.17.

Tabuľka B.17 – Syntax deskriptora typu obsahu

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
content_type_descriptor() {				
descriptor_tag	8	uimsbf	0x72	
descriptor_length	8	uimsbf		
for (i=0; i<descriptor_length; i++) {				
content_type_data_byte	8	uimsbf		A MIME type
}				
}				

príznak deskriptora (angl. **descriptor_tag**): osembitové celé číslo s hodnotou 0 x 72 identifikuje tento deskriptor

dĺžka deskriptora (angl. **descriptor_length**): osembitové celé číslo udáva počet bajtov, ktoré za ním nasledujú

bajt typu dátového obsahu (angl. **content_type_data_byte**): bajty tvoria reťazec, ktorý označuje typ obsahu objektu MIME; reťazec je špecifikovaný takto:

content_type_data = type "/" subtype *(";" parameter).

Ak sú typ, subtyp a parameter definované podľa článku 5 v RFC 2045 [10], potom content_type_data prenáša užitočné dáta v hlavičke Content-Type definovanej v RFC 2045 [10].

B.2.3.5 Adresár správy BIOP

BioP DirectoryMessage sa používa na prenos adresára objektov.

Tabuľka B.18 – Obmedzenia na adresár správy BioP

Pole	Obmedzenia	Zdroj
MessageSubHeader::ObjectInfo	Koncové zariadenie môže preskočiť N2 možných bajtov v poli objectInfo.	Tento dokument
MessageSubHeader::ServiceContextList	Koncové zariadenie môže preskočiť N3 možných štruktúr serviceContextList.	Tento dokument
BIOP::Name	Názov musí obsahovať presne jednu zložku NameComponent. Id_length musí byť 2 alebo väčšie. Id_data sa nesmie opakovať v ďalších názvoch zložiek v tomto adresári.	Tento dokument
BIOP::Binding::BindingType	"ncontext" (v prípade objektového adresára) alebo "nobject" (v prípade súboru alebo toku objektu). Povinný typ "composite" sa nesmie používať.	DVB
BIOP::Binding::ObjectInfo	ObjectInfo súvisiacich objektov môže byť prázdna (má nulovú dĺžku). Ak súvisiaci objekt je súbor a ObjectInfo nie je prázdna, potom prvých 8 bajtov ObjectInfo musí obsahovať atribút ContentSize. Voliteľne môže nasledovať slučka deskriptorov. Deskriptor definovaný na možné použitie v tejto časti je: deskriptor typu obsahu.	Tento dokument

Tabuľka B.19 – Syntax DirectoryMessage BIOP

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
BIOP::DirectoryMessage() {				
magic	4 x 8	uimsbf	0x42494F50	"BIOP"
biop_version.major	8	uimsbf	0x01	BIOP major version 1
biop_version.minor	8	uimsbf	0x00	BIOP minor version 0
byte_order	8	uimsbf	0x00	big endian byte ordering
message_type	8	uimsbf	0x00	
message_size	32	uimsbf	*	
objectKey_length	8	uimsbf	N1	1 to 4
for (i=0; i<N1; i++) {				
objectKey_data	8	uimsbf	+	
}				
objectKind_length	32	uimsbf	0x00000004	
objectKind_data	4 x 8	uimsbf	0x64697200	"dir" type_id alias
objectInfo_length	16	uimsbf	N2 = 0 (note)	objectInfo
for (i=0; i<N2; i++) {				
objectInfo_data	8	uimsbf	+	
}				
serviceContextList_count	8	uimsbf	N3	serviceContextList
for (i=0; i<N3; i++) {				
context_id	32	uimsbf		
context_data_length	16	uimsbf	N4	
for (j=0; j<N4; j++) {				
context_data_byte	8	uimsbf	+	
}				
}				
messageBody_length	32	uimsbf	*	
bindings_count	16	uimsbf	N5	
for (i=0; i<N5; i++) {				Binding
BIOP::Name() {				
nameComponents_count	8	uimsbf	N6 = 1	See Table B.16.
for (i=0; i<N6; i++) {				
id_length	8	uimsbf	N7	NameComponent id
for (j=0; j<N7; j++) {				
id_data	8	uimsbf	+	The "/" character shall not be used.
}				
kind_length	8	uimsbf	N8	NameComponent kind
for (j=0; j<N8; j++) {				
kind_data	8	uimsbf	+	as type_id (see Table 4-4 in TR 101 202 [i.2])
}				
}				
}				
BindingType	8	uimsbf	+	0x01 for nobject 0x02 for ncontext
IOP::IOR()			+	objectRef see table B.21
objectInfo_length	16	uimsbf	N9	
if(kind_data == "fil"){				
DSM::File::ContentSize	64	uimsbf	+	0 means that file size is not signalled
for (j=0; j<N9 - 8; j++) {				
descriptor_byte	8	uimsbf	+	
}				
}				
else {				
for (j=0; j<N9; j++) {				
descriptor_byte	8	uimsbf	+	
}				
}				
}				
}				
POZNÁMKA. – Pozri bod 2 v článku 11.3.2.2 "Formát adresára správy" v špecifikácii DSM-CC [4]: "pole objectInfo musí byť				

prázdne".

B.2.3.6 Správa služby sieťového priechodu

Syntax správy BIOP ServiceGateway je zhodná s BIOP DirectoryMessage (pozri skôr) s nasledujúcimi výnimkami:

- typ objektu je skôr "srg" ako "dir";
- používa sa kontextový zoznam služby.

B.2.3.7 Interoperabilné odkazy na objekt BIOP

Interoperabilné odkazy na objekt (IOR) sú odkazy na objekty a obsahujú informácie potrebné na vyhľadanie objektu. Štruktúra IOR môže obsahovať rôzne možnosti smerovania na objekty, ktoré sa môžu sprístupniť prostredníctvom rôznych typov pripojení. V tomto dokumente je použitie odkazov IOR obmedzené na odkazy objektov prenášaných vysielaním objektovými karuselmi. V objektových karuseloch existujú dva typy odkazov na objekt: jeden sa používa na odkazovanie objektov prenášaných v tom istom objektovom karuseli a druhý sa používa na odkazovanie objektov v iných objektových karuseloch.

Tabuľka B.20 – Obmedzenia IOR na BIOP

Pole	Obmedzenia	Zdroj
IOP::IOR::type_id	Obsahuje objectKind z odkazovaného objektu. Musia sa používať krátke trojpísmenové názvy a ukončenie nulami.	Tento dokument
IOP::IOR::taggedProfileList	V IOR sa musí zahrnúť aspoň jedným taggedProfile. Pri objektoch prenášaných vysielaním cez objektový karusel ako prvý taggedProfile musí byť profil TAG_BIOP alebo TAG_LITE_OPTIONS. Ak prvý označený profil je nejaký iný profil, objekt sa neprenáša vysielaním cez objektový karusel a koncové zariadenie môže ignorovať predmet objektu a riadi sa vlastnými schopnosťami.	Tento dokument

Tabuľka B.21 – Syntax IOR – IOP

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
IOP::IOR {				
type_id_length	32	uimsbf	N1	
for (i=0; i<N1; i++) {				
type_id_byte	8	uimsbf	+	Krátky názov type_id (napr. "dir")
}				
taggedProfiles_count	32	uimsbf	N2	Základné časti profilu
IOP::taggedProfile()				Objekty vysielané karuselmi: BIOPProfileBody alebo LiteOptionsProfileBody
for (n=0; n<N2 - 1;n++) {				
IOP::taggedProfile()				Koncové zariadenie môže ignorovať iné profily (2...N1), ak sú prítomné
}				
}				

B.2.3.7.1 Telo profilu BIOP

BiopProfileBody sa používa v odkazoch na objekty v rámci toho istého objektového karuselu.

Tabuľka B.22 – Obmedzenia tela profilu BIOP

Pole	Obmedzenia	Zdroj
BiopProfileBody::byte_order	0 (indikuje konečné radenie bajtu)	DVB
BiopProfileBody::LiteComponent	Tento zoznam musí obsahovať práve jeden BiopObjectLocation a práve jeden DSM::ConnBinder ako prvé dve zložky v uvedenom poradí. Koncové zariadenie môže ignorovať prípadné ďalšie zložky v tomto zozname.	Tento dokument
DSM::ConnBinder	Pri objektoch prenášaných vo vysielaní objektovým karuselom musí byť ako prvý TAP typu BIOP_DELIVERY_PARA_USE. Ak existuje iný typ TAP na prvej pozícii, môže koncové zariadenie ignorovať tento odkaz na objekt, pretože je to odkaz na objekt prístupný pomocou iného typu protokolu (napríklad na použitie spätného kanála). Koncové zariadenie môže ignorovať možné ďalšie TAP v zozname.	Tento dokument
DSM::Tap	V TAP BIOP_DELIVERY_PARA_USE sa id poľa nepoužíva a sa môže ignorovať koncovým zariadením.	Tento dokument
DSM::Tap::timeout	Toto pole je definované v jednotkách μ s. Príslušná hodnota sa musí výhradne kódovať karuselovým generovacím zariadením. Neexistuje žiadna predvolená hodnota, ktorá sa môže zakódovať, t. j. 0xFFFFFFFF nemá žiadny zvláštny význam. Prijímače nesmú namiesto toho používať prednastavenú signalizovanú hodnotu, pretože neexistuje žiadny spôsob, ako ich definovať bez znalosti zostavy konkrétneho karuselu.	Tento dokument

Tabuľka B.23 – Syntax tela profilu BIOP

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
BIOPProfileBody {				
profileId_tag	32	uimsbf	0x49534F06	TAG_BIOP (BIOP Profile Body)
profile_data_length	32	uimsbf	*	
profile_data_byte_order	8	uimsbf	0x00	konečné radenie bajtu
lite_component_count	8	uimsbf	N1	
BIOP::ObjectLocation {				
componentId_tag	32	uimsbf	0x49534F50	TAG_ObjectLocation
component_data_length	8	uimsbf	*	
carouselId	32	uimsbf	+	
moduleId	16	uimsbf	+	
version.major	8	uimsbf	0x01	Hlavná verzia 1 protokolu BIOP
version.minor	8	uimsbf	0x00	Vedľajšia verzia 0 protokolu BIOP
objectKey_length	8	uimsbf	N2	1 až 4
for (k=0; k<N2; k++) {				
objectKey_data	8	uimsbf	+	
}				
}				
DSM::ConnBinder {				
componentId_tag	32	uimsbf	0x49534F40	TAG_ConnBinder
component_data_length	8	uimsbf	N4	
taps_count	8	uimsbf	N3	
DSM::Tap {				
id	16	uimsbf	0x0000	neverejný používateľ
use	16	uimsbf	0x0016	Ak je prítomný BIOP_DELIVERY_PARAMETER_USE, tak musí byť ako prvý TAP. Ak existuje iný typ TAP na prvej pozícii, môže koncové zariadenie ignorovať tento odkaz na objekt, pretože je to odkaz na objekt prístupný pomocou iného typu protokolu (napríklad na použitie spätného kanála).
assocTag	16	uimsbf	+	
selector_length	8	uimsbf	0x0A	
selector_type	16	uimsbf	0x0001	
transactionId	32	uimsbf	*	
timeout	32	uimsbf	*	
}				
for (n=0; n<N4 - 18; n++) {				Koncové zariadenie môže preskočiť na ďalšie možné TAP.
additional_tap_byte	8	uimsbf		
}				
}				
for (n=0; n<N6; n++) {			N6=N1 - 2	
BIOP::LiteComponent{				
componentId_tag	32	uimsbf	+	
component_data_length	8	uimsbf	N7	
for (i=0; i<N7; i++) {				
component_data_byte	8	uimsbf		
}				
}				
}				
}				

B.2.3.7.2 Telo profilu redukovaných parametrov

Profil LiteOptionsProfileBody sa používa na tvorbu odkazov pri objektoch prenášaných v iných objektových karuseloch a na vytváranie odkazov na objekty prenášané v iných karuseloch v rámci tých istých prenosových tokov alebo v iných prenosových tokoch. Na používanie profilu LiteOptionsProfileBody sa aplikujú nasledujúce obmedzenia:

- odkazy LiteOptionsProfileBody sa nesmú nikdy používať v IOR, ktoré sú v odkazujúcom sieťovom priechode služby DSI;
- cieľový karusel sa nesmie nikdy zostavovať automaticky;
- špecifikácia platformy môže stanoviť pravidlá v prípade vyskytnutia sa profilu LiteOptionsProfileBody.

Tabuľka B.24 – Obmedzenia tela profilu redukovaných parametrov

Pole	Obmedzenia	Zdroj
LiteOptionsProfileBody::profile_data_byte_order	0 (indikuje konečné radenie bajtu)	DVB
LiteOptionsProfileBody::LiteOptionsComponents	Zoznam musí obsahovať zložku ServiceLocation ako prvú zložku. Konečné zariadenie môže ignorovať možné ďalšie zložky v zozname.	Tento dokument
DSM::ServiceLocation	Pri objektoch prenášaných vo vysielaní objektovým karuselom musí mať adresa domény služby NSAP formát adresy Carousel NSAP.	Tento dokument
DSM::ServiceLocation::InitialContext	Konečné zariadenie môže ignorovať kontext inicializácie.	Tento dokument

Tabuľka B.25 – Syntax tela profilu s redukovanými parametrami so zložkou ServiceLoc

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
LiteOptionsProfileBody {				
profileId_tag	32	uimsbf	0x49534F05	TAG_LITE_OPTIONS) (telo profilu s redukovanými parametrami)
profile_data_length	32	uimsbf	*	
profile_data_byte_order	8	uimsbf	0x00	konečné radenie bajtu
lite_component_count	8	uimsbf	N1	
DSM::ServiceLocation {				
componentId_tag	32	uimsbf	0x49534F46	TAG_ServiceLocation
component_data_length	8	uimsbf	*	
serviceDomain_length	8	uimsbf	0x14	Dĺžka adresy karuselu NSAP
serviceDomain_data()	160	uimsbf	+	Pozri tabuľku B.26
CosNaming::Name() {				pathName
nameComponents_count	32	uimsbf	N2	
for (i=0; i<N2; i++) {				
id_length	32	uimsbf	N3	NameComponent id
for (j=0; j<N3 j++) {				
id_data	8	uimsbf	+	
}				
kind_length	32	uimsbf	N4	Typ NameComponent
for (j=0; j<N4 j++) {				
kind_data	8	uimsbf	+	ako type_id (pozri tabuľku 4.4 v TR 101 202 [i.2])
}				
}				
}				
initialContext_length	32	uimsbf	N5	

for (n=0; n<N5 n++) {				
InitialContext_data_byte	8	uimsbf		
}				
}				
for (n=0;n<N6;n++) {			N6=N1-1	
BIOP::LiteComponent{				
componentId_tag	32	uimsbf	+	
component_data_length	8	uimsbf	N7	
for (i=0; i<N7; i++) {				
component_data_byte	8	uimsbf		
}				
}				
}				
}				

Tabuľka B.26 – Adresa karuselu NSAP DVB

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
DVBcarouselNSAPaddress {				
AFI	8	uimsbf	0x00	NSAP na neverejné použitie
type	8	uimsbf	0x00	Adresa objektového karuselu NSAP
carouselId	32	uimsbf	+	Ak v poli carousel_identifier_deskriptor je použitý carousel_id s rovnakým id, potom sa musí použiť signalizácia v PMT na službu uvedenú neskôr.
specifierType	8	uimsbf	0x01	IEEE OUI
specifierData { IEEE OUI }	24	uimsbf	0x00015A	Konštanta DVB OUI
dvb_service_location () {				
transport_stream_id	16	uimsbf	+	Toto sa môže nastaviť na 0x0000, čo znamená, že koncové zariadenie nesmie používať pri vyhľadávaní služby transport_stream_id. Toto pole sa používa pri všetkých ostatných hodnotách.
original_network_id	16	uimsbf	+	
service_id	16	uimsbf	+	(= MPEG-2 program_number)
reserved	32	bslbf	0xFFFFFFFF	
}				
}				

B.2.3.8 Tok správy BIOP**Tabuľka B.27 – Obmedzenia na tok správy BIOP**

Pole	Obmedzenia	Zdroj
MessageSubHeader::ObjectInfo	Pole ObjectInfo obsahuje štruktúru DSM::Stream::Info_T a prípadne ďalšie údaje po štruktúre Stream Info. Koncové zariadenia môžu ignorovať Description_bytes v štruktúre DSM::Stream::Info_T a možné ďalšie info údaje objektu nasledujúce po štruktúre. Vysielatelia môžu nastaviť dĺžku poľa na nulu na označenie nedefinovaného trvania.	Tento dokument
MessageSubHeader::ServiceContextList	Koncové zariadenie môže preskočiť možné štruktúry serviceContextList.	Tento dokument
MessageSubHeader::MessageBody	Telo správy MessageBody prenáša postupnosť TAP. V nej musí byť najviac jeden TAP obsahujúci BIOP_PROGRAM_USE. Tento TAP identifikuje službu, ktorá poskytuje mediálny tok súvisiaci s tokom objektu (cez deferred_association_tags_descriptor v PMT). TAP môže odkazovať len na programy, ktoré sú vysielané v rovnakom multiplexe (t. j. koncové zariadenia sa nesmú preladiť na iný multiplex s cieľom získať odkazovaný mediálny tok). V správe sa môže nachádzať najviac jeden TAP obsahujúci STR_NPT_USE alebo STR_DVBTIMEL_USE, ktorý indikuje časovú základňu súvisiacu s tokom objektu. TAP s použitím STR_DVBTIMEL_USE sa interpretujú v súlade s článkom B.2.3.10. Koncové zariadenia môžu ignorovať prípadné ďalšie TAP (napríklad BIOP_ES_USE).	Tento dokument
POZNÁMKA. – Použitie NPT je zastarané a nie je definované v tomto dokumente.		

Tabuľka B.28 – Syntax StreamMessage BioP

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
BIOP::StreamMessage() {				
magic	4 x 8	uimsbf	0x42494F50	"BIOP"
biop_version.major	8	uimsbf	0x01	vyššia verzia 1 BIOP
biop_version.minor	8	uimsbf	0x00	nižšia verzia 0 BIOP
byte_order	8	uimsbf	0x00	big endian byte ordering
message_type	8	uimsbf	0x00	
message_size	32	uimsbf	*	
objectKey_length	8	uimsbf	N1	1 to 4
for (i=0; i<N1; i++) {				
objectKey_data	8	uimsbf	+	
}				
objectKind_length	32	uimsbf	0x00000004	
objectKind_data	8	uimsbf	0x73747200	"str" type_id alias
objectInfo_length	16	uimsbf	N2	
DSM::Stream::Info_T {				objectInfo
aDescription_length	8	uimsbf	N3	aDescription
for (i=0; i<N3; i++) {				
aDescription_bytes	8	uimsbf	+	
}				
duration.aSeconds	32	simsbf	+	ak je nastavené na 0 indikuje nedefinované pole
duration.aMicroSeconds	32	uimsbf	+	ak je nastavené na 0 indikuje nedefinované pole
audio	8	uimsbf	+	Flag: 0x00 = false, non-zero = true.
video	8	uimsbf	+	Flag: 0x00 = false, non-zero = true.
data	8	uimsbf	+	Flag: 0x00 = false, non-zero = true.
}				
for (i=0; i<N2-(N3+10); i++) {				
objectInfo_byte	8	uimsbf	+	
}				
serviceContextList_count	8	uimsbf	N4	serviceContextList
for (i=0; i<N4; i++) {				
context_id	32	uimsbf		
context_data_length	16	uimsbf	N5	
for (j=0; j<N5; j++) {				
context_data_byte	8	uimsbf	+	
}				
}				
messageBody_length	32	uimsbf	*	
taps_count	8	uimsbf	N6	
for (i=0; i<N6; i++) {				
id	16	uimsbf	(note)	pozri článok B.2.4.4
use	16	uimsbf	+	pozri článok B.2.3.10 a tabuľku 4.12 in DVB Guidelines for Data Broadcasting [i.2]
assocTag	16	uimsbf	+	
selector_length	8	uimsbf	0x00	no selector
}				
}				

POZNÁMKA. – Ak TAP obsahuje STR_DVBTIMEL_USE, potom hodnota tohto šestnásťbitového celého čísla zodpovedá hodnote poľa časového prehľadu DVB broadcast_timeline_id, ktorý definuje časovú základňu tohto toku. Hodnota poľa nie je definovaná pri iných hodnotách TAP.

B.2.3.9 Tok správy o udalosti BIOP

Tabuľka B.29 – Obmedzenia na tok správy o udalosti BIOP

Pole	Obmedzenia	Zdroj
MessageSubHeader::ObjectInfo	Pole ObjectInfo obsahuje štruktúry DSM::Stream::Info_T a DSM::Stream::EventList_T nasledujúce voliteľne ďalšími info údajmi objektu, ktoré sa môžu ignorovať koncovými zariadeniami (pozri tabuľku B.27 ohľadne DSM::Stream::Info_T). Koncové zariadenia môžu ignorovať možné ďalšie údaje, ktoré nasledujú po DSM::Stream::EventList_T. EventList_T definuje postupnosť názvov udalostí, ktoré korelujú s postupnosťou identifikátorov id udalosti v MessageBody. eventNames_count sa musí rovnať eventIds_count.	Tento dokument
MessageSubHeader::ServiceContextList	Koncové zariadenie môže preskočiť možné štruktúry serviceContextList.	Tento dokument
MessageSubHeader::MessageBody	MessageBody prenáša postupnosť TAP, po ktorých nasleduje postupnosť identifikátorov id udalosti. Postupnosť TAP sa riadi týmito pravidlami: <ul style="list-style-type: none"> • Musí byť najviac jeden TAP, ktorý používa BIOP_PROGRAM_USE. Tento TAP identifikuje službu, ktorá poskytuje mediálny tok súvisiaci s tokom objektu (pomocou deferred_association_tags_descriptor v PMT). TAP môže odkazovať len na programy, ktoré sú vysielané v tom istom multiplexe (t. j. koncové zariadenia sa nesmú naladiť na iný multiplex s cieľom získať odkazované mediálne toky). • Musí existovať najviac jeden TAP obsahujúci STR_NPT_USE alebo STR_DVBTIMEL_USE, ktorý indikuje časovú základňu, ktorá súvisí s objektom StreamEvent. TAP používajúce STR_NPT_USE sa musia interpretovať v súlade s ISO/IEC 13818-6 [4]. TAP využívajúce STR_DVBTIMEL_USE sa musia interpretovať v súlade s bodom B.2.3.10. • Musí existovať najviac jeden TAP používajúci STR_EVENT_USE STR_STATUS_AND_EVENT_USE alebo STR_DVBEVENT_USE. Tento TAP indikuje PID, kde vysielané dáta udalosti sa vzťahujú na objekt StreamEvent. Koncové zariadenia môžu ignorovať možné ďalšie TAP (napríklad BIOP_ES_USE). 	Tento dokument
Poznámka. – Použitie NPT je zastarané a nie je definované v tomto dokumente.		

Tabuľka B. 30 – Syntax StreamEventMessage BioP

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
BIOP::StreamEventMessage() {				
magic	4 x 8	uimsbf	0x42494F50	"BIOP"
biop_version.major	8	uimsbf	0x01	vyššia verzia 1 BIOP
biop_version.minor	8	uimsbf	0x00	nižšia verzia 0 BIOP
byte_order	8	uimsbf	0x00	big endian byte ordering
message_type	8	uimsbf	0x00	
message_size	32	uimsbf	*	
objectKey_length	8	uimsbf	N1	
for (i=0; i<N1; i++) {				
objectKey_data	8	uimsbf	+	
}				
objectKind_length	32	uimsbf	0x00000004	
objectKind_data	4 x 8	uimsbf	0x73746550	"ste" type_id alias
objectInfo_length	16	uimsbf	N2	
DSM::Stream::Info_T {				
aDescription_length	8	uimsbf	N3	aDescription
for (i=0; i<N3; i++) {				
aDescription_bytes	8	uimsbf	+	pozri tok správy BIOP
}				
duration.aSeconds	32	simsbf	+	pozri tok správy BIOP
duration.aMicroSeconds	32	uimsbf	+	pozri tok správy BIOP
audio	8	uimsbf	+	pozri tok správy BIOP
video	8	uimsbf	+	pozri tok správy BIOP
data	8	uimsbf	+	pozri tok správy BIOP
}				
DSM::Event::EventList_T {				
eventNames_count	16	uimsbf	N4	
for (i=0; i<N4; i++) {				
eventName_length	8	uimsbf	N5	
for (j=0; j<N5; j++) {				
eventName_data	8	uimsbf	+	(including zero terminator)
}				
}				
for (i=0; i<N2 - (N3 + 14 + N4 + sum(N5)); i++) {				
objectInfo_byte	8	uimsbf	+	
}				
serviceContextList_count	8	uimsbf	N6	
for (i=0; i<N6; i++) {				
context_id	32	uimsbf		
context_data_length	16	uimsbf	N7	
for (j=0; j<N7; j++) {				
context_data_byte	8	uimsbf	+	
}				
}				
messageBody_length	32	uimsbf	*	
taps_count	8	uimsbf	N8	
for (i=0; i<N8; i++) {				
id	16	uimsbf	(note)	pozri B.2.4.4
use	16	uimsbf	+	pozri článok B.2.3.10 a tabuľku 4.12 v DVB Guidelines for Data Broadcasting [i.2]
assocTag	16	uimsbf	+	
selector_length	8	uimsbf	0x00	no selector
}				
eventIds_count	8	uimsbf	N4	(= eventNames_count)
for (i=0; i<N4; i++) {				
eventId	16	uimsbf	+	
}				
}				

Syntax	Počet bitov	Identifikátor	Hodnota	Komentár
POZNÁMKA. – Ak TAP obsahuje STR_DVBTIMEL_USE, potom hodnota tohto šesťnásťbitového celého čísla zodpovedá hodnote poľa časového prehľadu broadcast_timeline_id, ktorý definuje časovú základňu tohto toku. Ak TAP obsahuje STR_DVBEVENT_USE, potom hodnota tohto šesťnásťbitového poľa zodpovedá hodnote synchronised_event_context všetkých udalostí týkajúcich sa tohto toku. Hodnota poľa nie je definovaná pri iných hodnotách TAP.				

B.2.3.10 Ďalšie hodnoty TAPUse

Definované sú ďalšie dve hodnoty TAPUse, ktoré sa používajú na odkazovanie synchronizovaných pomocných dát v rámci DVB TS 102 823 [13] z objektového karuselu v súlade s tabuľkou B.31.

Tabuľka B.31 – Hodnoty TAPUse na odkazovanie na synchronizované pomocné dáta DVB

TAPUse	Hodnota
STR_DVBTIMEL_USE	0x8000
STR_DVBEVENT_USE	0x8001

Sémantika polí TAP ukazujúca na broadcast_timeline_descriptor DVB je opísaná takýmto spôsobom:

- použitie poľa indikuje použitie TAP; hodnota tohto poľa musí byť STR_DVBTIMEL_USE;
- hodnota poľa identifikátora musí špecifikovať broadcast_timeline_id časového plánu, na ktorý sa odkazuje;
- príznak assocTag identifikuje pripojenie, na ktorom sa vysiela broadcast_timeline_descriptor;
- pole selektora musí byť prázdne.

Sémantika polí TAP ukazujúca na deskriptor synchronised_event_descriptor DVB je opísaná takto:

- použitie poľa znamená použitie TAP; hodnota tohto poľa musí byť STR_DVBEVENT_USE;
- hodnota poľa identifikátora musí špecifikovať synchronised_event_context udalostí, na ktoré sa odkazuje;
- príznak assocTag identifikuje pripojenie, na ktorom sú vysielať deskriptory synchronised_event_descriptor;
- pole selektora musí byť prázdne.

B.2.4 Vysielanie časových základní a udalostí

Koncové zariadenia môžu podporovať časový prehľad DVB na poskytovanie vysielania časovej základne.

Koncové zariadenia sú povinné podporovať aj vysielanie udalostí takto:

- tok naplánovaných udalostí DSM-CC; tie sú s odkazom na vysielanie časovej základne definovanej pomocou DSM-CC NPT alebo mechanizmu časového prehľadu DVB;
- tok udalostí DSM-CC "urob to teraz"; ide o samostatné udalosti, ktoré sú nezávislé od vysielania časovej základne;
- synchronizované udalosti DVB; ide o samostatné udalosti, ktoré sú nezávislé od vysielania časovej základne, ale ktoré môžu poskytnúť oveľa väčšiu časovú presnosť, ako udalosti DSM-CC "urob to teraz".

Správy BioP StreamMessage a StreamEventMessage sú používané v rámci objektového karuselu DSM-CC na odkazovanie na časové základne a definovanie udalostí. Bio StreamMessage sa môže použiť len ako odkaz na vysielanie časovej základne. BioP StreamEventMessage sa môže použiť na definovanie vysielania udalostí s odkazom, alebo bez ohľadu na odkaz na vysielanie časovej základne.

POZNÁMKA. – Je známe, že mechanizmus NPT a plánované udalosti toku, ktoré od neho závisia, sú v mnohých digitálnych televíznych distribučných sieťach náchylnejšie na poruchy. Je nepravdepodobné, že existujúce využívané sieťové zariadenie, ktoré regeneruje STC, je informované o NPT, a z tohto dôvodu nebude vytvárať potrebnú zodpovedajúcu úpravu hodnôt STC vnútri odkazovaných deskriptorov NPT. Toto môže zapríčiniť, že plánované udalosti toku môžu pôsobiť proti spusteniu NPT v nesprávnom čase, alebo sa nikdy nespustia. Aplikácie sa majú používať len v plánovaných udalostiach tokov s NPT, keď sú presvedčené, že sieť, v ktorej sa majú použiť, nemá takýto problém. Časový prehľad DVB, udalosti DSM-CC "urob to teraz" a udalosti prenášané v rámci synchronizovaných pomocných dát DVB ponúkajú spoľahlivejšie alternatívy oproti NPT.

POZNÁMKA. – Použitie NPT je zastarané a nie je definované v tomto dokumente.

B.2.4.1 Tok a správy toku udalosti

B.2.4.1.1 Súvislosti s časovými základňami

Identifikátor poľa TAP STR_DVBTIMEL_USE správy StreamMessage alebo StreamEventMessage identifikuje časovú základňu pridruženú k toku objektu Stream/StreamEvent. Môžu sa používať súčasne viaceré správy StreamMessage alebo StreamEventMessage s povolenými viacerými časovými základňami tej istej služby (pozri článok B.2.4.4 "Vysielanie časových základní").

B.2.4.1.2 Názvy udalostí a identifikátory udalostí

V správach o udalosti toku StreamEventMessages zoznamu EventList_T je definovaná postupnosť názvov udalosti, ktorá koreluje 1 : 1 ako postupnosť identifikátorov ID udalosti v MessageBody. V každej správe BioP:StreamEventMessage sú názvy udalosti jednoznačne pridružené k hodnotám identifikátora udalosti:

- eventNames_count musia byť rovnaké ako eventIds_count;
- názvy v EventList_T sú reťazce ukončené nulami;
- hodnoty eventID v StreamEventMessage zodpovedajú hodnotám eventID prenášaným v StreamEventDescriptors alebo hodnotám synchronised_event_id v deskriptoroch synchronised_event DVB.

B.2.4.1.3 Životnosť toku udalosti

Množina udalostí v StreamEventMessage opísaných v správe BioP::StreamEvent je možnou podmnožinou udalostí, ktoré môže používať aplikácia v priebehu programu. Z tohto dôvodu sa musia aplikácie prispôbiť dynamickým zmenám týchto správ. Transparentnosť pamäte keď (pozri článok B.5.2.1 "Transparentné kešovanie") a verzia mechanizmu poslucháča poskytujú aplikácie s prostriedkami, ako to dosiahnuť.

Podobne množina deskriptorov toku udalostí prenášaných kedykoľvek nemusí zodpovedať množine udalostí opísaných v BioP::StreamEventMessage.

Identifikátor udalosti – názov udalosti sa nesmie zmeniť, pokiaľ existuje názov. Ak sa názov vymaže, tak sa nesmie znovu zaviesť počas 60 sekúnd.

B.2.4.2 Deskriptory toku

B.2.4.2.1 Deskriptor odkazu NPT

POZNÁMKA. – Použitie NPT je zastarané a nie je definované v tomto dokumente.

B.2.4.2.2 Deskriptor toku udalosti

B.2.4.2.2.1 Priradenie identifikátorov id udalosti k času udalosti

Keď NPT poskytne časovú základňu pri plánovanom toku udalosti, pole eventNPT oznámi hodnotu NPT, pri ktorej udalosť nastane (alebo už nastala). Keď Timeline DVB poskytne časovú základňu pri udalosti, pole eventNPT poskytne taktováciu (tick) hodnotu, pri ktorej udalosť nastane (alebo už nastala) a musí sa vyjadriť v jednotkách tick_rate (taktovacia rýchlosť aktualizácie), používaných na definovanie Timeline DVB.

Každý StreamEventDescriptor poskytuje jedinečné priradenie medzi eventID a eventNPT. Keď koncové zariadenie zistí zmenu v eventNPT súvisiacu s hodnotou eventID, tak predefinuje čas, pri ktorom sa má udalosť spustiť.

Koncové zariadenia musia ignorovať plánované udalosti, keď uplynul eventNPT (pozri tiež články B.2.4.2.2.3 "Signalizácia udalostí urob to teraz").

POZNÁMKA. – Použitie NPT je zastarané a nie je definované v tomto dokumente.

B.2.4.2.2.2 Opakované používanie identifikátora id udalosti

Hodnoty ID udalosti sa môžu znovu použiť v ľubovoľnom počte. Napríklad po spustení udalosti sa môžu vysielat' deskriptory toku udalosti s rovnakým eventID, ale s odlišným eventNPT.

B.2.4.2.2.3 Signalizácia udalostí "urob to teraz"

ISO/IEC 13818-6 [4] sa nezmieňuje o vysielaní signalizácie "udalostí "urob to teraz".

Tieto udalosti sa identifikujú hodnotou eventID a tiež rozšírením id tabuľky (pozri článok B.2.4.3.5 "Kódovanie rozšírenia id tabuľky").

Keď hodnota eventID identifikuje udalosť "urob to teraz", potom koncové zariadenie musí ignorovať hodnotu eventNPT.

B.2.4.2.2.4 Neverejnú dáta

Obsahy poľa privateDataByte sa nemusia interpretovať koncovým zariadením. Aplikácia môže pristupovať k poľu privateDataByte pomocou metódy org.dvb.dsmcc: StreamEvent.getEventData.

B.2.4.2.3 Nepoužité deskriptory

Koncové zariadenia môžu ignorovať nasledujúce deskriptory, ak sú prítomné:

- deskriptor NPT Endpoint;
- deskriptor módu toku.

B.2.4.3 Sekcie DSM-CC nesúce deskriptory toku

B.2.4.3.1 Číslo verzie sekcie

Pole čísla verzie sekcie sa zväčšuje odrážajúc tým zmeny deskriptora(ov) toku prenášaného sekciami s rovnakou hodnotou table_id (0 x 3D) a table_id_extension.

Číslo verzie sa zvyšuje z dôvodu uvažovania zmeny hodnoty eventNPT v danom eventId.

B.2.4.3.2 Jednoduché spustenie udalostí "urob to teraz"

Koncové zariadenia reagujú na prvý prípad udalosti "urob to teraz" zistený na základe určitej kombinácie id tabuľky, rozšírenia identifikátora tabuľky a čísla verzie. Príjem ďalších kópií určitej udalosti sa musí ignorovať dovtedy, pokiaľ nebude detegované iné číslo verzie.

B.2.4.3.3 Číslo sekcie

V dokumente sa predpokladá, že koncové zariadenia berú do úvahy len číslo sekcie s hodnotou nula.

B.2.4.3.4 Sekcie DSM-CC s DSMCC_descriptor_list()

Ak pole table_id má hodnotu 0 x 3D, potom bit current_next_indicator sa musí nastaviť na "1".

B.2.4.3.5 Kódovanie rozšírenia tabuľky identifikátora

Pole sekcie rozšírenia tabuľky identifikátora obsahuje informácie o toku deskriptora(ov) prenášaného sekciou.

Tabuľka B.32 – Kódovanie rozšírenia tabuľky identifikátora DSMCC_descriptor_lists

Bity table_id_extension			Užitočné dáta sekcie DSM-CC s tabuľkou ID 0x3D
[15]	[14]	[13 až 0]	
0	0	eventID[13 až 0]	Sekcia prenáša jedinú udalosť "urob to teraz"
0	1	xx xxxx xxxx xxxx	Sekcia prenáša deskriptory odkazu NPT
1	0	xx xxxx xxxx xxxx	Sekcia prenáša jeden alebo viac deskriptorov toku, t. j.: - deskriptor(y) toku udalosti s budúcimi udalosťami eventNPT; - deskriptor módu toku (v tomto dokumente sa neuvažuje); - deskriptor konca NPT (v tomto dokumente sa neuvažuje).
1	1	rezervované na budúce používanie	

POZNÁMKA. – Použitie NPT je zastarané a nie je definované v tomto dokumente.

Hodnota eventID pri udalosti "urob to teraz" musí byť v rozsahu od 0 x 0001 do 0 x 3FFF. Hodnota eventID pri plánovanej udalosti musí byť v rozmedzí od 0 x 8000 do 0 x BFFF. Hodnota 0 nie je povolená (pozri článok 5.5.2.2.1 v ISO/IEC 13818-6 [4]).

B.2.4.4 Vysielanie časových základní

V jednom programe MPEG sa môžu definovať viaceré súbežné časové základne, ale v danom okamihu môže byť k dispozícii iba jedna časová základňa, ostatné časové základne sa musia pozastaviť. Časové základne sa môžu poskytovať dvomi spôsobmi: DSM-CC NPT a DVB Timeline.

POZNÁMKA. – Použitie DSM-CC NPT je zastarané a nie je definované v tomto dokumente.

B.2.4.4.1 Časový prehľad DVB (voliteľný)

Vzťah medzi každým časovým prehľadom DVB a časovou základňou MPEG (STC) je definovaný pomocou deskriptora `broadcast_timeline_descriptor` prenášaného v synchronizovaných pomocných dátach DVB TS 102 823 [13]. Pole `broadcast_timeline_id` deskriptora `broadcast_timeline_descriptor` (osembitové celé číslo bez znamienka) označuje časovú základňu.

Hodnota identifikátora poľa `TAP_STR_DVBTIMEL_USE` (šestnásťbitové celé číslo bez znamienka) správy `StreamMessage` alebo `StreamEventMessage` označuje časovú základňu spojenú s objektom `Stream/StreamEvent`. Súčasne sa môže používať viac správ `StreamMessage` alebo `StreamEventMessage` na poskytnutie povolenia viac časových základní tej istej služby.

V tomto profile môže `broadcast_timeline_descriptors` indikovať dva stavy:

- nepozastavené: pole `running_status` sa nastaví na "beží";
- pozastavené: pole `running_status` sa nastaví na "pozastavené".

Odkazovaný časový prehľad DVB sa môže definovať pomocou metódy priameho kódovania, alebo kódovania s posunom, uvedenej v TS 102 823 [13].

V toku synchronizovaných pomocných dát môžu byť okrem `broadcast_timeline_descriptor` prítomné aj iné deskripty. Koncové zariadenia musia preskočiť deskripty, ktoré nepodporujú a pokračovať v spracovaní ďalšieho deskriptora. Nie sú podporované deskripty obsahujúce tieto definované, ale nepodporované, napríklad deskriptor `TVA_id` v zariadeniach nepožadujúcich podporu tohto deskriptora inou špecifikáciou, a deskripty, ktorých hodnota príznaku je rezervovaná, alebo je to hodnota neverejného používateľa (pozri tiež článok B.2.4.5.3).

Synchronizované pomocné dáta špecifikácie TS 102 823 [13] opisujú nespojitosti riadenia časového prehľadu DVB.

Koncové zariadenia musia byť v súlade s odporúčaniami na prijímače v článkoch D.5 TS 102 823 [13] vrátane tých, ktoré sa majú vymedziť v tomto článku. Koncové zariadenia nesmú spúšťať udalosti synchronizované časovým prehľadom DVB v prípadoch, keď stav časového prehľadu nie je známy, alebo ak sa časový prehľad vybral zo služby PMT, v ktorej sa prenáša.

B.2.4.5 Vysielanie udalostí

B.2.4.5.1 Tok udalostí DSM-CC "urob to teraz"

Koncové zariadenia musia podporovať tok udalostí DSM-CC "urobiť to teraz" definovaných v článku B.2.4.2.2. Ten poskytuje prostriedky na doručovanie samostatných udalostí bez nutnosti vysielania časovej základne.

Keď sú tieto udalosti doručované v užitočných dátach sekcie MPEG, nemôžu sa presne synchronizovať so spojitými mediálnymi tokmi.

B.2.4.5.2 Tok plánovaných udalostí DSM-CC

Koncové zariadenia musia podporovať tok plánovaných udalostí DSM-CC, ako sú definované v bode B.2.4.2.2 odkazom na vysielanie časovej základne definovanej pomocou niektorého mechanizmu časového prehľadu DVB.

Pri toku plánovaných udalostí DSM-CC sa predpokladajú nasledujúce scenáre použitia:

- môže sa použiť jedna spojitá časová základňa, t. j. jediná postupujúca hodnota času. V tomto prípade je vysielanie jeden logicky nepretržite pokračujúci interaktívny proces a vysielateľ je zodpovedný za priebežné spracovanie aplikácií, a to ešte pred vysielaním, aby zaručil ich vhodnosť;
- signál prijatý koncovým zariadením obsahuje jedinečnú časovú základňu v každom programe zvlášť; táto časová základňa sa môže prerušiť počas každého vkladania do programu a ukončiť na konci programu.

B.2.4.5.3 Udalosti synchronizované DVB

Koncové zariadenia musia podporovať udalosti synchronizované DVB, definované v TS 102 823 [13]. Tento mechanizmus umožňuje presné generovanie udalostí bez nutnosti časovej základne.

Udalosti synchronizované DVB sú definované pomocou `synchronised_event_descriptor`.

Pole `synchronised_event_context` tohto deskriptora identifikuje kontext konkrétnej aplikácie pri skupine udalostí a odkazuje sa pomocou `id` poľa `STR_DVBEVENT_USE_TAP_DSM-CC_StreamEventMessage`.

Obsah poľa `synchronised_event_data_byte` sa nemusí interpretovať koncovým zariadením. Aplikácia môže pristupovať k poľu `synchronised_event_data_byte` pomocou metódy `org.dvb.dsmcc.StreamEvent.getEventData`.

Koncové zariadenia musia podporovať zrušenie udalostí synchronizovaných DVB, ktoré neboli doteraz spustené pomocou `synchronised_event_cancel_descriptor`.

Podporovať sa musí zrušenie jednotlivých udalostí a skupín udalostí so spoločným `synchronised_event_context`.

V toku synchronizovaných pomocných dát môžu byť okrem `synchronised_event_descriptor` a `synchronised_event_cancel_descriptor` prítomné aj iné deskripty. Koncové zariadenia musia preskočiť tie deskripty, ktoré nepodporujú a pokračovať v spracovaní ďalšieho deskriptora. Nie sú podporované deskripty obsahujúce tieto definované, ale nepodporované, napríklad deskriptor `TVA_id` v zariadeniach nepožadujúcich podporu tohto deskriptora inou špecifikáciou, a deskripty, ktorých hodnota príznaku je rezervovaná, alebo je pre neverejného používateľa (pozri tiež článok B.2.4.4.2).

Udalosti synchronizované DVB majú zadaný čas prezentácie, určený prezentačnými časovými značkami zo štruktúr pomocných dát nesúcich deskripty `synchronised_event_descriptor` a pole `reference_offset_tick` deskriptora `synchronised_event_descriptor`. Udalosti sú synchronizované k časovému okamihu so spojitými mediálnymi tokmi, ako je video alebo audio. Prijímače musia doručiť udalosti aplikáciám načasované tak, aby nastala synchronizácia so spojitými médiami.

B.2.4.6 Monitorovanie vysielania časových základní a udalostí

B.2.4.6.1 Monitorovanie referenčnej časovej základne

Keď sa aplikácie zaregistrujú pri udalosti vynútené časovou základňou, koncové zariadenie musí prideliť dostatočné prostriedky na to, aby sa aktualizácia skupiny časových základní detegovala. Čas, za ktorý možno zistiť aktualizácie, závisí od typu aplikácie.

B.2.4.6.2 Monitorovanie udalosti vynútenej časovou základňou

Keď aplikácie boli zaregistrované pri udalosti vynútené časovou základňou, koncové zariadenie musí prideliť dostatočné prostriedky na to, aby sa aktualizácia skupiny udalostí vynútených časovou základňou detegovala. Čas za ktorý možno zistiť aktualizácie, závisí od typu aplikácie. Takisto, ak je udalosť prijatá alebo čas, ktorý je určený na jej spustenie, sa zmení, potom koncové zariadenie rešpektuje túto zmenu v stanovenom čase. Ak sa spustenie udalosti neuskutoční v pôvodne naplánovanom čase, ale spustí sa skôr, potom nie je žiadna záruka, že všetky prijímače detegujú zmenu včas.

Prijímač musí deaktivovať všetky udalosti poslucháčov závislé od časovej základne a môže uvoľniť prostriedky priradené týmto poslucháčom, ak:

- časová základňa je časová základňa NPT a je odstránená (odkaz na ňu je odstránený zo skupiny deskriptorov NPTReferenceDescriptor);
- časová základňa je časová základňa NPT a diskontinuita je detegovaná v tejto časovej základni;
- operácia výberu služby zmení aktuálnu službu.

POZNÁMKA. – Použitie NPT je zastarané a nie je definované v tomto dokumente.

B.2.4.6.3 Tok udalostí DSM-CC "urob to teraz"

Udalosti "urob to teraz" sú jednorazové udalosti, preto koncové zariadenia potrebujú vykonať osobitné činnosti s cieľom zabezpečiť vysokú pravdepodobnosť ich spoľahlivého prijatia.

Koncové zariadenie nepožaduje sledovať viac ako jednu zložku doručujúcu tok udalostí "urob to teraz" na každú aplikáciu. Preto, ak sú udalosti požadované viac ako z jedného DSM-CC StreamEventMessage a nie viac ako z jednej zložky toku, musia sa ako prostriedky špecifikovať deskriptory StreamEventDescriptor, ktoré prenášajú udalosti "urob to teraz" (t. j. TAP obsahujúce STR_EVENTUSE alebo STR_STATUS_AND_EVENT_USE musia mať rovnakú hodnotu v prípade odkazovania sa na udalosti "urob to teraz").

Koncové zariadenia musia vyhradiť časť filtra na sledovanie možného prenosu udalostí "urob to teraz", keď sú tam akékoľvek aplikácie pridružené k týmto udalostiam.

B.2.4.6.4 Tok plánovaných udalostí DSM-CC

Deskriptory toku pri plánovaných udalostiach sú prenášané niekoľkokrát v časovom intervale predtým, ako sa majú spúšťať. Tým sa vytvára vysoká pravdepodobnosť ich použitia, aj keď nie sú nepretržite monitorované koncovým zariadením.

Všetky naplánované deskriptory toku udalosti sa vysielajú aspoň raz za sekundu.

Koncové zariadenia vyvolajú udalosť ako odozvu na plánovaný tok udalosti za predpokladu, že deskriptory toku udalosti sú vysielané počas aspoň piatich sekúnd pred plánovaným časom.

Koncové zariadenie nepožaduje sledovať viac ako jednu zložku doručujúcu tok naplánovanej udalosti na každú aplikáciu. Preto, ak sú udalosti požadované viac ako z jedného DSM-CC StreamEventMessage a nie viac ako z jednej zložky toku, musia sa ako prostriedky špecifikovať deskriptory StreamEventDescriptor, ktoré prenášajú naplánované udalosti (t. j. TAP obsahujúce STR_EVENTUSE alebo STR_STATUS_AND_EVENT_USE musia mať rovnakú hodnotu v prípade odkazovania sa na naplánované udalosti).

POZNÁMKA. – Toky plánovaných udalostí a udalostí "urob to teraz" sa môžu prenášať rôznymi zložkami toku. Od koncového zariadenia sa požaduje, aby bolo schopné zobraziť z nich jeden tok.

B.2.4.6.5 Počet zložiek časovej základne

Od koncového zariadenia sa požaduje sledovať len jednu zložku časovej základne. Preto, ak sú udalosti požadované z viacerých tokov DSM-CC StreamEventMessage, každý tok StreamEventMessage, ktorý odkazuje na časovú základňu, musí odkazovať na rovnaký typ (NPT alebo časový prehľad DVB) a musí obsahovať TAP STR_NPT_USE alebo STR_DVBTIMEL_USE, ktorý špecifikuje rovnaký príznak priradenia.

POZNÁMKA. – Použitie NPT je zastarané a nie je definované v tomto dokumente.

B.2.4.6.6 Udalosti synchronizované DVB

Synchronizované udalosti DVB sú prechodné udalosti, t. j. vysielané len počas krátko časového intervalu, preto koncové zariadenia potrebujú vykonať osobitné činnosti s cieľom zabezpečiť vysokú pravdepodobnosť ich spoľahlivého prijatia.

Od koncového zariadenia sa nepožaduje sledovať viac ako jednu zložku doručujúcu synchronizované udalosti DVB na každú aplikáciu, preto ak sú udalosti požadované z viacerých DSM-CC StreamEventMessage a nie viac ako z jednej zložky toku, musia sa ako prostriedky špecifikovať synchronizované udalosti DVB, t. j. TAP obsahujúce STR_DVBEVENT_USE musia mať rovnakú hodnotu ako príznak priradenia.

Koncové zariadenia musia vyhradiť filter na sledovanie možného prenosu takýchto udalostí, ktoré obsahujú akékoľvek s nimi súvisiace aplikácie.

Od koncového zariadenia sa nepožaduje sledovať tok udalostí DSM-CC "urob to teraz" v rovnakom čase ako sledovanie synchronizovaných udalostí DVB.

B.2.5 Priradovanie a používanie hodnôt transactionId

B.2.5.1 Informatívny základ

Používanie TransactionId v objektovom karuseli sa prenáša z jeho používania definovaného v špecifikácii DSM-CC a samo osebe sa javí trochu zložitejšie. TransactionId má dvojakú úlohu, poskytuje identifikáciu a súčasne mechanizmy vytvárania verzií na riadenie správ, t. j. správy DownloadInfoIndication a DownloadServerInitiate. TransactionId má jednoznačne identifikovať správu riadenia sťahovania v rámci dátového karuselu a má sa "zvýšiť" vždy pri akejkoľvek zmene poľa správy.

POZNÁMKA. – Pojem "zvýšené" sa používa v špecifikácii DSM-CC. V rámci tohto dokumentu sa to má interpretovať ako "zmenené".

Objektový karusel sa prenáša na najvyššej úrovni jedného alebo viacerých dátových karuselov. Prostredníctvom dátového karuselu používaného pod objektovým karuselom sa v tomto dokumente myslí skupina správ, ktorú predstavuje správa DownloadInfoIndication prenášaná jedným PID, a správ DownloadDataBlock, ktoré prenášajú moduly opísané v správach DownloadInfoIndication. Správy DownloadDataBlock sa môžu rozdeliť na ďalšie elementárne toky ako správy DownloadInfoIndication. Správa DownloadServerInitiate v rámci objektového karuselu je považovaná za súčasť najvyššej úrovne objektového karuselu a nesúvisí so žiadnymi dátami karuselu.

Zmenou modulu sa musí zmeniť číslo verzie modulu. To znamená, že správa DownloadInfoIndication odkazujúca na modul, sa musí tiež aktualizovať. Ak je aktualizovaná DownloadInfoIndication, potom sa musí zmeniť aj TransactionId. Identifikátor transactionId správy

DownloadInfoIndication sa používa aj v iných správach, ale zmenám v nich sa má špecificky zabrániť a dôsledky aktualizácie modulu sa majú týkať len samotného modulu a DownloadInfoIndication, ktorý odkazuje na modul. Ďalšie pravidlá používania transactionId boli špecifikované týmto spôsobom.

B.2.5.2 Sémantika poľa transactionId DVB

Identifikátor transactionId sa rozdeľuje do niekoľkých subpolí, ktoré sú definované v tabuľke B.33. Rozdelenie odráža dvojité úlohu identifikátora transactionId (uvedenú skôr) a obmedzenia zavádzané na zníženie účinkov aktualizácie modulu. Zvyšovanie interoperability priradenia transactionId bolo navrhnuté tak, aby bolo nezávislé od predpokladanej filtrácie v cieľových koncových zariadeniach.

Tabuľka B.33 – Subpole transactionID

Bity	Hodnota	Subpole	Opis
0	Definované používateľom	Aktualizovaný príznak	Tento bit sa musí prepnúť vždy pri aktualizácii riadiacej správy
1 až 15	Definované používateľom	Identifikácia	Tieto bity musia a môžu obsahovať len samé nuly na správu DownloadServerInitiate. Všetky ostatné riadiace správy musia mať jeden bit alebo viac nenulových bitov.
16 až 29	Definované používateľom	Verzia	Číslo verzie sa zvýši vždy pri aktualizácii riadiacej správy. Táto hodnota sa zvýši o jeden.
30 až 31	Bit 30 – nulový Bit 31 – nenulový	Tvorca	Toto je definované v špecifikácii DSM-CC ISO/IEC 13818-6 [4] ako 0x02, ak transactionId bol priradený sieťou – v prípade vysielania je implicitný.

Vzhľadom na úlohu transactionId ako mechanizmu vytvárania verzií každá zmena riadiacej správy spôsobí, že transactionId tejto riadiacej správy sa zvýši. Akákoľvek zmena modulu si vyžiada zvýšenie moduleVersion jeho poľa. Táto zmena sa musí premietnuť do príslušného poľa v opise modulu v správe(vach) DownloadInfoIndication, ktorá to opisuje. Keďže pole v správe DownloadInfoIndication sa zmení, tak zvýšenie jeho identifikátora transactionId sa musí indikovať novou verziou správy. Rovnako, akákoľvek zmena v správe DownloadServerInitiate spôsobí, že jej identifikátor transactionId sa musí zvýšiť.

Ak je identifikátor transactionId rozdelený do subpolí, ako je uvedené skôr, aktualizácia správy zmení verziu len časti transactionId, zatiaľ čo časť identifikácie zostáva rovnaká.

Keďže transactionId sa používa aj na identifikáciu správ pri odkazovaní na správy v iných štruktúrach, je veľmi žiaduce, aby tieto odkazované správy nebolo potrebné aktualizovať zakaždým, keď je aktualizovaná riadiaca správa. Z tohto dôvodu sa toto pravidlo uplatňuje pri vyhľadávaní správ na základe odkazov.

Pri umiestnení správy na základe hodnoty identifikátora transactionId, použitej pri odkazovaní na správu, sa porovnáva len identifikačná časť identifikátora (bity 1 – 15).

Dôsledky aktualizácie modulu použitím tohto pravidla môžu obmedziť samotný modul a správu DownloadInfoIndication, ktorá opisuje modul. To tiež znamená, že ak koncové zariadenie chce zistiť, či konkrétny modul, ktorý bol vybraný, sa predtým zmenil, je potrebné filtrovať správu DownloadInfoIndication, ktorá opisuje daný modul a kontroluje, či došlo k zmene.

B.2.6 Mapovanie objektov do modulov dátového karuselu

Objektové karusely DSM-CC umožňujú prenos jedného alebo viacerých objektov v jednom module dátového karuselu. S cieľom optimalizovať výkon a požiadavky na pamäť sú definované tri ďalšie požiadavky:

- pri mapovaní objektov do modulov dátového karuselu sa majú umiestniť do jedného modulu len úzko súvisiace objekty; objekty, ktoré úzko nesúvisia, nemajú byť v tom istom module; keď sa počas vyberania objektu z karuselu koncového zariadenia získa modul, ktorý obsahuje viac objektov, majú sa tieto objekty uložiť do pamäte keď z toho dôvodu, že ďalšie objekty sa môžu spojiť s požadovaným objektom a pravdepodobne sa môžu čoskoro použiť;
- veľkosť modulu s viacerými objektmi nemá presiahnuť 65 536 bajtov po dekomprimovaní, t. j. keď bol súbor dekomprimovaný z prenášaného súboru skôr, ako sa začalo dekódovanie obsahu; koncové zariadenia vyhovujúce tomuto dokumentu sú povinné spracovať moduly obsahujúce niekoľko objektov, keď veľkosť modulu pri dekomprimovaní je 65 536 bajtov alebo menej; moduly, ktoré obsahujú jeden súbor správy, môžu presiahnuť 65 536 bajtov a hornú hranicu obmedzujú len pamäťové prostriedky v koncovom zariadení;
- okrem obmedzení stanovených limitom 65 536 bajtov sú správy adresára a služby sieťového priechodu obmedzené na 512 objektov viazaných na jednu správu.

B.2.7 Kompresia modulov

Moduly sa môžu prenášať v nekomprimovanej alebo komprimovanej podobe. V prípade, že sa modul prenáša v komprimovanej podobe, je to signalizované, vrátane deskriptora `compressed_module_descriptor` v poli `userInfo` informačného modulu `ModuleInfo` v správe `DownloadInfoIndication`.

Tabuľka B.34 ukazuje syntax deskriptora kompresie modulov `compressed_module_descriptor`.

Tabuľka B.34 – Deskriptor `compressed_module_descriptor`

	Počet bitov	Identifikátor	Hodnota
<code>compressed_module_descriptor(){</code>			
<code>descriptor_tag</code>	8	<code>uimsbf</code>	<code>0x09</code>
<code>descriptor_length</code>	8	<code>uimsbf</code>	
<code>compression_method</code>	8	<code>uimsbf</code>	
<code>original_size</code>	32	<code>uimsbf</code>	
<code>}</code>			

Prítomnosť deskriptora `compressed_module_descriptor` indikuje, že dáta v module majú štruktúru "zlib", ktorá je definovaná v RFC 1950 [11].

Koncové zariadenie musí podporovať kompresný algoritmus Deflate, ako je uvedené v RFC 1951 [12]. Toto je signalizované nastavením najmenej významných 4 bitov (nibble) `compression_method` na `0 x 8` (t. j. `compression_method` je `xxxx1000`). Koncové zariadenie nie je povinné podporovať iné kompresné algoritmy.

B.2.8 Zostavenie objektového karuselu

Objekt `ServiceGateway` je koreňový adresár systému súborov dodávaných objektovým karuselom a musí sa získať stiahnutím pred akýmkoľvek iným objektom. To sa môže dosiahnuť dvoma kompatibilnými spôsobmi. Signalizáciu, ktorej spôsoby sú podporované vysielaním, poskytuje `carousel_identifier_descriptor`.

V tomto dokumente je použitie `carousel_identifier_descriptor` na signalizáciu povinné v druhej slučke deskriptora PMT (zodpovedajúci PID vysielajú správu DSI objektovému karuselu, t. j. boot-PID). Dôsledkom je, že ak PMT druhej slučky deskriptora obsahuje `data_broadcast_id_descriptor`, ktorý poskytuje signalizáciu podľa tohto dokumentu, musí tiež obsahovať `carousel_identifier_descriptor`.

POZNÁMKA. – Jeden PID obsahuje správy len z jedného objektového karuselu, a preto je prítomný len jeden carousel_identifier_descriptor v každom druhom deskriptore slučky. Jedna služba môže obsahovať aj viac objektových karuselov, preto sa môže carousel_identifier_descriptor objaviť viackrát v rámci jednej PMT.

Objekt ServiceGateway sa môže získať prostredníctvom štandardného mechanizmu DSI-DII. Toto sa musí podporovať všetkými vysielaniami bez ohľadu na signalizáciu v carousel_identifier_descriptor a musia byť dostatočné do všetkých koncových zariadení (pozri tiež článok 5.3.2 "Špecifické informácie o programe").

Vysielanie môže tiež obsahovať ďalšie informácie v carousel_identifier_descriptor na podporu "rozšíreného" zavádzacieho mechanizmu. Toto je signalizované nastavením poľa formatId s týmto deskriptorom na 0 x 01. Táto dodatočná informácia je spojením všetkých polí potrebných na vyhľadanie ServiceGateway a hľadanie v správach DSI a DII; v tomto prípade modul obsahujúci objekt ServiceGateway sa musí vysielat' v PID identifikovaným data_broadcast_id_descriptor, čo je voliteľné pri oboch vysielaniach a koncové zariadenia podporujú tento mechanizmus.

B.2.8.1 Deskriptor identifikátora karuselu

Tento deskriptor je definovaný v MPEG a v tomto dokumente sa môže nachádzať v druhom deskriptore slučky PMT.

Tabuľka B.35 – Syntax deskriptora identifikátora karuselu

Syntax	Počet bitov	Identifikátor	Hodnota
carousel_identifier_descriptor {			
descriptor_tag	8	uimsbf	0x13
descriptor_length	8	uimsbf	N1
carousel_id	32	uimsbf	
FormatID	8	uimsbf	
if(FormatID == 0x00) {			
for(i=0; i<N1 - 5; i++){			
private_data_byte	8		
}			
}			
if(FormatID == 0x01) {			
ModuleVersion	8	uimsbf	
ModuleId	16	uimsbf	
BlockSize	16	uimsbf	
ModuleSize	32	uimsbf	
CompressionMethod	8	uimsbf	
OriginalSize	32	uimsbf	
TimeOut	8	uimsbf	
ObjectKeyLength	8	uimsbf	N2 ≤ 4
for(i=0; i<N2; i++){			
ObjectKeyData	8	bslbf	
}			
for(i=0; i<N1 - N2 - 21; i++){			
private_data_byte	8		
}			
}			
}			

identifikátor karuselu (angl. **carousel_id**): tridsaťdvabitové pole identifikuje objektový karusel so zodpovedajúcim carousel_id; ak je karusel určený na podieľanie viacerými prenosovými tokmi, potom 24 najvýznamnejších bitov carousel_id nesie 24 najmenej významných bitov organisation_id prevádzkovateľa, ak sa karusel nemá podieľať, potom týchto 24 bitov musí byť nulových; zvyšných 8 najmenej významných bitov môže mať ľubovoľnú hodnotu

V rámci programu musí byť karusel carousel_id jedinečný.

formát identifikátora (angl. **FormatID**): osembitové celé číslo určuje, či karusel podporuje mechanizmus rozšíreného zavádzania alebo nie; hodnota 0 x 00 indikuje "štandardné zavádzanie", 0 x 01 znamená "rozšírené zavádzanie"

verzia modulu (angl. **ModuleVersion**): osembitové celé číslo je číslo verzie modulu, ktoré obsahuje sieťový priechod služby, čo je ekvivalentom moduleVersion v DII

identifikátor modulu (angl. **ModuleId**): šesťnásťbitové celé číslo je identifikátor modulu v karuseli, čo je ekvivalentom moduleId v DII

veľkosť bloku (angl. **BlockSize**): šesťnásťbitové celé číslo predstavuje veľkosť každého bloku v module v bajtoch, s výnimkou posledného bloku, ktorý môže byť rovnaký alebo je menší, čo je ekvivalentnom BlockSize v DII

veľkosť modulu (angl. **ModuleSize**): tridsaťdvabitové celé číslo predstavuje veľkosť modulu v bajtoch, čo je ekvivalentom moduleSize v DII

metóda kompresie (angl. **CompressionMethod**): osembitové pole predstavuje kompresný algoritmus definovaný v RFC 1950 [11], používaný na kompresiu modulu, čo je ekvivalentom compression_method prenášanú v compressed_module_descriptor v DII

veľkosť originálu (angl. **OriginalSize**): tridsaťdvabitové celé číslo predstavuje veľkosť dát v bajtoch, prenášané modulom predtým, ako boli komprimované, čo je ekvivalentom original_size prenášanú v compressed_module_descriptor v DII

V prípade, že modul nebol komprimovaný, hodnoty OriginalSize a ModuleSize sa musia rovnať a hodnota CompressionMethod nie je definovaná.

časový limit (angl. **TimeOut**): osembitové celé číslo určuje časový limit v sekundách na získanie všetkých blokov modulu

dĺžka kľúčového objektu (angl. **ObjectKeyLength**): osembitové celé číslo určuje počet bajtov ObjectKeyData

dáta kľúčového objektu (angl. **ObjectKeyData**): osembitové hodnoty tvoria oktetový reťazec, ktorý identifikuje správu BIOP, ktorá je správou ServiceGateway

B.2.9 Nedostupnosť karuselu

Vysielacie karusely sa stanú trvale nedostupné v dôsledku zmien v signalizácii, vrátane nasledujúcich prípadov:

- zložka signalizovaná ako prenos DSI je odstránená z PMT;
- hodnota identifikátora karuselu súvisiaceho so zmenami karuselu;
- program nie je prístupný z PAT;
- po celkovom časovom výpadku signalizácie závislom od implementácie, napríklad keď sa neprenáša PMT.

Navyše sa môže stať, že karusely sa stanú trvale nedostupné pri strate spojenia dočasným odpojením karuselu.

B.2.10 Poskytovanie viacerých služieb karuselmi

Karusely sa považujú za zhodné, ak v tabuľkách PMT služieb platia všetky nasledujúce podmienky:

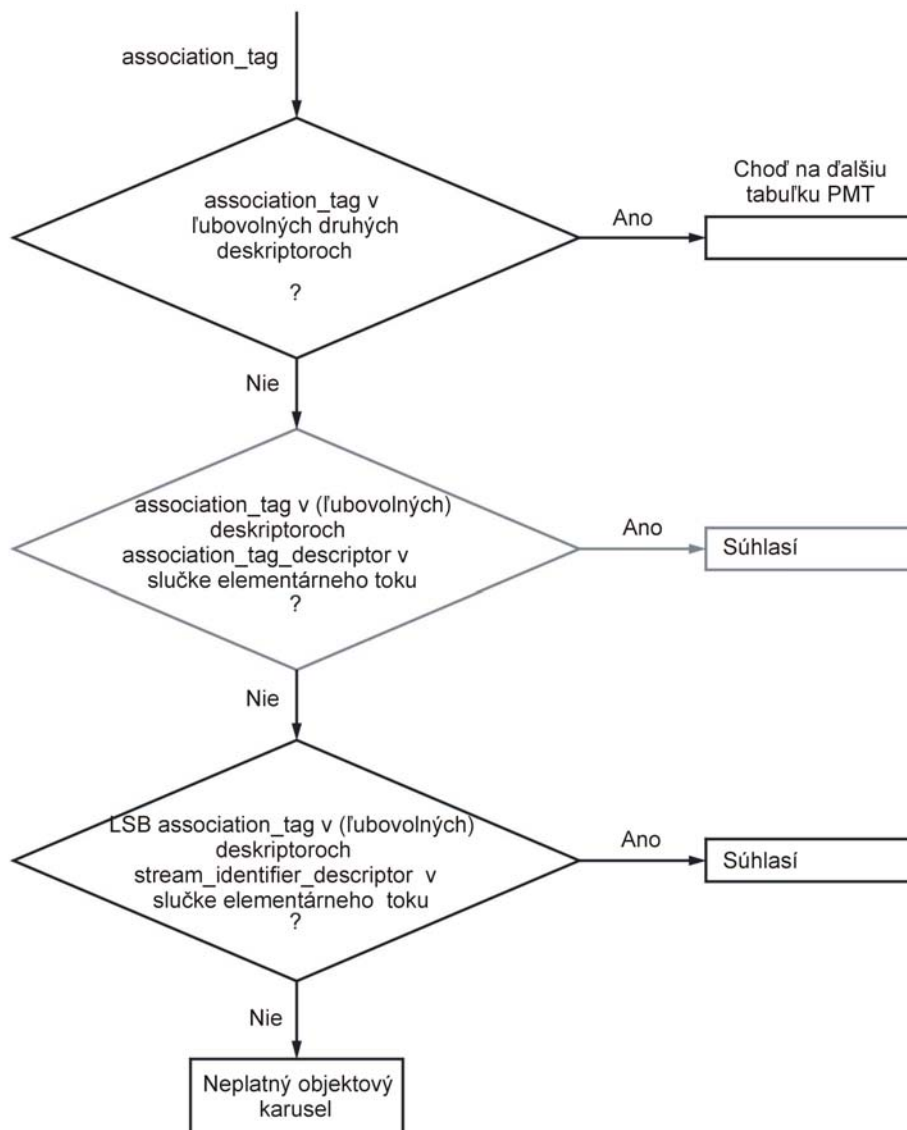
- a) obidve služby sa poskytujú v rámci rovnakého prenosového toku;
 - obidve služby prehľadávajú spúšťacie komponenty v karuseli s rovnakým PID;
 - deskriptor karuselu `carousel_identifier_descriptor` je zhodný v oboch službách (karusely majú rovnaký `carousel_id` a zavádzacie parametre);
 - všetky súvisiace príznaky používané v karuseli mapujú rovnaké PID v oboch službách;
 - v tomto prípade karusel vysiela jednou cestou, ale služby sa môžu odkazovať na karusel cez viaceré cesty, vrátane prerušenia druhej tabuľky PMT cez prerušené príznaky priradenia;
- b) obidve služby sa poskytujú v rámci viacerých prenosových tokov;
 - deskriptor `carousel_id` v `carousel_identifier_descriptor` je v rozsahu od 0 x 100 do 0 x ffffffff; obsahuje identifikátor vysielateľa `organisation_id` v najvýznamnejších 24 bitoch v `carousel_id`;
 - deskriptor `carousel_identifier_descriptor` do karusela je zhodný v oboch službách (karusely majú rovnaký karuselový `Id` a zavádzacie parametre).

B.3 Mapovanie príznaku priradenia

B.3.1 Algoritmus rozhodovania na mapovanie príznaku priradenia

B.3.1.1 TAPUse nie je BIOP_PROGRAM_USE

Obrázok B.1 znázorňuje vývojový diagram na identifikáciu elementárneho toku(ov), ktorý prenáša objektový karusel.



Obrázok B.1 – Vývojový diagram identifikácie ES objektovým karuselom

V tomto dokumente sa používa deskriptor `stream_identifier_descriptor` vždy na priradenie `component_tag` elementárnych tokov. Použitie deskriptorov `association_tag_descriptor` podľa DSM-CC [4] sa nevyžaduje. Ak sa používa `association_tag_descriptor` voľiteľne, musí byť prítomný aj `stream_identifier_descriptor`, ako je definované v EN 300 468 [1] a hodnoty príznaku sa stanovujú konzistentne v každom deskriptore. Toto obmedzenie zjednodušuje vývojový diagram uvedený na obrázku B.1 tak, že druhé rozhodnutie sa môže vynechať.

B.3.1.2 TAPUse je BIOP_PROGRAM_USE

Vývojový diagram v článku B.1 nepokračuje, keď sa rieši TAP BIOP_PROGRAM_USE ako jediné platné kódovanie vysielania, potom v tomto TAP je použitý BIOP_PROGRAM_USE na vyriešenie deferred_association_tags_descriptor v PMT, aj keď deferred_association_tags_descriptor identifikuje aktuálnu službu (t. j. tok alebo odkaz na jeho vlastný streamEvent). Ak toto rozhodnutie zlyhá, potom služba zostavená objektovým karuselom sa musí vrátiť ako odkazovaná služba.

B.3.2 Príznyky priradenia

Deskriptor component_tag v stream_identifier_descriptor tabuľky PMT, ako je definované v EN 300 468 [1], sa používa na spojenie informácií o zložke služby SI s elementárnym tokom, bez priameho odkazovania na hodnotu PID. Podobne DSM-CC používa príznaky association_tag na účely odkazovania na elementárny tok bez priameho odkazovania na hodnotu PID. Hodnota association_tag je mapovaná elementárnym tokom porovnaním LSB association_tag s hodnotou component_tag. Deskriptor stream_identifier_descriptor je povinný do všetkých zložiek, ktoré sa odkazujú na aplikáciu alebo objektový karusel.

Vysielatelia sa môžu rozhodnúť o používaní deskriptorov association_tag_descriptor podľa normy ISO/IEC 13818-6 [4], ktoré sa majú (teoreticky) testovať na porovnanie pred vyskúšaním príznakov component_tag. Hodnoty LSB association_tag v association_tag_descriptor sa musia zhodovať s hodnotou príznaku component_tag s daným PID. Vzhľadom na to, že príznak component_tag je jedinečný v rámci PMT, tak nie je potrebné, aby sa zhodoval s deskriptorom association_tag_descriptor.

Deskriptor deferred_association_tags_descriptor, požadovaný týmto dokumentom, je úpravou deskriptora v ISO/IEC 13818-6 [4], ktorý je definovaný v TR 101 202 [i.2]. Táto druhá definícia normalizuje mechanizmus signalizácie pôvodného identifikátora siete.

Pri pokuse o mapovanie príznaku elementárneho toku association_tag musí association_tag najprv zistiť opäť všetky deskriptory deferred_association_tags_descriptor v aktuálnej tabuľke PMT. Aktuálne v tomto kontexte znamená tabuľku PMT služby, v rámci ktorej je príznak association_tag mapovaný. V prípade, že príznak association_tag zodpovedá niektorému z príznakov association_tag, ktoré sú prítomné v deferred_association_tags_descriptor, potom zodpovedajúci proces pokračuje indikáciou služby v uvedenom deskriptore. Od koncového zariadenia sa nepožaduje, aby pokračovalo vo svojom hľadaní v pozadí tejto druhej služby.

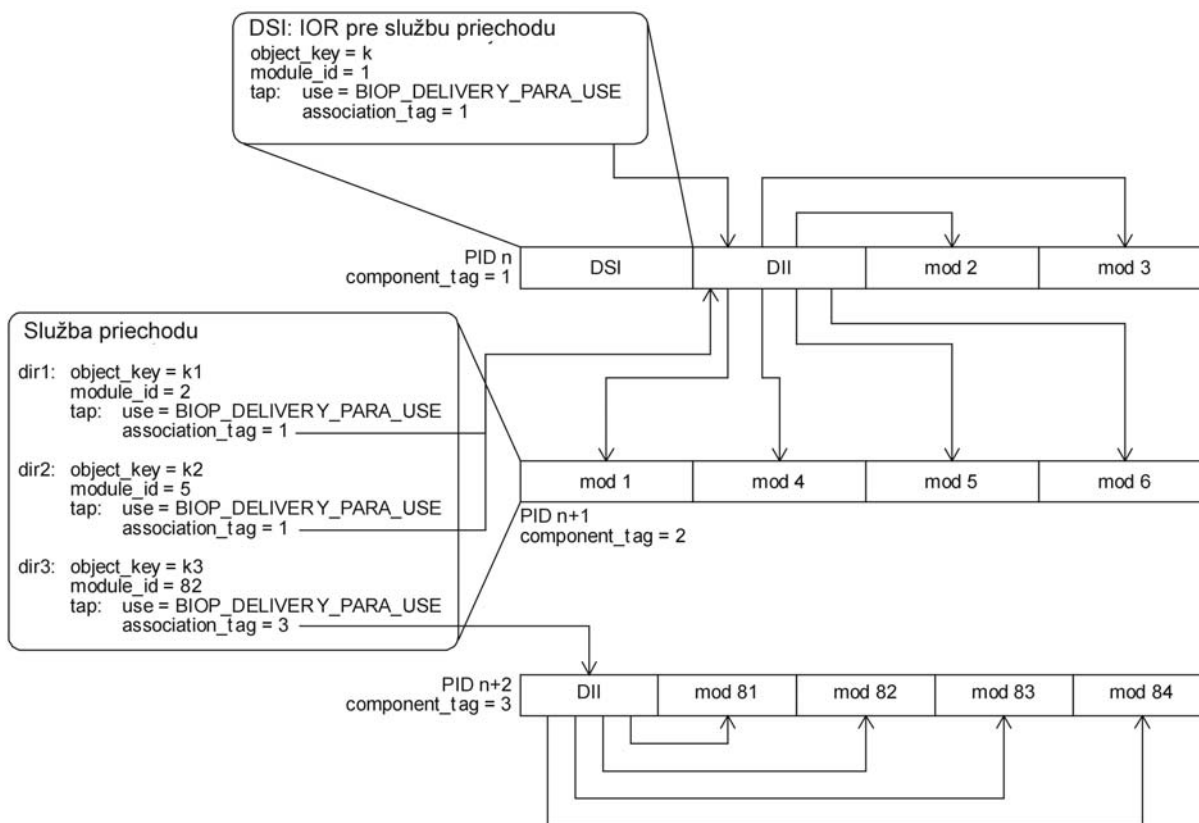
Keď je pole transport_stream_id v deferred_association_tags_descriptor nastavené na 0 x 0000, potom toto nastavenie treba ignorovať a koncové zariadenie si môže zvoliť, ktorý identifikátor prenosového toku vyberie pri získavaní služby.

B.3.3 Deskriptor prerušenia príznakov priradenia

Pole transport_stream_id deskriptora deferred_association_tags_descriptor, ako je definované v DSMCC [4], môže mať hodnotu 0 x 0000, ale v takom prípade sa musí ignorovať pri riešení odkazu.

B.4 Príklad objektového karuselu (informatívne)

Obrázok B.2 znázorňuje objektový karusel, ktorý je rozdelený do troch základných tokov, ktoré patria tej istej službe.



Obrázok B.2 – Príklad karuselu

Správa DownloadServerInitiate (DSI) sa prenáša prvým elementárnym tokom. Obsahuje odkaz na objekt, ktorý odkazuje na ServiceGateway. TAP s BIOP_DELIVERY_PARA_USE odkazuje na správu DownloadInfoIndication (DII), ktorá poskytuje informácie o module a mieste, kde je modul vysielaný. V tomto prípade je objekt ServiceGateway v module číslo 1, ktorý sa prenáša druhým elementárnym tokom (indikovaný štruktúrou TAP BIOP_OBJECT_USE v správe DII).

Objekt ServiceGateway je hlavný adresár, ktorý v tomto prípade odkazuje na tri podadresáre. TAP s BIOP_DELIVERY_PARA_USE sa používajú v objekte odkazov podadresárov na poskytovanie spojení na moduly cez správu DownloadInfoIndication (DII). Prvé dva podadresáre "dir1" a "dir2" sú uvedené v správe DII, ktorá sa prenáša v prvom elementárnom toku. Tretí podadresár sa odkazuje v správe DII prenášanej v treťom elementárnom toku.

V tomto prípade, prvé dva elementárne toky prenášajú správy jedného logického dátového karuselu, zatiaľ čo tretí elementárny tok prenáša správy ďalšieho logického dátového karuselu. Všetky tieto karusely patria tomu istému objektovému karuselu. Napríklad tretí elementárny tok obsahuje objekty v podadresári "dir3" a objekty v podadresároch "dir1" a "dir2" sú distribuované cez prvý a druhý elementárny tok.

POZNÁMKA. – Je potrebné si uvedomiť, že tretí elementárny tok môže pochádzať z úplne iného zdroja ako prvé dva elementárne toky. Adresárová štruktúra a objekty obsiahnuté v treťom elementárnom toku sú "zostavené" v hlavnom adresári za predpokladu, že položky adresára "dir3" obsahujú príslušnú informáciu o umiestnení.

Táto štruktúra sa môže použiť, napríklad v národnej informačnej službe, ktorá obsahuje niektoré regionálne časti spoločné národné časti sa môžu prenášať v tomto prípade v prvých dvoch elementárných tokoch, ktoré sú distribuované bez zmeny v rámci štátu. Regionálne časti sú prenášané v treťom elementárnom toku, ktorý je lokálne vložený v každej oblasti. Z pohľadu

aplikácie, spoločné národné časti sú v podadresároch "dir1" a "dir2", zatiaľ čo regionálne časti sú v podadresári "dir3".

Iný príklad, kde sa tento typ štruktúry môže použiť, je v prípade služby, ktorá obsahuje viac nezávislých aplikácií. V tomto prípade sa môže každá aplikácia umiestniť do vlastného podadresára a tieto podadresáre sa môžu prenášať ako samostatné dátové karusely v rôznych elementárnych tokoch.

B.5 Kešovanie

Tento článok opisuje obmedzenia na koncové zariadenia, ktoré musia implementovať, v súlade s týmto dokumentom, kešovanie akéhokoľvek obsahu z objektového karuselu do pamäte koncového zariadenia. Kešovanie je v koncových zariadeniach voliteľné, ale ak sa implementuje, tak sa musí vykonať v súlade s obmedzeniami uvedenými v tomto článku.

B.5.1 Určenie verzie súboru

Číslo verzie nie je v priamom vzťahu k súborom (alebo iným správam BIOP), najbližšia väzba je moduleVersion v DII, ktorý odkazuje na modul so správou BioP. Preto je potrebné zaistiť aktuálny súbor, aby koncové zariadenie rozhodlo, ktorá moduleVersion príslušného modulu je aktuálna a, ak je to nutné, aj o aktualizácii. Okolnosti, za ktorých je táto kontrola požadovaná, sú definované úrovňou transparentnosti, ako je uvedené v nasledujúcom článku.

B.5.2 Úroveň transparentnosti kešovania

Definícia úrovni transparentnosti opisuje činnosť, ktorú koncové zariadenie musí realizovať, keď sa mení obsah v objektovom karuseli. Úroveň transparentnosti určuje, ako sa od určitého koncového zariadenia požaduje platnosť obsahu pri vrátení obsahu aplikácii. Objektový karusel poskytuje mechanizmus na stanovenie verzie zmeny obsahu sledovaním správ DII.

Platnosť obsahu je tu uvedená v tvare čísla verzie modulu, ktorý je vysielaný v správe DII. Obsahy objektu kešované v pamäti koncového zariadenia sú definované ako platné v určitom časovom okamihu, keď číslo verzie modulu v pamäti keš zodpovedá číslu verzie modulu signalizovanom v správe DII, ktorá opisuje tento modul, ako tomu bolo v predchádzajúcom vysielaní.

POZNÁMKA. – Definícia vychádza zo správy DII, ktorá bola naposledy vysielaná, a môže sa stať, že koncové zariadenie v tom čase túto správu nefiltrovalo a neprijalo ju.

Z pohľadu koncového zariadenia úroveň transparentnosti indikuje obmedzenia, ktoré koncové zariadenie potrebuje aplikovať na sledovanie správy DII.

Vysielateľ môže indikovať primeranú úroveň transparentnosti, ktorá sa použije v danej časti obsahu pomocou deskriptora spojeného s modulom v správe DII (pozri článok B.2.2.4.2 "Deskriptor priority kešovania"). Ak modul neobsahuje tento deskriptor, transparentné kešovanie predstavuje východiskovú úroveň.

B.5.2.1 Transparentné kešovanie

Transparentné kešovanie je úroveň kešovania, ktorá zaisťuje, že aplikácia nemôže prakticky zaznamenať rozdiel v platnosti vráteného obsahu medzi implementáciou, ktorá kešuje obsah, a implementáciou, ktorá nemá žiadny kešovaný obsah. Prirodzene, implementácia kešujúca obsah vráti obsah aplikácii rýchlejšie.

Koncové zariadenie pri vrátení obsahu z pamäte keš aplikácii zaisťuje, že číslo verzie kešovaného obsahu zodpovedá číslu verzie uvedenému v aktuálnej správe DII, ktorá opisuje tento modul. Len čo bola prijatá správa DII, možno predpokladať, že táto indikácia je aktuálna najmenej 500 ms a po

tomto čase až do prijatia ďalšieho výskytu príslušnej správy DII. Ak filtrácia správy DII nie je obnovená do konca tohto intervalu, stav tejto správy DII je potrebné považovať za neznámy až do prijatia novej správy DII.

Z uvedeného dôvodu koncové zariadenia nesmú vrátiť transparentne kešované dáta, ak čakali dlhšie ako pol sekundy medzi prijatím príslušnej DII a opätovným začatím filtrácie správy DII. Ak koncové zariadenie nebude pokračovať vo filtrácii v intervale 500 ms, musí stiahnuť príslušnú DII znova, keď chce danú DII použiť na kontrolu platnosti pamäte keš.

Voľba 500 ms vychádza z bežnej neistoty časového riadenia poskytovania údajov prostredníctvom vysielacieho reťazca a je nezávislá od frekvencie opakovania správ DII.

B.5.2.1.1 Aktívne kešovanie

Existuje niekoľko spôsobov, ako môže koncové zariadenie organizovať svoju stratégiu kešovania. Jednou z možných stratégií je tzv. aktívne kešovanie. To znamená, že koncové zariadenie má vyhradenú sekciu filtra pri každej správe DII, ktorú potrebuje sledovať. Správa, ktorú filter neustále filtruje pri DII, zaručuje, že len čo koncové zariadenie zaznamená aktualizáciu modulu, celý kešovaný obsah je platný.

V niektorých prípadoch sa môžu správy DII odosielať s vysokou opakovacou frekvenciou, čo môže spôsobiť vysoké zaťaženie, pretože koncové zariadenie spracováva každú správu DII, ktorú prijme. Časový interval 500 ms je navrhnutý tak, aby umožnil koncovému zariadeniu zastaviť sekciu filtra na 500 ms po prijatí správy DII. Tým sa znižuje pracovné zaťaženie koncového zariadenia tak, že spracovanie každej správy DII sa uskutočňuje dvakrát za sekundu, a to aj v prípade častejšieho opakovania prenosu.

B.5.2.1.2 Pasívne kešovanie

Koncové zariadenie s aktívnym kešovaním musí mať špeciálne vyhradenú sekciu filtra pri každej správe DII, ktorú je potrebné sledovať. Tým sa účinne obmedzuje množstvo obsahu, ktorý sa môže kešovať, prípadne vo veľmi malom množstve. Z tohto dôvodu môže koncové zariadenie zvoliť tzv. pasívnu stratégiu kešovania. To znamená, že koncové zariadenie sa ani nepokúša o nepretržité sledovanie správ DII, ale zakaždým, keď aplikácia chce získať objekt, ten v danom momente načíta aktuálny DII a kontroluje, či kešovaný obsah je stále platný. Aj keď je táto stratégia ukladania pomalšia v porovnaní s vrátením obsahu aplikácii, toto oneskorenie je zvyčajne podstatne menšie ako získanie obsahu z vysielaného toku.

B.5.2.1.3 Rýchlosť opakovania DII

Je potrebné poznamenať, že opis aktívneho a pasívneho kešovania je uvedený len pre informáciu a koncové zariadenie môže na implementáciu použiť ľubovoľnú stratégiu, ktorá spĺňa skôr stanovené normatívne obmedzenia. Je potrebné, aby vysielatelia nastavili opakovaciu frekvenciu správ DII tak, aby koncové zariadenie implementujúce stratégiu pasívneho kešovania poskytovalo očakávané výhody kešovania oproti koncovému zariadeniu, ktoré neimplementuje žiadne kešovanie.

B.5.2.2 Semitransparentné kešovanie

Semitransparentná úroveň kešovania umožňuje koncovému zariadeniu kešovať dáta a tiež vrátiť mierne neaktuálne údaje aplikácii. Výhodou tejto úrovne kešovania je, že umožňuje koncovým zariadeniam kešovať väčšie množstvo obsahu s primeraným využitím zdrojov, ale zároveň umožňuje dáta vrátiť obvykle okamžite aplikácii. Semitransparentná úroveň kešovania poskytuje menšiu záruku platnosti obsahu, ale nespôsobuje oneskorenie implikované stratégiou pasívneho kešovania s transparentnou úrovňou kešovania.

Pri vrátení obsahu z pamäte keš aplikácii musí koncové zariadenie zabezpečiť, aby číslo verzie kešovaného obsahu zodpovedalo číslu verzie uvedenému v platnej správe DII opisujúcej tento modul. Len čo DII bola prijatá, môže sa považovať za platnú minimálne počas 30 sekúnd a po uplynutí tohto času až do prijatia ďalšej príslušnej správy DII. Ak filtrácia danej správy DII nie je obnovená do skončenia tohto intervalu, stav správy DII je potrebné považovať za neznámy, pokiaľ opäť nebude prijatá.

Koncové zariadenia nesmú vrátiť semitransparentné kešované dáta v prípade, ak čakanie trvá dlhšie ako 30 sekúnd od prijatia príslušnej správy DII a opäť začne filtrácia správy DII. Ak nebude koncové zariadenie pokračovať vo filtrácii v priebehu 30 s, musí opäť stiahnuť príslušnú správu DII, ak chce správu DII používať na kontrolu platnosti kešovania.

B.5.2.2.1 Dôsledky – koncové zariadenie (informatívne)

Dôvody na výber hodnoty 30 s na čas čakania pri semitransparentnej úrovni kešovania sú odlišné od dôvodov na čas čakania 500 ms pri transparentnej úrovni. Pri tejto úrovni je čas čakania 30 s určený napríklad povolením, aby koncové zariadenia mali typicky platnú kópiu každej DII získaním každej DII v podobe cyklickej obsluhy pomocou jedinej sekcie filtra. Prirodzene, či tento cieľ možno dosiahnuť, závisí od rýchlosti opakovania správ DII a množstva obsahu, ktorý je kešovaný. Ak to nie je možné, koncové zariadenie môže použiť stratégiu pasívneho kešovania aj s danou úrovňou transparentnosti. Tieto stratégie sú len príklady a koncové zariadenie môže použiť akúkoľvek stratégiu, pokiaľ budú splnené skôr definované normatívne obmedzenia (to zahŕňa implementáciu bez kešovania, ak je voliteľná, ako aj spracovanie semitransparentnej úrovne rovnako ako spracovanie transparentnej úrovne).

B.5.2.3 Statické kešovanie

Ak sa používa úroveň transparentnosti statického kešovania, koncové zariadenie overí platnosť kešovaného obsahu z čísla verzie v správe DII v prípade, ak je použitá prvýkrát v priebehu životnosti aplikácie. Koncové zariadenie nemusí kontrolovať platnosť obsahu v priebehu životnosti aplikácie po prvom použití času.

B.5.2.3.1 Dôsledky pre vysielateľa (informatívne)

Obsah s touto úrovňou transparentnosti je vhodný ak je veľký statický obsah, ktorý je aktualizovaný len zriedka, pri ktorom sa možnosť aktualizácie obsahu nemusí zaznamenať v priebehu životnosti jednej aplikácie.

B.5.2.3.2 Dôsledky – koncové zariadenie (informatívne)

Koncové zariadenie môže aktualizovať obsah staticky kešovaných súborov, ak zistí, že boli aktualizované v karuseli, rovnako ako môže použiť akúkoľvek stratégiu kešovania, pokiaľ budú splnené uvedené normatívne obmedzenia (to zahŕňa implementáciu bez kešovania, ak je voliteľná, ako aj spracovanie statickej úrovne rovnako ako semitransparentnej úrovne alebo transparentnej úrovne).

B.5.3 Dynamická štruktúra karuselu

Objektový karusel môže meniť štruktúru v priebehu času, t. j. súbory a adresáre sa môžu pridať alebo odstrániť. Taktiež, moduly negarantujú prenos tých istých objektov počas životnosti karuselu. Prijímače nesmú predpokladať, že adresárové štruktúry sú statické, alebo že daná trasa bude patriť vždy tomu istému objektu. Všetky kešované informácie adresára sa musia kešovať podľa signalizovanej priority kešovania. To znamená, že pred použitím objektu, ktorý sa kešuje, musí prijímač potvrdiť k nemu trasu.

POZNÁMKA. – Overenie trasy nemusí nutne znamenať zakaždým sťahovanie trasy všetkých prvkov. Napríklad pri jednoduchom určení, že v rámci trasy sa žiadny z objektov nezmenil od posledného prepojenia, stačí potvrdiť, že sa trasa sama od seba nezmenila.

**Príloha C (normatívna):
Všeobecné aplikovanie západoeurópskeho súboru znakov**

Tabuľka C.1 definuje znakový súbor vhodný na používanie špecifikáciami platformy so zameraním sa na trh západnej Európy.

Tabuľka C.1 – Všeobecný západoeurópsky súbor znakov

Kód znaku v Unicode	Znak	Názov skriptu v Unicode
0020 to 007E		Základná latinská abeceda
00A0 to 00FF		Latin-1 doplnok
0100 to 017E		Rozšírená latinská abeceda (s výnimkou 017F)
01CD	Ă	Latinské veľké písmeno A s háčikom
01CE	ă	Latinské malé písmeno a s háčikom
02C6	ˆ	Modifikácia vokáľa
02C7	ˇ	Háčik (mandarínska čínština tretí akcent)
02C9	—	Modifikátor macron (mandarínska čínština tretí akcent)
02D8	˘	Guľatý háčik
02D9	˙	Bodka hore
02DA	˚	Krúžok hore
02DB	˛	Ogonek
02DC	˜	Malá vinovka
066B	.	Arabský desiatinný oddeľovač
1E80	Ŵ	Latinské veľké písmeno W s čiarkou nad vľavo
1E81	ŵ	Latinské malé písmeno w s čiarkou nad vľavo
1E82	Ŷ	Latinské veľké písmeno W s čiarkou nad vpravo
1E83	ŷ	Latinské malé písmeno w s čiarkou nad vpravo
1E84	Ẁ	Latinské veľké písmeno W s dvomi bodkami
1E85	ẁ	Latinské malé písmeno w s dvomi bodkami
1EF2	Ỳ	Latinské veľké písmeno Y s čiarkou nad vľavo
1EF3	ỳ	Latinské malé písmeno y s čiarkou nad vľavo
2007		Obrázok voľné miesto
2013	–	En pomlčka
2014	—	Em pomlčka
2018	ˆ	Dolná jednoduchá úvodzovka
2019	ˆ	Horná jednoduchá úvodzovka
201A	ˆ	Ľavé úvodzovky
201C	ˆ	Pravé úvodzovky
201D	ˆ	Pravá dvojité úvodzovka
201E	ˆ	Úvodzovky Low-9
2022	•	Koliesko
2026	...	Vynechanie písmena
2030	‰	Promile
2039	◀	Jednoduchá ostrá zátvorka ľavá
203A	▶	Jednoduchá ostrá zátvorka pravá
2044	/	Deliaca lomka
20AC	€	Znak eura
2122	™	Ochranná známka
2190	←	Šípka vľavo
2191	↑	Šípka nahor
2192	→	Šípka vpravo
2193	↓	Šípka nadol
2212	–	Znak mínus
2214	+	Znak plus s bodkou
2215	/	Deliaca lomka
221E	∞	Nekonečno
266B	♯	Spojené osminové noty
2713	✓	Začiarknutie
2717	×	Voľba X

**Príloha D (informatívna):
Literatúra**

"Platforma aplikácie OpenCable 1.0 Profil", OC-SP-OCAP1.0-I16-050803.

**Príloha NA (informatívna):
Doplňujúce skratky**

V technickej špecifikácii ETSI TS 102 809 V1.1.1 nie sú špecifikované tieto skratky:

TAP	Test anything protocol	protokol na všeobecné testovanie (pozri ETSI TR 101 202 v1.2.1, s.28)
BIOP	Broadcast Inter ORB Protocol	vysielací protokol vnútorného ORB
ORB	Object Request Broker	sprostredkovanie objektu na požiadanie

História

História dokumentu		
V1.1.1	Január 2010	Publikované