|  |  |
| --- | --- |
|  | **T I Ê U C H U Ẩ N Q U Ố C G I A** |

**TCVN**

**TCVN xxx:201x**

**ISO/IEC 14496-12:2012**

**Xuất bản lần 1**

**MÃ HÓA CÁC ĐỐI TƯỢNGHÌNH ẢNHÂM THANH – KHUÔN DẠNG FILE ĐA PHƯƠNG TIỆN THEO CHUẨN ISO**

***Coding of audio-visual objects — ISO base media file format***

HÀ NỘI – 201x

**Mục lục**

[1. Phạm vi áp dụng 10](#_Toc446058166)

[2. Tài liệu viện dẫn 10](#_Toc446058167)

[3. Thuật ngữ, định nghĩa, danh mục các từ viết tắt 11](#_Toc446058168)

[3.1. Thuật ngữ và định nghĩa 11](#_Toc446058169)

[3.2. 3.2. Ký hiệu và thuật ngữ viết tắt 13](#_Toc446058170)

[4. Cấu trúc của tệp tin hướng đối tượng 15](#_Toc446058171)

[4.1. Cấu trúc tệp tin 15](#_Toc446058172)

[4.2. Cấu trúc đối tượng 15](#_Toc446058173)

[4.3. Hộp Kiểu Tệp tin(File Type Box) 16](#_Toc446058174)

[4.3.1. Định nghĩa 16](#_Toc446058175)

[4.3.2. Cú pháp 16](#_Toc446058176)

[4.3.3. Ngữ nghĩa 17](#_Toc446058177)

[5. Những lưu ý khi thiết kế 17](#_Toc446058178)

[5.1. Cách thức sử dụng 17](#_Toc446058179)

[5.1.1. Giới thiệu 17](#_Toc446058180)

[5.1.2. Sự trao đổi 17](#_Toc446058181)

[5.1.3. Xây dựng nội dung 17](#_Toc446058182)

[5.1.4. Chuẩn bị cho streaming 18](#_Toc446058183)

[5.1.5. Trình chiếu cục bộ 18](#_Toc446058184)

[5.1.6. Bản trình diễn dạng luồng 18](#_Toc446058185)

[5.2. Các nguyên tắc thiết kế 19](#_Toc446058186)

[6. Cấu trúc tệp tin đa phương tiên cơ sởtheo chuẩn ISO 19](#_Toc446058187)

[6.1. Cấu trúc bản trình diễn 19](#_Toc446058188)

[6.1.1. Cấu trúc tệp tin 19](#_Toc446058189)

[6.1.2. Cấu trúc đối tượng 20](#_Toc446058190)

[6.1.3. Siêu dữ liệu và dữ liệu đa phương tiện 20](#_Toc446058191)

[6.1.4. Định danh rãnh 20](#_Toc446058192)

[6.2. Cấu trúc của siêu dữ liệu (các đối tượng) 20](#_Toc446058193)

[6.2.1. Hộp 20](#_Toc446058194)

[6.2.2. Các trường và các kiểu dữ liệu 21](#_Toc446058195)

[6.2.3. Thứ tự của hộp 22](#_Toc446058196)

[6.2.4. Bộ chỉ thị URI 26](#_Toc446058197)

[6.3. Định danh nhãn 26](#_Toc446058198)

[7. Hỗ trợ streaming 27](#_Toc446058199)

[7.1. Xử lý các giao thức streaming 27](#_Toc446058200)

[7.2. Rãnh chỉ dẫn giao thức 27](#_Toc446058201)

[7.3. Định dạng rãnh chỉ dẫn 28](#_Toc446058202)

[8. Cấu trúc hộp 29](#_Toc446058203)

[8.1. Cấu trúc tệp tin và các hộp thông thường 29](#_Toc446058204)

[8.1.1. Hộp Dữ liệu Đa phương tiện (Media Data Box) 29](#_Toc446058205)

[8.1.2. Hộp Không gian Trống (Free Space Box) 29](#_Toc446058206)

[8.1.3. Hộp Thông tin Tải xuống Liên tục (Progressive Download Information Box) 30](#_Toc446058207)

[8.2. Cấu trúc của movie 30](#_Toc446058208)

[8.2.1. Hộp Movie (Hộp Movie) 30](#_Toc446058209)

[8.2.2. Hộp Mào đầu Movie (Movie Header Box) 30](#_Toc446058210)

[8.3. Cấu trúc rãnh 32](#_Toc446058211)

[8.3.1. Hộp rãnh (Track Box) 32](#_Toc446058212)

[8.3.2. Hộp Mào đầu Rãnh (Track Header Box) 32](#_Toc446058213)

[8.3.3. Hộp Tham chiếu Rãnh (Track Reference Box) 34](#_Toc446058214)

[8.3.4. Hộp Nhóm Rãnh (Track Group Box) 35](#_Toc446058215)

[8.4. Cấu trúc rãnh đa phương tiện 35](#_Toc446058216)

[8.4.1. Hộp Đa phương tiện (MediaBox) 35](#_Toc446058217)

[8.4.2. Hộp Mào đầu Đa phương tiện (Media Header Box) 36](#_Toc446058218)

[8.4.3. Hộp Tham chiếu Xử lý (Handler Reference Box) 37](#_Toc446058219)

[8.4.4. Hộp Thông tin Đa phương tiện (Media Information Box) 38](#_Toc446058220)

[8.4.5. Hộp Mào đầu Thông tin Đa phương tiện (Media Information Header Box) 38](#_Toc446058221)

[8.5. Các bảng mẫu 40](#_Toc446058222)

[8.5.1. Hộp Bảng Mẫu(Sample Table Box) 40](#_Toc446058223)

[8.5.2. Hộp Miêu tả Mẫu (Sample Description Box) 41](#_Toc446058224)

[8.5.3. Hộp ưu tiên suy giảm (Degradation Priority Box) 49](#_Toc446058225)

[8.5.4. Hộp tần số lấy mẫu (Sample Scale Box) 50](#_Toc446058226)

[8.6. Cấu trúc thời gian của rãnh 50](#_Toc446058227)

[8.6.1. Hộp Thời gian Mẫu (Time to Sample Boxs) 50](#_Toc446058228)

[8.6.2. Hộp Mẫu Đồng bộ (Sync Sample Box) 54](#_Toc446058229)

[8.6.3. Hộp Mẫu Đồng bộ Vùng tối (Shadow Sync Sample Box) 55](#_Toc446058230)

[8.6.4. Hộp các Mẫu Độc lập và Dùng một lần (Independent and Disposable Samples Box) 56](#_Toc446058231)

[8.6.5. Hộp Biên tập (Edit Box) 57](#_Toc446058232)

[8.6.6. Hộp Danh sách Biên tập(Edit List Box) 58](#_Toc446058233)

[8.7. Câu trúc sắp xếp dữ liệu của rãnh 59](#_Toc446058234)

[8.7.1. Hộp Thông tin Dữ liệu (Data Information Box) 59](#_Toc446058235)

[8.7.2. Hộp Tham chiếu Dữ liệu (Data Reference Box) 59](#_Toc446058236)

[8.7.3. Các Hộp Kích thước Mẫu (Sample Size Boxes) 60](#_Toc446058237)

[8.7.4. Hộp Ánh xạ Mẫu vàoĐoạn (Sample To Chunk Box) 62](#_Toc446058238)

[8.7.5. Hộp Độ dịch chuyển Đoạn (Chunk Offset Box) 62](#_Toc446058239)

[8.7.6. Hộp CácBit Đệm (Padding Bits Box) 63](#_Toc446058240)

[8.7.7. Hộp Thông tin Mẫu con (Sub-sample Information Box) 64](#_Toc446058241)

[8.7.8. Hộp Kích cỡ Thông tin Bổ trợ Mẫu (Sample Auxiliary Information Sizes Box) 65](#_Toc446058242)

[8.7.9. Hộp Độ dịch chuyểnThông tin Bổ trợMẫu (Sample Auxiliary Information Offsets Box) 66](#_Toc446058243)

[8.8. Phân mảnh movie 67](#_Toc446058244)

[8.8.1. Hộp Movie Mở rộng (Movie Extends Box) 67](#_Toc446058245)

[8.8.2. Hộp Mào đầu Movie Mở rộng (Movie Extends Header Box) 68](#_Toc446058246)

[8.8.3. Hộp Rãnh Mở rộng (Track Extends Box) 68](#_Toc446058247)

[8.8.4. Hộp Phân mảnh Movie (Movie Fragment Box) 69](#_Toc446058248)

[8.8.5. Hộp Mào đầu Phân mảnh Movie (Movie Fragment Header Box) 69](#_Toc446058249)

[8.8.6. Hộp Phân mảnh Rãnh (Track Fragment Box) 70](#_Toc446058250)

[8.8.7. Hộp Mào đầu Phân mảnh Rãnh (Track Fragment Header Box) 70](#_Toc446058251)

[8.8.8. Hộp Xử lý Phân mảnh Rãnh (Track Fragment Run Box) 71](#_Toc446058252)

[8.8.9. Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie (Movie Fragment Random Access Box) 73](#_Toc446058253)

[8.8.10. Hộp Truy nhập Ngẫu nhiên Phân mảnh Rãnh (Track Fragment Random Access Box) 73](#_Toc446058254)

[8.8.11. Hộp Độ dịch chuyểnTruy nhập Ngẫu nhiên Phân mảnh Movie (Movie Fragment Random Access Offset Box) 74](#_Toc446058255)

[8.8.12. Hộp Thuộc tính Mở rộng Rãnh (Track Extension Properties Box) 75](#_Toc446058256)

[8.8.13. Hộp Thuộc tính Chuỗi Khởi động Thay thế (Alternative Startup Sequence Properties Box) 75](#_Toc446058257)

[8.8.14. Hộp Thời gian Giải mã Phân mảnh Rãnh (Track Fragment Decode Time Box) 76](#_Toc446058258)

[8.8.15. Hộp Chỉ định Mức (Level Assignment Box) 77](#_Toc446058259)

[8.8.16. Thông tin bổ trợ mẫu trong các phân mảnh movie 79](#_Toc446058260)

[8.9. Cấu trúc nhóm mẫu 79](#_Toc446058261)

[8.9.1. Giới thiệu 79](#_Toc446058262)

[8.9.2. Hộp Ánh xạ Mẫu vào Nhóm (Sample to Group Box) 79](#_Toc446058263)

[8.9.3. Hộp Mô tả Nhóm Mẫu (Sample Group Description Box) 80](#_Toc446058264)

[8.9.4. Đại diện của các cấu trúc nhóm trong các phân mảnh movie 82](#_Toc446058265)

[8.10. Dữ liệu người dùng 83](#_Toc446058266)

[8.10.1. Hộp Dữ liệu Người dùng (User Data Box) 83](#_Toc446058267)

[8.10.2. Hộp Bản quyền (Copyright Box) 83](#_Toc446058268)

[8.10.3. Hộp Chọn lựa Rãnh (Track Selection Box) 84](#_Toc446058269)

[8.11. Hỗ trợ siêu dữ liệu 86](#_Toc446058270)

[8.11.1. Hộp Siêu dữ liệu (Meta Box) 86](#_Toc446058271)

[8.11.2. Hộp XML (XML Box) 87](#_Toc446058272)

[8.11.3. Hộp Định vị Hạng mục (Item Location Box) 87](#_Toc446058273)

[8.11.4. Hộp Hạng mục Cơ bản (Primary Item Box) 90](#_Toc446058274)

[8.11.5. Hộp Bảo vệ Hạng mục (Item Protection Box) 91](#_Toc446058275)

[8.11.6. Hộp Thông tin Hạng mục (Item Information Box) 91](#_Toc446058276)

[8.11.7. Hộp chứa Siêu dữ liệu Bổ sung (Additional Metadata Container Box) 94](#_Toc446058277)

[8.11.8. Hộp Quan hệ Hộp siêu dữ liệu (Metabox Relation Box) 94](#_Toc446058278)

[8.11.9. Các dạng URL của Hộp Siêu dữ liệu 95](#_Toc446058279)

[8.11.10. Siêu dữ liệu tĩnh (Static Metadata) 96](#_Toc446058280)

[8.11.11. Hộp Dữ liệu Hạng mục (Item Data Box) 96](#_Toc446058281)

[8.11.12. Hộp Tham chiếuHạng mục (Item Reference Box) 97](#_Toc446058282)

[8.11.13. Siêu dữ liệu hình ảnh bổ trợ 98](#_Toc446058283)

[8.12. Hỗ trợ các luồng được bảo vệ 98](#_Toc446058284)

[8.12.1. Hộp Thông tin Lược đồ Bảo vệ (Protection Scheme Information Box) 99](#_Toc446058285)

[8.12.2. Hộp Định dạng Gốc (Original Format Box) 99](#_Toc446058286)

[8.12.3. Hộp thông tin IPMP (IPMP Info Box) 100](#_Toc446058287)

[8.12.4. Hộp Điều khiển IPMP (IPMP Control Box) 100](#_Toc446058288)

[8.12.5. Hộp Kiểu Lược đồ (Scheme Type Box) 100](#_Toc446058289)

[8.12.6. Hộp Thông tin Lược đồ (Scheme Information Box) 100](#_Toc446058290)

[8.13. Hỗ trợ định dạng vận chuyển tệp tin 101](#_Toc446058291)

[8.13.1. Giới thiệu 101](#_Toc446058292)

[8.13.2. Hộp Thông tin Hạng mục FD (FD Item Information Box) 101](#_Toc446058293)

[8.13.3. Hộp Phân vùng Tệp tin (File Partition Box) 102](#_Toc446058294)

[8.13.4. Hộp Dự trữ FEC (FEC Reservoir Box) 103](#_Toc446058295)

[8.13.5. Hộp Nhóm Phiên FD (FD Session Group Box) 104](#_Toc446058296)

[8.13.6. Hộp Ánh xạ Tên và Nhóm định danh (Group ID to Name Box) 105](#_Toc446058297)

[8.13.7. Hộp Dự trữ Tệp tin (File Reservoir Box) 106](#_Toc446058298)

[8.14. Rãnh con 106](#_Toc446058299)

[8.14.1. Giới thiệu 106](#_Toc446058300)

[8.14.2. Khả năng tương thích ngược 107](#_Toc446058301)

[8.14.3. Hộp Rãnh con (SubTrack Box) 107](#_Toc446058302)

[8.14.4. Hộp Thông tin Rãnh con (SubTrackInformation Box) 107](#_Toc446058303)

[8.14.5. Hộp Định nghĩa Rãnh con (Sub Track Định nghĩa Box) 109](#_Toc446058304)

[8.14.6. Hộp Nhóm Mẫu Rãnh con (Sub TrackSample Group Box) 109](#_Toc446058305)

[8.15. Yêu cầu đối với bộ tiền giải mã 109](#_Toc446058306)

[8.15.1. Tổng quan 109](#_Toc446058307)

[8.15.2. Chuyển đổi 110](#_Toc446058308)

[8.15.3. Hộp Thông tin Lược đồ Giới hạn (Restricted Scheme Information box) 110](#_Toc446058309)

[8.15.4. Lược đồ bố trí hình ảnh lập thể 111](#_Toc446058310)

[8.16. Các phân đoạn 112](#_Toc446058311)

[8.16.1. Giới thiệu 112](#_Toc446058312)

[8.16.2. Hộp Kiểu Phân đoạn (Segment Type Box) 112](#_Toc446058313)

[8.16.3. Hộp Chỉ số Phân đoạn (Segment Index Box) 113](#_Toc446058314)

[8.16.4. Hộp Chỉ số Phân đoạn con (Subsegment Index Box) 117](#_Toc446058315)

[8.16.5. Hộp Thời gian Tham chiếu Nguồn (Producer Reference Time Box) 118](#_Toc446058316)

[8.17. Hỗ trợ các rãnh không hoàn chỉnh 119](#_Toc446058317)

[8.17.1. Tổng quan 119](#_Toc446058318)

[8.17.2. Chuyển đổi 120](#_Toc446058319)

[8.17.3. Hộp Thông tin Rãnh Hoàn chỉnh (Complete Track Information Box) 120](#_Toc446058320)

[9. Định dạng rãnh chỉ dẫn 121](#_Toc446058321)

[9.1. Định dạngrãnh chỉ dẫn RTP và SRTP 121](#_Toc446058322)

[9.1.1. Giới thiệu 121](#_Toc446058323)

[9.1.2. Định dạng miêu tả mẫu 121](#_Toc446058324)

[9.1.3. Định dạngmẫu 123](#_Toc446058325)

[9.1.4. Thông tin SDP 126](#_Toc446058326)

[9.1.5. Thông tin thống kê 127](#_Toc446058327)

[9.2. Định dạngrãnh chỉ dẫnALC/LCT và FLUTE 128](#_Toc446058328)

[9.2.1. Giới thiệu 128](#_Toc446058329)

[9.2.2. Nguyên tắc thiết kế 128](#_Toc446058330)

[9.2.3. Định dạng miêu tả mẫu 130](#_Toc446058331)

[9.2.4. Định dạng mẫu 130](#_Toc446058332)

[9.3. Định dạngrãnh chỉ dẫn truyền tảiMPEG-2 134](#_Toc446058333)

[9.3.1. Giới thiệu 134](#_Toc446058334)

[9.3.2. Nguyên tắc thiết kế 134](#_Toc446058335)

[9.3.3. Định dạng miêu tả mẫu 136](#_Toc446058336)

[9.3.4. Định dạngmẫu 138](#_Toc446058337)

[9.3.5. Rãnh chỉ dẫnluồng truyền tải MPEG-2 được bảo vệ 140](#_Toc446058338)

[9.4. Rãnh chỉ dẫn tiếp nhận RTP, RTCP, SRTP và SRTCP 141](#_Toc446058339)

[9.4.1. Rãnh chỉ dẫn tiếp nhận RTP 141](#_Toc446058340)

[9.4.2. Rãnh chỉ dẫn tiếp nhận RTCP 144](#_Toc446058341)

[9.4.3. Rãnh chỉ dẫn tiếp nhận SRTP 145](#_Toc446058342)

[9.4.4. Rãnh chỉ dẫn tiếp nhận SRTCP 147](#_Toc446058343)

[9.4.5. Rãnh chỉ dẫn tiếp nhận RTP được bảo vệ 148](#_Toc446058344)

[9.4.6. Thủ tục ghi 148](#_Toc446058345)

[9.4.7. Thủ tục phân tích 148](#_Toc446058346)

[10. Các nhóm mẫu 149](#_Toc446058347)

[10.1. Các điểm khôi phục truy nhập ngẫu nhiên 149](#_Toc446058348)

[10.1.1. Định nghĩa 149](#_Toc446058349)

[10.1.2. Cú pháp 149](#_Toc446058350)

[10.1.3. Ngữ nghĩa 149](#_Toc446058351)

[10.2. Các nhóm chia sẻ tỉ lệ 150](#_Toc446058352)

[10.2.1. Giới thiệu 150](#_Toc446058353)

[10.2.2. Đầu vào mục nhóm mẫu chia sẻ tỉ lệ 151](#_Toc446058354)

[10.2.3. Mối quan hệ giữa các rãnh 152](#_Toc446058355)

[10.2.4. Phân bổ tốc độ bit 152](#_Toc446058356)

[10.3. Các chuỗi khởi động luân phiên 153](#_Toc446058357)

[10.3.1. Định nghĩa 153](#_Toc446058358)

[10.3.2. Cú pháp 153](#_Toc446058359)

[10.3.3. Ngữ nghĩa 153](#_Toc446058360)

[10.3.4. Một số ví dụ 154](#_Toc446058361)

[10.4. Nhóm mẫu điểm truy nhập ngẫu nhiên 156](#_Toc446058362)

[10.4.1. Định nghĩa 156](#_Toc446058363)

[10.4.2. Cú pháp 156](#_Toc446058364)

[10.4.3. Ngữ nghĩa 156](#_Toc446058365)

[10.5. Gộp nhóm mẫu theo mức thời gian 156](#_Toc446058366)

[10.5.1. Định nghĩa 156](#_Toc446058367)

[10.5.2. Cú pháp 156](#_Toc446058368)

[10.5.3. Ngữ nghĩa 157](#_Toc446058369)

[11. Khả năng mở rộng 157](#_Toc446058370)

[11.1. Các đối tượng 157](#_Toc446058371)

[11.2. Các định dạng lưu trữ 158](#_Toc446058372)

[11.3. Các định dạng tệp tinphái sinh 158](#_Toc446058373)

[12. Phụ lục A (Tham khảo) Tổng quan 159](#_Toc446058374)

[13. Phụ lục B (Tham khảo) Tuyên bố về quyền sáng chế 163](#_Toc446058375)

[14. Phụ lục C (Tham khảo) Hướng dẫn tạo định dạng tệp tin dựa trên tiêu chuẩn này 164](#_Toc446058376)

[15. Phụ lục D (Tham khảo) Tổ chức chứng nhận 172](#_Toc446058377)

[16. Phụ lục E (Quy định) Các nhãn định dạng tệp tin 175](#_Toc446058378)

[17. Phụ lục F (Tham khảo) Các dạng siêu dữ liệu được dán nhãn URI 181](#_Toc446058379)

[19. Phụ lục G (Tham khảo) Quá trình xử lý các luồng RTP và rãnh chỉ dẫn tiếp nhận 183](#_Toc446058380)

[21. Phụ lục H (Quy định) Các điểm truy nhập luồng 199](#_Toc446058381)

[22. Phụ lục I (Quy định) Đăng ký dạng MIME của các phân đoạn 201](#_Toc446058382)

[23. Danh mục tài liệu tham khảo 202](#_Toc446058383)

**Lời nói đầu**

TCVN xxx:201x hoàn toàn tương đương với ISO/IEC 14496-12:2012 Information technology – Coding of audio-visual objects – Part 12: ISO base media file format (Công nghệ thông tin – Mã hóa các đối tượng hình ảnh âm thanh – Phần 12: Định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO).

TCVN xxx:201x do Học viện Công nghệ Bưu chính Viễn thông biên soạn, Bộ Thông tin và Truyền thông đề nghị,Tổng cục Tiêu chuẩn Đo lường Chất lượng thẩm định, Bộ Khoa học và Công nghệ công bố.

|  |  |
| --- | --- |
| **T I Ê U C H U Ẩ N Q U Ố C G I A** | **TCVN XXX:201X** |

**Mã hóa các đối tượng hình ảnh âm thanh – Khuôn dạng file đa phương tiện theo chuẩn ISO**

*Coding of audio-visual objects — ISO base mediafileformat*

# Phạm vi áp dụng

Tiêu chuẩn này quy định về định dạng của tệp tinđa phương tiệncơ sở theo chuẩn ISO, đây là định dạng được dùng để xây dựng nên các định dạng tệp tinđa phương tiệncụ thể hơn. Định dạng này chứa thông tin về thời gian, cấu trúc và thông tin đa phương tiện của chuỗi dữ liệu đa phương tiện tuân theo thời gian, ví dụ như các bản trình diễn hình ảnh, âm thanh.

Tiêu chuẩn này được xây dựng dựa trên tiêu chuẩn ISO/IEC 14496-12, ISO/IEC 14496-12 lại là một phẩn của bộ tiêu chuẩn ISO/IEC 14496 áp dụng cho MPEG-4, nội dung kỹ thuật của ISO/IEC 14496-12 giống với tiêu chuẩn ISO/IEC 15444-12, áp dụng cho JPEG-2000.

Các tổ chức, cá nhân tại Việt Nam có thể áp dụng tiêu chuẩn này để xây dựng nên các định dạng tệp tinđa phương tiện, hoặc để nghiên cứu và phát triển các thiết bị phần cứng, phần mềm liên quan đến việc thu phát, xử lý âm thanh hình ảnh của các tệp tin có định dạng được xây dựng nên từ tiêu chuẩn cơ sở này.

# Tài liệu viện dẫn

Các tài liệu viện dẫn sau rất cần thiết cho việc áp dụng tiêu chuẩn này. Đối với các tài liệu viện dẫn ghi năm công bố thì áp dụng phiên bản được nêu. Đối với các tài liệu viện dẫn không ghi năm công bố thì áp dụng phiên bản mới nhất, bao gồm cả các sửa đổi, bổ sung (nếu có).

ISO 639-2:1998, Bộ quy tắc cho tên của các ngôn ngữ - Phần 2: Mã Alpha-3.

ISO/IEC 9834-8:2005, Công nghệ thông tin – Kết nối các hệ thống mở - Các thủ tục liên quan đến quyền đăng ký OSI: Việc tạo và đăng ký UUID, cách thức sử dụng các thành phần định danh đối tượng ASN.1.

ISO/IEC 11578:1996, Công nghệ thông tin – Kết nối các hệ thống mở - Gọi thủ tục từ xa (RPC).

ISO/IEC 14496-1:2010: Công nghệ thông tin – Mã hóa các đối tượng âm thanh hình hảnh – Phần 1 – Các hệ thống.

ISO/IEC 14496-10, Công nghệ thông tin – Mã hóa các đối tượng âm thanh hình hảnh – Phần 10: Mã hóa hình ảnh tiên tiến.

ISO/IEC 14496-14, Công nghệ thông tin – Mã hóa các đối tượng âm thanh hình hảnh – Phần 14: định dạng tệp tin MP4.

ISO/IEC 15444-1, Công nghệ thông tin – Hệ thống mã hóa hình ảnh JPEG 2000: Hệ thống mã hóa lõi.

ISO/IEC 15444-3, Công nghệ thông tin – Hệ thống mã hóa hình ảnh JPEG 2000: Ảnh động JPEG 2000.

ISO/IEC 15938-1, Công nghệ thông tin -Giao diện miêu tả nội dung đa phương tiện – Phần 1: Các hệ thống.

ISO/IEC 23001-1, Công nghệ thông tin –Các công nghệ của hệ thống MPEG – Phần 1: định dạng MPEG nhị phân của XML.

ISO/IEC 23002-3, Công nghệ thông tin – Các công nghệ hình ảnh MPEG – Phần 3: Hình ảnh phụ trợ và các thông tin bổ sung.

ISO/IEC 29199-2:2012, Công nghệ thông tin –Hệ thống mã hóa hình ảnh JPEG XR – Phần 2: Tiêu chuẩn mã hóa hình ảnh.

ISO 15076-1:2010, Kiểm soát màu của ảnh – Kiến trúc, định dạng và cấu trúc dữ liệu – Phần 1: Dựa trên ICC.1:2010.

IETF RFC 2045, Mở rộng Internet Mail đa mục đích phần 1: định dạng cơ bản của tin nhắn trên Internet, FREED, N. và BORENSTEIN, N., tháng 11 năm 1996.

IETF RFC 2046, Mở rộng Internet Mail đa mục đích phần 2: Các dạng đa phương tiện, FREED, N. và BORENSTEIN, N., tháng 11 năm 1996.

IETF RFC 3550, RTP: Giao thức truyền tải các ứng dụng thời gian thực, SCHULZRINNE, H. et al., tháng 7 năm 2003.

IETF RFC 3711,”Giao thức truyền tải thời gian thực an toàn (SRTP)”, BAUGHER, M. et al., tháng tư năm 2004.

IETF RFC 5052, Khối hiệu chỉnh phòng lỗi, WATSON, M. et al., tháng 8 năm 2007.

IETF RFC 5905, Giao thức quản lý thời gian mạng phiên bản 4: Đặc tả giao thức và các thuật toán, MILLS, D., et al, tháng 6 năm 2010.

SMIL 1.0 “Đặc tả ngôn ngữ tích hợp đa phương tiện đồng bộ 1.0(SMIL)”, <http://www.w3.org/TR/REC-smil/>.

Rec. ITU-R TF.460-6, Sự phát xạ của tín hiệu theo thời gian và tần số chuẩn (Định nghĩa của UTC được đề cập đến trong Phụ lục 1).

# Thuật ngữ, định nghĩa, danh mục các từ viết tắt

## Thuật ngữ và định nghĩa

Trong tiêu chuẩn này sử dụng các thuật ngữ và định nghĩa sau:

**3.1.1**

**Hộp** (box)

Là khối cấu trúc hướng đối tượng được định nghĩa bằng một định danh và độ dài đơn nhất.

LƯU Ý: trong một số tiêu chuẩn, hộp còn được gọi là nguyên tử (atom).

**3.1.2**

**Đoạn** (chunk)

Là tập các mẫu liền kề trong một rãnh.

**3.1.3**

**Hộp chứa** (container box)

Là hộp có mục đích duy nhất là chứa và nhóm một tập các hộp có liên quan.

LƯU Ý: Các hộp chứa thường không được tạo ra từ hộp đầy đủ (fullbox).

**3.1.4**

**Rãnh chỉ dẫn** (hint track)

Làkiểu rãnh đặc biệt, không chứa dữ liệu đa phương tiện mà là các chỉ dẫn cho việc đóng gói một hoặc nhiều rãnh vào một kênh truyền streaming.

**3.1.5**

**Công cụ tạo chỉ dẫn** (hinter)

Là công cụ dùng để thêm một hoặc nhiều rãnh chỉ dẫn vào tệp tin chỉ chứa đa phương tiện.

**3.1.6**

**Tệp tin đa phương tiện cơ sở theo chuẩn ISO** (ISO Base Media File)

Là tệp tin tuân theo định dạng được miêu tả trong tiêu chuẩn này.

**3.1.7**

**Phân đoạn con lớp lá** (leaf subsegment)

Là phân đoạn con không chứa bất kỳ thông tin chỉ mục nào hỗ trợ việc phân chia thành các phân đoạn nhỏ hơn nữa.

**3.1.8**

**Hộp dữ liệu đa phương tiện** (media data box)

Là hộp có thể chứa dữ liệu đa phương tiện thực dùng để trình diễn (‘*mdat*’).

**3.1.9**

**Hộp Movie** (Movie box)

Là hộp chứa các hộp con định nghĩa siêu dữ liệu cho một trình diễn (‘*moov*’).

**3.1.10**

**Trình diễn** (presentation)

Là một hoặc nhiều chuỗi chuyển động, có thể bao gồm cả âm thanh.

**3.1.11**

**Điểm truy nhập ngẫu nhiên** (random access point (RAP))

Là điểm trỏ tới mẫu trong một rãnh, bắt đầu tại điểm ISAU của điểm truy nhập luồng SAP kiểusố 1, 2 hoặc 3 như miêu tảtrong Phụ lục H.

LƯU Ý:Thông thường, có thể giải mã cơ bản xác được các mẫu đã được sắp xếp đúng thứ tự.

**3.1.12**

**Điểm phục hồi ngẫu nhiên** (random access recovery point)

Là điểm trỏ tới mẫu trong một rãnh với thời gian trình diễn bằng TSAP của điểm truy nhập luồng SAP kiểu số 4 như miêu tả trong Phụ lục H.

LƯU Ý:Thông thường, một mẫu có thể được giải mã cơ bản xác sau khi giải mã một số mẫu trước đó, kiểu giải mã này nhiều khi được gọi là làm mới giải mã từng bước (gradual decoding refresh).

**3.1.13**

**Mẫu** (sample)

Là toàn bộ dữ liệu liên kết bởi một nhãn thời gian duy nhất.

LƯU Ý 1:2 mẫu trong cùng một rãnh không bao giờ có cùng một nhãn thời gian.

LƯU Ý 2: Trong các rãnh không có chỉ dẫn, một mẫu có thể là một khung hình, một chuỗi các khung hình,hoặc một đoạn âm thanh; trong các rãnh chỉ dẫn, một mẫu xác định định dạng của một hoặc nhiều gói streaming.

**3.1.14**

**Mô tả mẫu** (sample description)

Mô tả định dạng của một số mẫu trong một rãnh.

**3.1.15**

**Bảng mẫu** (sample table)

Là bảng thể hiện sự xắp xếp của các mẫu trong một rãnh theo thời gian và theo cấu trúc vật lý.

**3.1.16**

**Mẫu đồng bộ** (sync sample)

Là mẫu nằm trong rãnh, bắt đầu tại vị trí ISAU của điểm truy nhập luồng SAP kiểu số 1 hoặc 2 như miêu tảtrong Phụ lục H.

LƯU Ý: Mẫu đồng bộ có thể được định nghĩa một cách không cơ bản thức là mẫu khởi đầu của một chuỗi các mẫu độc lập mới; nếu việc giải mã bắt đầu tại mẫu đồng bộ thì bản thân mẫu đó cùng với các mẫu tiếp theo trong thứ tự giải mã có thể được giải mã đúng, và chuỗi mẫu được giải mã đúng đó sẽ thể hiện đúng nội dung cần trình diễn, bắt đầu tại mẫu được giải mã có thời gian hình thành sớm nhất; một định dạng đa phương tiện có thể định nghĩa một cách cơ bản xác hơn mẫu đồng bộ của định dạng đó.

**3.1.17**

**Phân đoạn** (segment)

Là một phần của tệp tin có định dạng đa phương tiện cơ sở theo chuẩn ISO, trong đó có chứa cả (a) mộtHộp Movie với dữ liệu đa phương tiện và các hộp kháccó liên quan hoặc (b) một hay một số Hộp Phân mảnh Movie, dữ liệu đa phương tiện và các hộp khác có liên quan.

**3.1.18**

**Phân đoạn con** (subsegment)

Là một phần của một phân đoạn được tạo ra từ các Hộp Phân mảnh Movie, phân đoạn con cũng có đặc điểm của một phân đoạn.

**3.1.19**

**Rãnh** (track)

Là chuỗi các mẫu liên tục theo thời gian có liên hệ với nhau trong một tệp tinđa phương tiệncơ sở theo chuẩn ISO.

LƯU Ý: Đối với dữ liệu đa phương tiện, một rãnhlà một chuỗi hình ảnh hoặc âm thanh; đối với các rãnh chỉ dẫn, một rãnhlà một kênh streaming.

**3.1.20**

**Streaming**

Là một phương pháp truyền và nhận dữ liệu (đặc biệt đối với âm thanh và hình ảnh), thông qua mạng máy tính, một cách đều đặt, liên lục và cho phép bên thu trình chiếu trong khi nhận dữ liệu mới.

## 3.2. Ký hiệu và thuật ngữ viết tắt

Các thuật ngữ viết tắt dưới đây được sử dụng trong tài liệu này.

|  |  |  |
| --- | --- | --- |
| ALC | Asynchronous Layered Coding | Mã hóa lớp không đồng bộ |
| DVI | Digital Visual Interface | Giao diện hình ảnh số |
| FD | File Delivery | Vận chuyển tệp tin |
| FDT | File Delivery Table | Bảng vận chuyển tệp tin |
| FEC | Forward Error Correction | Sửa lỗi trước |
| FLUTE | File Delivery over Unidirectional Transport | Vận chuyển tệp tin qua truyền tải đơn hướng |
| IANA | Internet Assigned Numbers Authority | Tổ chức cấp phát số hiệu Internet |
| IEC | International Electrotechnical Commission | Uỷ ban Kỹ thuật Điện Quốc tế |
| IETF | Internet Engineering Task Force | Nhóm đặc trách kỹ thuật Internet |
| IP | Internet Protocol | Gian thức Internet |
| ISO | International Organization for Standardization | Tổ chức tiêu chuẩn hoá quốc tế |
| ITU | International Telecommunication Union | Liên minh Viễn thông Quốc tế |
| LCT | Layered Coding Transport | Truyền tải mã hóa phân lớp |
| MBMS | Multimedia Broadcast/Multicast Service | Dịch vụ Broadcast/Multicast đa phương tiện |
| MIME | Multi-Purpose Internet Mail Extensions | Chuẩn thư điện tử Internet mở rộng sử dụng cho nhiều mục đích khác nhau |
| MPEG | Motion Picture Expert Group | Nhóm chuyên gia về ảnh động |
| RTP | Real-time Transport Protocol | Giao thức truyền tải thời gian thực |
| SDL | Syntax Description Language | Ngôn ngữ miêu tả cú pháp |
| URI | Uniform Resource Identifier | Định danh tài nguyên thống nhất |
| URL | Uniform Resource Locator | Định vị Tài nguyên thống nhất |
| UTC | Universal Time Coordinated | Giờ phối hợp quốc tế |
| UUID | Universal Unique Identifier | Định danh duy nhất toàn cầu |

# Cấu trúc của tệp tin hướng đối tượng

## Cấu trúc tệp tin

Các tệp tin được tạo bởi một loạt các đối tượng mà theo tiêu chuẩn này được gọi là các hộp. Tất cả dữ liệu đều được chứa trong các hộp. Dữ liệu trong tệp tincó thể bao gồm cảchứ ký khởi tạo ứng với định dạng tệp tin cụ thể.

Tất cả các tệp tin có cấu trúc hướng đối tượng tuân theo trong phần này của tiêu chuẩn (tất cả các tệp tin hướng đối tượng) đều chứa một Hộp Kiểu Tệp tin (File Type Box).

## Cấu trúc đối tượng

Đối tượng được đề cập đến ở đây cơ bản là hộp.

Mỗi hộp đều có phần mào đầu chỉ rõ kích cỡ và kiểucủa hộp. Phần mào đầu này hỗ trợ cả kích cỡ thu gọn và mở rộng (32 bit hoặc 64 bit) cũng như các kiểu thu gọn hoặc mở rộng (32 bit hoặc mã định danh đơn nhất phổ biến đầy đủ, tức là UUID). Các hộp tuân theo tiêu chuẩn này đều sử dụng kiểu thu gọn (32 bit) và hầu hết các hộp đều sử dụng kích cỡ thu gọn (32 bit). Duy chỉ có Hộp Dữ liệu Đa phương tiện (Media Data Box) là có kích cỡ 64 bit.

Kích cỡ được đề cập đến ở đây là kích cỡ của toàn bộ hộp, bao gồm cả kích cỡ và kiểu của mào đầu, các trường và tất cả các hộp chứa trong đó. Điều này thuận tiện cho việc phân tích tệp tin.

Các hộp được miêu tả thông qua ngôn ngữ miêu tả cú pháp (SDL) được định nghĩa trong MPEG-4 (xem 2). Phần ghi chú trong các đoạn mã mang ý nghĩa bổ sung thông tin cho đoạn mã đó.

Các trường của các đối tượng được lưu trữ theo thứ tự các byte có trọng số lớn nhất được lưu trước, kiểu lưu này giống như việc trữ dữ liệu trong máy tính (dạng big-endian). Trong trường hợp các trường nhỏ hơn hoặc lớn hơn một byte, các bit sẽ được sắp xếp theo thứ tựtừ bit có trọng số lớn nhất đến bit có trọng số nhỏ nhất. Ví dụ, trường thứ nhất có hai bit và trường kế tiếp có sáu bit thì hai bit của trường đầu tiên sẽ là các bit có thứ tự cao của byte được ghép lại từ hai trường này.

*aligned(8) class Box (unsigned int(32) boxtype, optional unsigned int(8)[16] extended\_type) {*

*unsigned int(32) size;*

*unsigned int(32) type = boxtype;*

*if (size==1) {*

*unsigned int(64) largesize;*

*} else if (size==0) {*

*// box extends to end offile*

*}*

*if (boxtype==‘uuid’) {*

*unsigned int(8)[16] usertype = extended\_type;*

*}*

*}*

Trong đó:

*size*có kiểu số nguyên, chỉ số lượng byte có trong hộp, bao gồm tất cả các trường và các hộp chứa trong đó; nếu size bằng 1 thì kích cỡ thực tế của hộp cơ bản là trường *largesize*, nếu *size* bằng 0 thì hộp này là hộp cuối cùng của tệp tin và dữ liệu của nó được mở rộng tới hết tệp tin (thường chỉ sử dụng cho Hộp Dữ liệu Đa phương tiện).

*type* xác định kiểu của hộp, các hộp tiêu chuẩn sử dụng dạng thu gọn, thường là bốn ký tự có thể in được, thuận lợi cho việc nhận dạng và được miêu tả trong các hộp bên dưới. Với kiểu người dùng tự định nghĩa thì trường đó được thiết lập là “*uuid* “.

Các hộp không nhận dạng được kiểu sẽ bị bỏ qua.

Nhiều đối tượng cũng có thêm trường phiên bản và cờ:

*aligned(8) class FullBox(unsigned int(32) boxtype, unsigned int(8) v, bit(24) f) extends Box(boxtype) {*

*unsigned int(8)version = v;*

*bit(24) flags = f;*

*}*

Trong đó:

*version*có kiểu số nguyên chỉ phiên bảncủa hộp.

*flags*thể hiện giá trị của các cờ.

Các hộp không rõ phiên bản sẽ bị bỏ qua.

## Hộp Kiểu Tệp tin(File Type Box)

### Định nghĩa

Loại hộp:‘*ftyp*’.

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: bắt buộc.

Số lượng: một(xem các ngoại lệ bên dưới).

Các tệp tintuân theo phiên bản này của tiêu chuẩn phải có một hộp kiểutệp tin. Để tương thích với phiên bản cũ của tiêu chuẩn này, các tệp tin có thể vẫn tuân theo tiêu chuẩn này nhưng không có hộp chứa kiểutệp tin. Các tệp tin không có hộp chứa kiểu tệp tinsẽ được đọc như làtệp tincó chứa *FTYP* với *Major\_brand=‘mp41’*, *minor\_version=0* và nhãnlà *‘mp41’*.

Tệp tinđa phương tiệncó cấu trúc tuân theo phần này của tiêu chuẩn có thể tương thích với các đặc tả kỹ thuật khác nhau, do vậy, không phải lúc nào cũng có thể xác định “kiểu “hoặc “nhãn “duy nhất cho một tệp tin. Điều đó có nghĩa rằng việc sử dụng phần mở rộng của tên tệp tin và dạng MIME (Multipurpose Internet Mail Extension) bị giới hạn lại đôi chút.

Hộp Kiểu Tệp tincần được đưa vào tệp tin càng sớm càng tốt (ví dụ: ngay sau phần chữ ký bắt buộc và trước các hộp có kích cỡ thay đổi, như mộtHộp Movie, Hộp Dữ liệu Đa phương tiệnhoặc Hộp Không giantrống). Hộp Kiểu Tệp tinxác định chỉ tiêu kỹ thuật phù hợp nhất đối vớitệp tin,đối với phiên bản thứ cấp của chỉ tiêu kỹ thuật đó, cũng như một tập các chỉ tiêu kỹ thuật mà tệp tin này phải tuân theo. Những đối tượng sử dụng định dạng này nên quan tâm đến tất cả các chỉ tiêu kỹ thuật mà tệp tin này tương thích. Bất kỳ thay đổi nào liên quan đến nhãn thì cầnphải đăng ký bởi nhãn nhận dạng mới nếu không được đăng ký thì không thể nhận dạng được và không thể kiểm tra được tính tương thích khi đưa nhãn mới đó vào chỉ tiêu kỹ thuật.

Phiên bản thứ cấp (minor version) chỉ mang tính chất cung cấp thông tin. Phiên bản này không xuất hiện trong các nhãn tương thích và không được sử dụng để xác định sự tuân thủ của tệp tintheo một chuẩn nào đó. Phiên bản thứ cấp có thể cung cấp thêm các nhận dạng cơ bản xác hơn của các tiêu chuẩn cơ bản nhằm mục đích giám sát, gỡ rối hoặc cải thiện việc giải mã.

Các tệp tin thường được nhận dạng bởi phần mở rộng (ví dụ như với phần đuôi tệp tin hoặc dạng mime), đó cơ bản là dấu hiệu nhận dạng dễ nhận thấy nhất.

Phần này của tiêu chuẩn không đề cập đến nhãn cụ thể mà sẽ được đề cập đến trong mục 6.3 và Phụ lục E.

### Cú pháp

*aligned(8) class FileTypeBox extends Box(‘ftyp’) {*

*unsigned int(32) major\_brand;*

*unsigned int(32) minor\_version;*

*unsigned int(32)compatible\_brands[];// to end of the box*

*}*

### Ngữ nghĩa

Hộp này xác định các chỉ tiêu kỹ thuật mà tệp tincần phải tuân theo.

Mỗi nhãn(brand) là một mã gồm 4 ký tự có thể in được, đã được đăng ký với tổ chức ISO, nhãn đó xác định cơ bản xác một tính năng nhất định.

*major\_brand* là mã định danh nhãn.

*minor\_version*có kiểu số nguyên chỉ phiên bản thứ cấp của *major\_brand*.

*compatible\_brands* là một danh sách các nhãn.

# Những lưu ý khi thiết kế

## Cách thức sử dụng

### Giới thiệu

Định dạngtệp tin được đề cập đến trong tiêu chuẩn này là định dạng cơ bản được sử dụng trong các trường hợp khác nhau. Do tính đa dạng như vậy nên định dạng này có thể được sử dụng theo các cách khách nhau cũng như trong các trường hợp khác nhau trong quá trình thiết kế thực tế nói chung.

### Sự trao đổi

Khi được sử dụng như là một định dạng trao đổi (có thể đọc được trên các phần mềm, trên các nền tảng khác nhau,...), các tệp tin thường thuộc dạng tệp tinđộc lập (tức là dữ liệu đa phương tiện nằm cơ bản trong tệp tin đó chứ phải nằm trong các tệp tin khác), khi đó tệp tinchỉ chứa dữ liệu đa phương tiện được sử dụng trong quá trình trình diễn và không chứa bất cứ thông tin nào liên quan đến streaming. Điều đó giúptệp tin nhỏ gọn, độc lập với giao thức, chứa dữ liệu đa phương tiện gốc và các thông tin cần thiết cho việc sử dụng các tệp tin này.

Hình vẽ dưới đây đưa ra một ví dụ về một tệp tintrao đổi, trong đó có 2 luồng dữ liệu.



Hình 1 - Ví dụ về tệp tin trao đổi đơn giản

### Xây dựng nội dung

Trong quá trình xây dựng nội dung,các tính năng của định dạng tệp tin này có thể được sử dụng để tạo ra các khả năng ứng dụng khác nhau, cụ thể là:

* Khả năng lưu các luồng dữ liệu riêng biệt (không chồng lấn), có thể nằm trong các tệp tin khác nhau;
* Khả năng cho phép thay đổitrên một bản trình diễn chứa dữ liệu đa phương tiện và các luồng dữ liệu khác nhau (ví dụ: chỉnh sửa rãnh âm thanh của định dạng không nén để đồng bộ với rãnh hình ảnh đã có trước đó).

Các đặc tính nêu trên cho thấy các bản trình diễn có thể được chuẩn bị trước, được sửa đổi, phát triển và tích hợp thêm nội dung mà không phải ghi lại nhiều lần, điều này cần thiết trong trường hợp cần phải ghép dữ liệu và xóa dữ liệu không sử dụng đến, ngoài ra cũng không phải thực hiện nhiều lần việc mã hóa và giải mã dữ liệu trong trường hợp dữ liệu cần lưu ở dạng mã hóa.

Trong hình vẽ dưới đây thể hiện việc tạo nội dung cho tệp tin.



Hình 2–Tệp tin tạo lập nội dung

### Chuẩn bị cho streaming

Để hỗ trợ cho việc streaming, trong tệp tin được truyền đi phải có thông tin để hướng dẫn cho máy chủ streaming biết làm thế nào để thực hiện việc gửi thông tin đi. Việc truyền xen kẽcả thông tin hướng dẫn và dữ liệu đa phương tiệnsẽ giúp đạt hiệu quả tốt hơn vì có thể sẽ giảm được độ phức tạp do việc tìm kiếm khi thực hiện trình diễn. Việc lưu dữ liệu đa phương tiện ban đầu rất quan trọng vì dữ liệu đó có thể được dùng để xác nhận, để chỉnh sửa lại hoặc tái sử dụng. Ngoài ra, nếu một tệp tin có thể được dùng cho nhiều giao thức khác nhau thì hiệu quả sử dụng sẽ cao hơn so với việc mỗi tệp tin chỉ dùng cho một giao thức.

### Trình chiếu cục bộ

Việc trình diễn cục bộ(ví dụ như xem trực tiếp tệp tin lưu trên thiết bịchứ không phải xem trực tuyến)là rất cần thiết. Hình thức xem cục bộ này đã được thực hiện từ lâu và khá phổ biến, ví dụ xem những nội dung được phân phối dạng đơn lẻnhư nội dung trên CD ROM hay DVD ROM, hoặc để kiểm tra nội dung trong quá trình dựng hay để kiểm tra nội dung trong máy chủ streaming. Việc xem cục bộ này cần hỗ trợ tính năng truy nhập ngẫu nhiên đối với toàn bộ nội dung cần trình diễn. Nếu nội dung được lưu trên CD hay DVD ROM thì việc đan xen dữ liệu là cần thiết bởi việc tìm kiếm có thể rất chậm.

### Bản trình diễn dạng luồng

Khi máy chủ xử lý một tệp tin để tạo ra một luồng (stream), luồng được tạo ra phải tuân thủ các đặc tả kỹ thuật của các giao thức được sử dụng và không lưu lại dấu vết các thông tin định dạng tệp tin trong tệp tin đó. Máy chủ nêu trên cần có khả năng truy nhập ngẫu nhiên đến bất kỳ phần nào của dữ liệu được trình diễn. Việc sử dụng nhiều lần một nội dung lưu trong máy chủ bằng cách tham chiếu đến cùng dữ liệu đa phương tiện từ các bản trình diễn khác nhau thực sự là có lợi. Máy chủ cũng có thể hỗ trợ việc streaming nếu như dữ liệu đa phương tiện chỉ có thể ở đạng chỉ đọc (ví dụ như trên đĩa CD), không cho phép sao chép mà chỉ được phép thêm vào khi chuẩn bị cho streaming.

Hình vẽ sau miêu tả một bản trình diễn được sử dụng cho việc streaming trên các giao thức khác khau, trong đó chỉ yêu cầu một rãnh chỉ dẫn.



Hình 3 - Cấu trúc tệp tin dùng cho streaming

## Các nguyên tắc thiết kế

Cấu trúc của tệp tin được đề cập đến trong tài liệu này theo dạng hướng đối tượng, vì vậy, mỗi tệp tin có thể dễ dàng được phân tách thành các đối tượng cấu thành, kiểu của đối tượng cũng chỉ ra cấu trúc của đối tượng.

Dữ liệu đa phương tiện không được đóng gói theo định dạng tệp tinmiêu tả trong tài liệu này bởi vì định dạng tệp tin này chỉ chứa kích thước, kiểu và vị trí của các đơn vị dữ liệu đa phương tiện chứ không đi kèm với dữ liệu đa phương tiện. Điều đó cho phép tách riêng dữ liệu đa phương tiện và sử dụng như các dữ liệu thông thường. Siêu dữ liệu được sử dụng để tham chiếu đến dữ liệu đa phương tiện.

Cũng tương tự, thông tin cụ thể về giao thức sử dụng cho việc streaming cũng không chứa dữ liệu đa phương tiện, phần mào đầu của giao thức không gắn liền với dữ liệu đa phương tiện. Thay vào đó, dữ liệu đa phương tiện này sẽ được liên kết thông qua việc tham chiếu. Vì thế việc trình diễn dữ liệu đa phương tiện được thực hiện như dữ liệu thông thường và dùng chung cho bất cứ một giao thức nào. Như vậy cùng một tập dữ liệu đa phương tiện có thể dùng đểtrình diễn cục bộ và dùng cho nhiều giao thức khác nhau.

Thông tin về giao thức được xây dựng theo cáchmà máy chủ streaming chỉ cần biết đến giao thức và cách truyền, do thông tin về đa phương tiện có thể được trích xuất trong phần thông tin giao thức vì vậy các máy chủ không cần biếtđến kiểu đa phương tiện. Tương tự như vậy các công cụ tương tác với đa phương tiện cũng không cần quan tâm đến giao thức được sử dụng là gì.

Định dạng tệp tin được đề cập đến ở đây không yêu cầu nhất thiết một bản trình diễn phải nằm chọn trong một tệp tin. Điều này cho phép chia nhỏ cấu hình và tái sử dụng nội dung. Khi kết hợp với cơ chế không đóng khung thì hoàn toàn có thể tạo ra tệp tin chứa dữ liệu đa phương tiện không tuân theo định dạng được nêu ra trong chuẩn này (ví dụ: các tệp tin “thô “chỉ chứa dữ liệu đa phương tiện mà không chứa thông tin miêu tả).

Định dạng tệp tin này dựa trên một bộ thiết kế phổ biến và tập hợp rất nhiều cấu trúc và ứng dụng. Cùng một định dạng có thể sử dụng theo nhiều cách khác nhau, không phải chuyển đổi. Tuy nhiên, khi sử dụng trong một số trường hợp đặc biệt (ví dụ như trong việc trình diễn cục bộ), tệp tin này có thể cần được hiệu chỉnh lại nhằm tối ưu cho việc xử lý (ví dụ như trật tự thời gian của dữ liệu). Các quy tắc về cấu trúc không chuẩn không được đề cập đến trong tài liệu này, hãn hữu mới có trường hợp được sử dụng.

# Cấu trúc tệp tin đa phương tiên cơ sởtheo chuẩn ISO

## Cấu trúc bản trình diễn

### Cấu trúc tệp tin

Một bản trình diễn có thể bao gồm nhiềutệp tin. Trong đó có một tệp tin chứa siêu dữ liệu cho toàn bộ bản trình diễn và được định dạng theo chỉ tiêu kỹ thuật được đề cập đến trong tài liệu này. Nếu tệp tinchứa siêu dữ liệu cùng với toàn bộdữ liệu đa phương tiện, khi đó bản trình diễn này được gọi là bản trình diễnđộc lập. Các tệp tin còn lại, nếu được sử dụng, không nhất thiết phải được định dạng theo tiêu chuẩn này; các tệp tin này đượcdùng để chứa dữ liệu đa phương tiện, dữ liệu đa phương tiệnchưa sử dụng, hoặc các thông tin khác. Phạm vi của tiêu chuẩn này chỉ liên quan đến cấu trúc của tệp tintrình diễn. Định dạng của các tệp tindữ liệu đa phương tiệncần phải tuân thủ theo chỉ tiêu kỹ thuật nêu trong tiêu chuẩn nàykhi dữ liệu đa phương tiện trong các tệp tinđa phương tiện đóliên kếtvới siêu dữ liệu được đề cập đến ở đây.

Những tệp tin khác có thể là các tệp tin ISO, tệp tinhình ảnh, hoặc cáctệp tin có định dạng khác. Chỉ có cơ bản bản thân dữ liệu đa phương tiện, ví dụ như các hình ảnh JPEG 2000 được lưu trong các tệp tin khác, tất cả thông tin về thời gian và khung (vị trí và kích thước) cần nằm trong tệp tinđa phương tiện cơ sở theo tiêu chuẩn ISO, còn các tệp tin phụ trợ về cơ bản là không bó buộc về định dạng.

Nếu tệp tin ISO chứa các rãnh chỉ dẫn, các rãnh đa phương tiện(ví dụ như rãnh âm thanh, rãnh hình ảnh) tham chiếu đến dữ liệu đa phương tiệnđể tạo nên các chỉ dẫn sẽ cùng nằm trong tệp tinISO đó,điều này vẫn đúng ngay cả trong trường hợp dữ liệu đa phương tiệnđó không nằm trong cùng một tệp tin với các rãnh; nếu xóa tất cả các rãnh chỉ dẫn, toàn bộ phần bản trình diễn không được chỉ dẫn sẽ vẫn tồn tại. Lưu ý rằng các rãnh đa phương tiệnhoàn toàn có thể tham chiếu đến các tệp tinđa phương tiện bên ngoài.

Phụ lục A sẽ cung cấp những thông tin cơ bản hữu ích cho những độc giả lần đầu tiên tiếp cận đến lĩnh vực này.

### Cấu trúc đối tượng

Các tệp tin được đề cập đến ở đây có cấu trúc gồmnhiều đối tượng, một số đối tượng lại có thể chứa các đối tượng khác. Trình tự của các đối tượng trong tệp tinphải được thể hiện một cách cơ bản xác bởi gói siêu dữ liệu (Hộp Movie). Gói siêu dữ liệu này thường nằm ở phần đầu hoặc phần cuối của tệp tin để thuận tiện cho việc xác định vị trí. Các đối tượng tương tự có thể là Hộp Kiểu Tệp tin (File Type Box),Hộp Không gian Trống (Free Space Box), Hộp Phân mảnh Movie, Hộp Siêu dữ liệu hoặc các Hộp Dữ liệu Đa phương tiện (Media Data Box).

### Siêu dữ liệu và dữ liệu đa phương tiện

Siêu dữ liệu được đóng gói trong gói siêu dữ liệu (Hộp Movie). Dữ liệu đa phương tiện(thuộc Hộp Dữ liệu Đa phương tiện) có thể nằm trong cùng một tệp tin với siêu dữ liệu, hoặccó thể nằm trong tệp tin khác nhau. Dữ liệu đa phương tiện bao gồm dữ liệu hình ảnh hoặc dữ liệu âm thanh, các đối tượng dữ liệu đa phương tiện, hoặc các tệp tin dữ liệu đa phương tiện, và có thể là thông tin không tham chiếu khác.

### Định danh rãnh

Các định danh rãnh trong tệp tin ISO trong một tệp tin là duy nhất, không có trường hợp hai rãnh có cùng chung một định danh.

Giá trị định danh của rãnh kế tiếp được lưu trữ trong next\_*track\_id* của Hộp Mào đầu Movie và thường có giá trị lớn hơn giá trị định danh rãnh lớn nhất trong tệp tin. Điều này cho phép dễ dàng tạo mộtđịnh danh rãnh trong hầu hết các trường hợp. Tuy nhiên, nếu giá trị này toàn là các số 1 (toàn bộ 32 bit 1), thì cần tìm kiếm định danh rãnh chưa sử dụng khi thêm mới.

## Cấu trúc của siêu dữ liệu (các đối tượng)

### Hộp

Các kiểu không được đề cập đến trong tài liệu này sẽ dùng để dự phòng. Những kiểu mở rộng mang tính cá nhân sẽ được chỉ ra thông qua kiểu *‘uuid’*.

Các kiểu sau đây sẽ không được sử dụng từ nay về sau, nếu sử dụng thì chỉ trong một số trường hợp đặc biệt trong phiên bản sau của tiêu chuẩn nàynhằm tránh xung đột với nội dung của các phiên bản trước:

*clip, crgn, matt, kmat, pnot, ctab, load, imap;*

Các kiểu tham chiếu các rãnh (như trong *reference\_type* của Hộp Tham chiếu Rãnh): *tmcd, chap, sync, scpt, ssrc*.

Có một số hộp được đánh số, hộp đầu tiên sẽ được đánh số 1.

### Các trường và các kiểu dữ liệu

Trong tiêu chuẩn này, nhiều hộp có hai kiểu biến: phiên bản 0 sử dụng các trường 32 bit và phiên bản 1 sử dụng các trường 64 bit. Các trường 32 bit thường được khuyến nghị sử dụng, các trường 64 bit chỉ sử dụng khi có yêu cầu. Các giá trị như bộ đếm, độ dịch chuyển, thời gian, khoảng thời gian,...sẽ không quay về 0 khi vượt qua giá trị lớn nhất của trường đó, do vậy cần xác định độ lớn nhất có thể của mỗi trường một cách phù hợp.

Trong quá trình tạo ra nội dung, để thuận tiện thì thời gian tạo và chỉnh sửa đều được lưu trong tệp tin. Giá trị của thời gian có thể là các số 32 bit hoặc 64 bit, được tính bằng giây và bắt đầu từ nửa đêm ngày 01 tháng 01 năm 1904. Nếu sử dụng 32 bit thì có thể biểu diễn được gần đến năm 2040. Các giá trị thời gian này sử dụng kiểu thời gian toàn cầu (Universal Time Coordinated – UTC) và có thể chuyển thành thời gian cục bộ nếu cần.

Các số dấu chấm tĩnh có thể là các số âm hoặc dương, được tính bằng cách chia một số nguyên cho một số mũ 2 thích hợp. Ví dụ: số dấu phẩy tĩnh 30.2 được tạo ra bằng cách chia một số nguyên 32 bit cho 4.

Các trường được thể hiện là “khuôn mẫu “(template) trong các miêu tả hộp là tùy chọn đặc tả được sử dụng trongtài liệu này. Nếu dùng trong kỹ thuật khác, trường mẫu được sử dụng phải tuân thủ theo các định nghĩa nêu ở đây và đặc điểm kỹ thuật đó phải được chỉ rõ là tùy chọn hay bắt buộc. Cũng tương tự, các trường được đánh dấu là “đã xác định trước” (pre-defined) là trường được sử dụng trong phiên bản trước của tiêu chuẩn này. Đối với cả hai trường, nếu một trường không được sử dụng trong một chỉ tiêu nào đó thì nên thiết lập giá trị mặc định. Nếu trường không được sử dụng, khi sao chép các hộp, trường đó cũng vẫn được sao chép nhưng khi đọc sẽ bị bỏ qua.

Các giá trị trong ma trận nằm trong phần mào đầu chỉ định việc chuyển đổi định dạng của hình ảnh cho việc trình diễn. Không phải tất cả các thông số kỹ thuật đều sử dụng ma trận, nếu không được sử dụng, các thông số sẽ được thiết lập để nhận dạng ma trận. Nếu sử dụng ma trận, các điểm (p,q) được chuyển đổi thành (p’,q’) bằng cách:

Trong đó:

m = ap + cq + x;

n = bp + dq + y;

z = up + vq + w;

p' = m/z;

q' = n/z

Tọa độ (p,q) thể hiện khung đã giải nén, (p’,q’) là kết quả sau khi kết xuất. Vì vậy, một ma trận (2,0,0, 0,2,0, 0,01) sẽ tăng gấp đôi kích thước của một ảnh. Các tọa độ được biến đổi bởi ma trận không phải là các giá trị thực và chỉ đại diện cho các vị trí mẫu thực tế. Do đó (x,y) được coi là một vector chuyển đổi của hình ảnh.

Tọa độ gốc nằm ở góc trên bên trái, X sẽ tăng về phía bên phải, Y tăng theo hướng từ trên xuống. (p,q) và (p’,q’) là vị trí điểm ảnh tuyệt đối tính từ góc trên bên trái của ảnh gốc (sau khi điều chỉnh kích thước bởi chiều cao và độ rộng của mào đầu rãnh) và chuyển đổi bề mặt tương ứng.

Mỗi một rãnh được tạo ra bằng cách sử dụng ma trận của nó như quy định trong một ảnh tổng thể, sau đó được chuyển đổi và tạo dựng theo ma trận ở mức movie trong Hộp Mào đầu Movie. Đó là ứng dụng phụ thuộc vào việc liệu hình ảnh kết quả có bị cắt để kiểu bỏ các điểm ảnh không được trình diễn, hay để một vùng hình vuông thẳng đứng trong cửa sổ. Vì vậy, ví dụ nếu một rãnh hình ảnh được hiển thị và chuyển đổi sang (20,30) cùng với một ma trận thống nhất nằm trong Hộp Mào đầu Movie thì một ứng dụng có thể chọn cách không hiển thị vùng khoảng trống L giữa ảnh đó và ảnh gốc.

Tất cả các con số trong một ma trân đều được lưu theo các dạng dấu chấm tĩnh 16.16, trừ các giá trị u,v và w được lưu theo dạng dấu chấm tĩnh 2.30.

Các số được lưu trong ma trận theo thứ tự (a,b,u, c,d,v, x,y,w).

### Thứ tự của hộp

Bảng miêu tả dưới đây sẽ thể hiệncách thức đóng gói một cách tổng quát.Các hộp ở mức cao sẽ nằm ở các cột bên trái của bảng, việc thụt vào đầu dòng thể hiện các mục có thể có trong một đối tượng. Ví dụ: Hộp Mào đầu Rãnh (Track Header Box – *tkhd*) nằm trong Hộp Rãnh (Track Box – *trak*), Hộp Rãnh này lại nằm trongHộp Movie (MovieBox– *moov*). Không phải tất cả các hộp đều cần được sử dụng trong các tệp tin, các hộp bắt buộc phải sử dụng được đánh dấu hoa thị (\*).

Các đối tượng dữ liệu người dùng sẽ chỉ nằm trong cácHộp Movie hoặc Hộp Rãnh, các đối tượng sử dụng kiểu mở rộng có thể nằm trong các hộp khác nhau, không chỉ ở các hộp mức cao (top-level).

Để tăng cường khả năng tương tác và hiệu quả sử dụng của các tệp tin, các quy định và hướng dẫn sau đây sẽ được tuân thủ khi đề cập đến thứ tự của hộp:

1. Hộp kiểu tệp tin ‘*ftyp*’ sẽ đứng trước tất cả các hộp có độ dài thay đổi (ví dụ:Hộp Movie, Hộp Không gian Trống, Hộp Dữ liệu Đa phương tiện). Chỉ có những hộp có kích thước cố định như hộp chữ ký tệp tin sẽ đứng trước Hộp Kiểu Tệp tin khi cần.
2. Tất cả các hộp mào đầu được khuyến cáo đặt ở phần đầu tiên của các đối tượng chứa hộp mào đầu đó. Các hộp mào đầu bao gồm: Hộp Mào đầu Movie, Hộp Mào đầu Rãnh, Hộp Mào đầu Đa phương tiện và các hộp mào đầu nằm trong Hộp Thông tin Đa phương tiện (ví dụ: Hộp Mào đầu Đa phương tiện hình ảnh).
3. Bất kỳ Hộp Phân mảnh Movienào cũng sẽ được sắp xếp theo thứ tự (xem 8.8.5).
4. Các hộp trong Hộp Bảng Mẫuđược khuyến nghị xắp xếp theo thứ tự như sau: Hộp Miêu tả Mẫu, Hộp Ánh xạ Thời gian và Mẫu, Hộp Ánh xạ Đoạn và Mẫu, Hộp Kích thước Mẫu, Hộp Độ dịch chuyển Đoạn.
5. Hộp Tham chiếu Rãnh và Hộp Danh sách Chỉnh sửa (nếu có) được khuyến nghị đứng trước Hộp Đa phương tiện, Hộp Tham chiếu Xử lý nên đứng trước Hộp Thông tin Đa phương tiện và Hộp Thông tin Dữ liệu nên đứng trước Hộp Bảng Mẫu.
6. Các Hộp Dữ liệu Người dùng nên được đặt ở phần cuối của hộp chứa chúng, cơ bản là Hộp Movie và Hộp Rãnh.
7. Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie (nếu có) nên nằm ở phần cuối cùng của tệp tin.
8. Hộp Thông tin Tải xuống Liên tục nên đặt ở càng gần phần đầu tệp tin càng tốt để tăng tối đa việc sử dụng.

Bảng 1 – Các kiểu hộp, cấu trúc và mục tham chiếu

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Các kiểu hộp** | | | | | | **Bắt buộc (\*)** | **Mục tham chiếu** | **Diễn giải** |
| *ftyp* |  |  |  |  |  | \* | 4.3 | kiểu tệp tin và tính năng |
| *pdin* |  |  |  |  |  |  | 8.1.3 | thông tin tải xuống liên tục |
| *moov* |  |  |  |  |  | \* | 8.2.1 | vật chứa của tất cả siêu dữ liệu |
|  | *mvhd* |  |  |  |  | \* | 8.2.2 | mào đầu movie, các khai báo tổng quan |
|  | *trak* |  |  |  |  | \* | 8.3.1 | vật chứa của một rãnh hoặc một luồng |
|  |  | *tkhd* |  |  |  | \* | 8.3.2 | mào đầu rãnh, thông tin tổng quát về rãnh |
|  |  | *tref* |  |  |  |  | 8.3.3 | vật chứa tham chiếu rãnh |
|  |  | *trgr* |  |  |  |  | 8.3.4 | chỉ thị việc nhóm các rãnh |
|  |  | *edts* |  |  |  |  | 8.6.4 | vật chứa danh sách biên tập |
|  |  |  | *elst* |  |  |  | 8.6.6 | danh sách biên tập |
|  |  | *mdia* |  |  |  | \* | 8.4 | vật chứa thông tin đa phương tiện trong một rãnh |
|  |  |  | *mdhd* |  |  | \* | 8.4.2 | mào đầu đa phương tiện, thông tin chung về đa phương tiện |
|  |  |  | *hdlr* |  |  | \* | 8.4.3 | bộ xử lý, khai báo kiểu xử lý đa phương tiện |
|  |  |  | *minf* |  |  | \* | 8.4.4 | vật chứa thông tin đa phương tiện |
|  |  |  |  | *vmhd* |  |  | 8.4.5.2 | mào đầu đa phương tiện hình ảnh, thông tin chung cho rãnh hình ảnh |
|  |  |  |  | *smhd* |  |  | 8.4.5.3 | mào đầu đa phương tiện âm thanh, thông tin chung cho rãnh âm thanh |
|  |  |  |  | *sthd* |  |  | 8.4.5.6 | mào đầu đa phương tiện phụ đề, thông tin chung cho rãnh âm thanh |
|  |  |  |  | *hmhd* |  |  | 8.4.5.4 | mào đầu đa phương tiện chỉ dẫn, thông tin chung cho rãnh chỉ dẫn |
|  |  |  |  | *nmhd* |  |  | 8.4.5.5 | mào đầu rỗng, chỉ áp dụng cho một số rãnh |
|  |  |  |  | *dinf* |  | \* | 8.5 | hộp thông tin dữ liệu, vật chứa |
|  |  |  |  |  | *dref* | \* | 8.7.2 | hộp tham chiếu dữ liệu, khai báo các nguồn của dữ liệu đa phương tiện trong rãnh |
|  |  |  |  | *stbl* |  | \* | 8.5 | hộp bảng mẫu, chứa ánh xạ giữa thời gian và không gian |
|  |  |  |  |  | *stsd* | \* | 8.5.2 | các miêu tả mẫu (các kiểu codec, thiết lập,...) |
|  |  |  |  |  | *stts* | \* | 8.6.1.2 | (giải mã) ánh xạ thời gian vào mẫu |
|  |  |  |  |  | *ctts* |  | 8.6.1.3 | (tổng hợp) ánh xạ thời gian vào mẫu |
|  |  |  |  |  | *cslg* |  | 8.6.1.4 | ánh xạ thời gian tổng hợp và thời gian giải mã |
|  |  |  |  |  | *stsc* | \* | 8.7.4 | ánh xạ giữa mẫu và đoạn, thông tin một phần của độ dịch chuyển dữ liệu |
|  |  |  |  |  | *stsz* |  | 8.7.3.2 | các kích cỡ của mẫu (đóng gói) |
|  |  |  |  |  | *stz2* |  | 8.7.3.3 | các kích cỡ của mẫu thu gọn (đóng gói) |
|  |  |  |  |  | *stco* | \* | 8.7.5 | độ dịch chuyển của đoạn, thông tin một phần của độ dịch chuyển dữ liệu |
|  |  |  |  |  | *co64* |  | 8.7.5 | độ dịch chuyển của đoạn 64 bit |
|  |  |  |  |  | *stss* |  | 8.6.2 | bảng mẫu đồng bộ |
|  |  |  |  |  | *stsh* |  | 8.6.3 | bảng mẫu bóng đồng bộ |
|  |  |  |  |  | *padb* |  | 8.7.6 | mẫu chèn bit |
|  |  |  |  |  | *stdp* |  | 8.7.6 | ưu tiên suy giảm mẫu |
|  |  |  |  |  | *sdtp* |  | 8.6.4 | các mẫu phụ thuộc và mẫu dùng một lần |
|  |  |  |  |  | *sbgp* |  | 8.9.2 | ánh xạ giữa mẫu và nhóm |
|  |  |  |  |  | *sgpd* |  | 8.9.3 | miêu tả nhóm mẫu |
|  |  |  |  |  | *subs* |  | 8.7.7 | thông tin mẫu phụ |
|  |  |  |  |  | *saiz* |  | 8.7.8 | các kích thước thông tin phụ của mẫu |
|  |  |  |  |  | *saio* |  | 8.7.9 | các độ dịch chuyển thông tin phụ của mẫu |
|  |  | *udta* |  |  |  |  | 8.10.1 | dữ liệu người dùng |
|  | *mvex* |  |  |  |  |  | 8.8.1 | hộp mở rộng movie |
|  |  | *mehd* |  |  |  |  | 8.8.2 | hộp mào đầu mở rộng mẫu |
|  |  | *trex* |  |  |  | \* | 8.8.3 | giá trị mặc định mở rộng rãnh |
|  |  | *leva* |  |  |  |  | 8.8.13 | phân cấp |
| *moof* |  |  |  |  |  |  | 8.8.4 | phân mảnh movie |
|  | *mfhd* |  |  |  |  | \* | 8.8.5 | mào đầu phân mảnh movie |
|  | *traf* |  |  |  |  |  | 8.8.6 | phân mảnh rãnh |
|  |  | *tfhd* |  |  |  | \* | 8.8.7 | mào đầu phân mảnh rãnh |
|  |  | *trun* |  |  |  |  | 8.8.8 | hoạt động phân mảnh rãnh |
|  |  | *sbgp* |  |  |  |  | 8.9.2 | ánh xạ giữa mẫu và nhóm |
|  |  | *sgpd* |  |  |  |  | 8.9.3 | miêu tả nhóm mẫu |
|  |  | *subs* |  |  |  |  | 8.7.7 | thông tin mẫu phụ |
|  |  | *saiz* |  |  |  |  | 8.7.8 | các kích thước thông tin phụ của mẫu |
|  |  | *saio* |  |  |  |  | 8.7.9 | các độ dịch chuyển thông tin phụ của mẫu |
|  |  | *tfdt* |  |  |  |  | 8.8.12 | thời gian giải mã phân mảnh rãnh |
| *mfra* |  |  |  |  |  |  | 8.8.9 | truy nhập ngẫu nhiên phân mảnh movie |
|  | *tfra* |  |  |  |  |  | 8.8.10 | truy nhập ngẫu nhiên phân mảnh rãnh |
|  | *mfro* |  |  |  |  |  | 8.8.11 | độ dịch chuyển truynhập ngẫu nhiên phân mảnh movie |
| *mdat* |  |  |  |  |  |  | 8.2.2 | vật chứa dữ liệu đa phương tiện |
| *free* |  |  |  |  |  |  | 8.1.2 | không gian trống |
| *skip* |  |  |  |  |  |  | 8.1.2 | không gian trống |
|  | *udta* |  |  |  |  |  | 8.10.1 | dữ liệu người dùng |
|  |  | *cprt* |  |  |  |  | 8.10.2 | bản quyền,... |
|  |  | *teel* |  |  |  |  | 8.10.3 | hộp chọn lựa rãnh |
|  |  | *strk* |  |  |  |  | 8.14.3 | hộp rãnh con |
|  |  |  | *stri* |  |  |  | 8.14.4 | hộp thông tin rãnh con |
|  |  |  | *strd* |  |  |  | 8.14.5 | hộp định nghĩa rãnh con |
| *meta* |  |  |  |  |  |  | 8.11.1 | siêu dữ liệu |
|  | *hdlr* |  |  |  |  | \* | 8.4.3 | bộ xử lý, khai báo kiểu siêu dữ liệu |
|  | *dinf* |  |  |  |  |  | 8.5 | hộp thông tin dữ liệu, vật chứa |
|  |  | *dref* |  |  |  |  | 8.7.2 | hộp tham chiếu dữ liệu, khai báo nguồn của các mục siêu dữ liệu |
|  | *iloc* |  |  |  |  |  | 8.11.3 | vị trí hạng mục |
|  | *ipro* |  |  |  |  |  | 8.11.5 | bảo vệ hạng mục |
|  |  | *sinf* |  |  |  |  | 8.12.1 | hộp thông tin lưu đồ bảo vệ |
|  |  |  | *frma* |  |  |  | 8.12.2 | hộp định dạng gốc |
|  |  |  | *schm* |  |  |  | 8.12.5 | hộp kiểu lưu đồ |
|  |  |  | *schi* |  |  |  | 8.12.6 | hộp thông tin lưu đồ |
|  | *iinf* |  |  |  |  |  | 8.11.6 | thông tin hạng mục |
|  | *xml* |  |  |  |  |  | 8.11.2 | vật chất XML |
|  | *bxml* |  |  |  |  |  | 8.11.2 | vật chứa XML nhị phân |
|  | *pitm* |  |  |  |  |  | 8.11.4 | tham chiếu hạng mục cơ bản |
|  | *fiin* |  |  |  |  |  | 8.13.2 | thông tin hạng mục phân phối tệp tin |
|  |  | *paen* |  |  |  |  | 8.13.2 | đầu vào mục phân vùng |
|  |  |  | *fire* |  |  |  | 8.13.7 | dự trữ tệp tin |
|  |  |  | *fpar* |  |  |  | 8.13.3 | phân vùng tệp tin |
|  |  |  | *fecr* |  |  |  | 8.13.4 | dự trữ FEC |
|  |  | *segr* |  |  |  |  | 8.13.5 | nhóm phiên phân phối tệp tin |
|  |  | *gitn* |  |  |  |  | 8.13.6 | ánh xạ giữa định danh nhóm và tên |
|  | *idat* |  |  |  |  |  | 8.11.11 | dữ liệu hạng mục |
|  | *iref* |  |  |  |  |  | 8.11.12 | tham chiếu hạng mục |
| *meco* |  |  |  |  |  |  | 8.11.7 | vật chứa siêu dữ liệu cộng thêm |
|  | *mere* |  |  |  |  |  | 8.11.8 | liên hệ của hộp siêu dữ liệu |
| *styp* |  |  |  |  |  |  | 8.16.2 | kiểu phân đoạn |
| *sidx* |  |  |  |  |  |  | 8.16.3 | chỉ số phân đoạn |
| *ssix* |  |  |  |  |  |  | 8.16.4 | chỉ số phân đoạn con |
| *prft* |  |  |  |  |  |  | 8.16.5 | thời gian tham chiếu sản xuất |

### Bộ chỉ thị URI

Khi sử dụng URInhư một bộ chỉ thị (ví dụ như trong đầu vào mục mẫu hoặc siêu dữ liệu không định thời) thì URI phải là giá trị tuyệt đối, định dạng và nghĩa của dữ liệu phải được xác định trong URI. Việc nhận dạng này có thể được phân cấp theo cách:một phần chuỗi khởi tạo của URI sẽ thể hiện đặc điểm tổng quát hoặc họ dữ liệu (ví dụ: urn:oid xác định siêu dữ liệu được dán nhãn bởi một định danh đối tượng theo chuẩn ISO).

URI này nên thuộc dạng de-referencable. URI có thể là chuỗi được thiết bị đọc so sánh với tập các kiểu URI đã biết. URI cung cấp các kiểu định danh lớn, không trùng lặp và không phải đăng ký.

Nếu URI chứa tên miền (ví dụ như là một URL) thì URI đó nên chứa ngày tháng theo dạng mmyyyy. Thời gian này phải gần với thời gian định nghĩa của phần mở rộng và phải là một tên miền được xác định bởi chủ nhân có quyền hợp pháp tại thời điểm đó (Điều này cần thiết để tránh các vấn đề liên quan đến việc thay đổi chủ sở hữu tên miền).

## Định danh nhãn

Các định nghĩa liên quan đến nhãn áp dụng cho định dạng tệp tin được miêu tả trong Phụ lục E.

# Hỗ trợ streaming

## Xử lý các giao thức streaming

Định dạng tệp tin được đề cập đến ở đây hỗ trợ việc streaming dữ liệu đa phương tiện qua mạng cũng như trình diễn cục bộ. Quá trình gửi các đơn vị dữ liệu dựa trên giao thức phụ thuộc vào thời gian, giống như việc trình diễn dữ liệu dựa vào thời gian, vì vậy định dạng được đề cập đến trong tài liệu này được coi là định dạng dựa trên thời gian. Một tệp tin nếu hỗ trợ streaming sẽ chứa thông tin về các đơn vị dữ liệu để thực hiện việc streaming. Thông tin này bao gồm cả các rãnh chỉ dẫn. Các rãnh chỉ dẫncũng cóthể được dùng để ghi một luồng, được gọi lại rãnh chỉ dẫn tiếp nhận (Reception Hint Track), kiểu rãnh chỉ dẫn này khác với các rãnh chỉ dẫnthông thường (rãnh chỉ dẫn máy chủ hoặc rãnh chỉ dẫn truyền tải).

Rãnh chỉ dẫn máy chủ hoặc rãnh chỉ dẫn truyền tải chứa thông tin chỉ dẫn để giúp máy chủ streaming thực hiện việc truyền các gói tin. Những chỉ dẫn này có thể bao gồm dữ liệu trực tiếp mà máy chủ gửi đi (ví dụ thông tin mào đầu) hoặc các phần tham chiếu của dữ liệu đa phương tiện. Những chỉ dẫn này được mã hóa trong tệp tin giống như việc mã hóa thông tin sửa đổi hoặc trình diễn trong tệp tin dùng cho việc trình diễn cục bộ. Thay vì các thông tin sử đổi hoặc thông tin trình diễn, thông tin chỉ dẫn chỉ cho máy chủ biết cách thức đóng gói dữ liệu đa phương tiện phù hợp với việc streaming ứng với một mạng truyền tải cụ thể nào đó.

Dữ liệu đa phương tiện trong tệp tin có các chỉ dẫn có thể được sử dụng trong việc trình diễn cục bộ hoặc được streaming thông qua nhiều giao thức khác nhau. Các rãnh chỉ dẫnriêng biệt cho từng giao thức có thể nằm cùng trong một tệp tin và việc trình diễn thông qua các giao thức này sẽ sử dụng chung một nguồn dữ liệu đa phương tiện. Ngoài ra, với dạng tệp tinđa phương tiệntrình diễn dạng cục bộ thì hoàn toàn có thể thêm các rãnh chỉ dẫn phù hợp với từng giao thức cụ thể, khi đó dữ liệu đa phương tiện không cần phải xử lý hay định dạng lại.

Việc streaming hay ghi theo cách tiếp cận trên mang lại hiệu quả về không gian tốt hơn so với cách tiếp cận mà thông tin đa phương tiện đi kèm với dữ liệu đa phương tiện. Theo cách tiếp cận như vậy, việc trình diễn cục bộ sẽ yêu cầu cả việc đóng gói lại đa phương tiện và tạo ra hai bản sao của đa phương tiện đó, một cho việc trình diễn cục bộ và một cho streaming. Tương tự, nếu streamingđa phương tiệnđó theo các giao thức khác nhau thì cần phải sao chép riêng từng dữ liệu đa phương tiện cho mỗi luồng truyền tải. Điều này rất tốn không gian lưu trữ và chỉ sử dụng khi đã chuyển đổi dữ liệu đa phương tiện cần truyền đó rồi (ví dụ như đã sử dụng kỹ thuật mã hóa sửa lỗi hoặc đã mã hóa).

Các rãnh chỉ dẫn tiếp nhận có thể được sử dụng khi ghi một hoặc nhiều luồng dữ liệu. Các rãnh này chỉ rõ thứ tự, thời gian nhận và nội dung của các gói.

LƯU Ý: Thiết bị phát có thể tái tạo luồng dữ liệu nhận được dựa trên các rãnh chỉ dẫn tiếp nhận và xử lý luồng dữ liệu tái tạo giống như vừa nhận được.

## Rãnh chỉ dẫn giao thức

Việc hỗ trợ streaming dựa trên 3 tham số sau:

* Dữ liệu đa phương tiện: Dữ liệu đa phương tiện là một tập các rãnh tiêu chuẩn không phụ thuộc vào mạng, dữ liệu này có thể được dùng để trình diễn, được chỉnh sửa,...
* Rãnh chỉ dẫnthông thường: Rãnh chỉ dẫn có cấu trúc cơ bản và cách thức khai báo chung, độc lập với giao thức nhưng vẫn chứa tham số chỉ rõ giao thức được sử dụng.
* Rãnh chỉ dẫn giành riêng: Rãnh chỉ dẫn này được thiết kế cụ thể cho từng giao thức, nhưng vẫn dùng chung một kiến trúc cơ bản. Ví dụ, có thể có rãnh dành riêng cho giao thức RTP (truyền trên Internet), cho truyền tải MPEG-2 (quảng bá), hoặc cho một giao thức mới cũng như cho giao thức của một nhà cung cấp nào đó.

Các luồng dữ liệu được máy chủ truyền đi dựa trên rãnh chỉ dẫn hoặc được tái tạo lại dựa trên rãnh chỉ dẫn tiếp nhận không cần vết của thông tin tệp tin cụ thể. Cách thiết kế nêu trên không yêu cầu cấu trúc tệp tin hoặc dạng khai báodùng cho cả dữ liệu trên đường truyền hoặc cho các bộ phận mã hóa. Ví dụ, một tệp tin sử dụng hình ảnh theo chuẩn ITU-T H.261 và âm thanh DVI, giao thức truyền RTP, khi đó luồng dữ liệu được truyền đi hoàn toàn phù hợp với tiêu của IETF về việc đóng gói dữ liệu vào RTP.

## Định dạng rãnh chỉ dẫn

Các rãnh chỉ dẫn được sử dụng để miêu tả dữ liệu luồng cơ bản trong tệp tin. Mỗi giao thức hoặc mỗi họ giao thức sẽ có một định dạng rãnh chỉ dẫn riêng. Định dạng rãnh chỉ dẫn cho máy chủ và định dạng rãnh chỉ dẫn tiếp nhận của cùng giao thức được phân biệt bởi 4 ký tự trong đầu vào mục miêu tả mẫu. Nói cách khác, đoạn mã bốn ký tự này được sử dụng cho rãnh chỉ dẫn máy chủ và rãnh chỉ dẫn tiếp nhập của cùng giao thức. Cú pháp của định dạnh rãnh chỉ dẫn máy chủ và định dạng của rãnh chỉ dẫn tiếp nhận của cùng giao thức nên được xây dựng giống nhau hoặc tương thích với nhau để rãnh chỉ dẫn tiếp nhận có thể sử dụngcho việc truyền lại luồng dữ liệu, nhằm đảm bảo rằng khả năng suy giảm chất lượng của các luồng dữ liệu nhận được trong phạm vi cho phép. Đối với hầu hết các giao thức thì chỉ cần duy nhất một định dạng miêu tả mẫu cho mỗi rãnh.

Trình tự tìm rãnh chỉ dẫn của máy chủ như sau: ban đầu máy chủ sẽ thực hiện việc tìm tất cả các rãnh chỉ dẫn, sau đó sẽ tìm trong đó rãnh chỉ dẫn sử dụng giao thức mà máy chủ đó cần (nằm trong định dạng miêu tả mẫu). Nếu có nhiều sự chọn lựa khác nhau, máy chủ sẽ chọn lựa dựa trên giao thức được ưu tiên hoặc bằng cách so sánh các tính năng trong mào đầu của rãnh chỉ dẫn, hoặc dựa trên thông tin về giao thức cụ thể khác trong các miêu tả mẫu. Đặc biệt trong trường hợp không có rãnh chỉ dẫn máy chủ, máy chủ có thể sẽ sử dụng rãnh chỉ dẫn tiếp nhận cho giao thức cần truyền. Tuy nhiên, máy chủ cần tính toán khả năng suy giảm chất lượng của luồng dữ liệu nhận thông qua việc sử dụng rãnh chỉ dẫn tiếp nhận một cách thích hợp.

Rãnh chỉ dẫn tạo ra các luồng bằng cách lấy dữ liệu từ các rãnh thông qua các tham chiếu. Các rãnh này có thể là các rãnh chỉ dẫn hoặc rãnh luồng cơ bản. Định dạng cơ bản xác của các tham chiếu này được xác định trong định dạng mẫu của giao thức đó, nhưng nhìn chung thì các tham chiếu này gồm 4 thông tin sau: chỉ số tham chiếu rãnh, số mẫu, độ dịch chuyển và độ dài. Đối với một số giao thức có thể sẽ không có một vài tham số vừa nêu.Các tham chiếu luôn chỉ đến đúng dữ liệu thực. Nếu một rãnh chỉ dẫn (a1) được xây dựng trên một rãnh chỉ dẫn khác (a) thì rãnh chỉ dẫn đó (a1) phải tham chiếu trực tiếp đến rãnh đa phương tiện được trỏ đến bởi rãnh chỉ dẫn trước đó (a).

Tất cả các rãnh chỉ dẫn sẽ sử dụng chung một tập các khai báo và chung cấu trúc như sau:

* Tất cả các rãnh chỉ dẫn được liên kết với các rãnh luồng cơ bản thông qua các tham chiếu rãnh có kiểu “chỉ dẫn “(“hint “).
* Tất cả các rãnh chỉ dẫn đều sử dụng một kiểu xử lý “chỉ dẫn” trong Hộp Tham chiếu Xử lý.
* Tất cả các rãnh chỉ dẫn đều sử dụng một Hộp Mào đầu Đa phương tiệnChỉ dẫn.
* Tất cả các rãnh chỉ dẫn đều sử dụng một đầu vào mục mẫu chỉ dẫn trong miêu tả mẫu, với tên và định dạng duy nhất ứng với mỗi giao thức được sử dụng.

Các rãnh chỉ dẫn máy chủ thường được đánh dấu không sử dụng chotrình diễn cục bộ, với giá trị của các cờ *track\_in\_movie* và *track\_in\_preview* bằng 0.

Các rãnh chỉ dẫn có thể được tạo ra bởi công cụ thiết kế (authoring tool), hoặc có thể được gắn thêm vào một bản trình diễn đã có sẵn bằng công cụ chỉ dẫn (hint tool). Các công cụ được dùng như là “cầu nối “liên kết giữa đa phương tiện và giao thức do công cụ “hiểu “được cả 2 đối tượng này. Điều đó cho phép công cụ thiết kế hiểu được định dạng đa phương tiện nhưng không hiểu các giao thức, máy chủ thì hiểu được giao thức (và các rãnh chỉ dẫn) nhưng lại chẳng hiểu được dữ liệu đa phương tiện.

Các rãnh chỉ dẫn không sử dụng thời gian tổng hợp một cách riêng biệt do bảng ‘*ctts*’ không có trong rãnh chỉ dẫn. Quá trình tính toán thời gian truyền bằng cơ bản thời gian giải mã.

LƯU Ý 1: Máy chủ sử dụng rãnh chỉ dẫn tiếp nhận như là các chỉ dẫn cho việc gửi luồng dữ liệu nên có khả năng xử lý sự suy giảm chất lượng của luồng dữ liệu, ví dụ như độ trễ, jitter, mất gói và đảm bảo rằng những yêu cầu đối với giao thức và các định dạng chứa dữ liệu phù hợp với tiêu chuẩn, không phụ thuộc vào sự suy giảm chất lượng của luồng nhận được.

LƯU Ý 2:Việc chuyển đổi từ các luồng sang các rãnh đa phương tiện cho phép các hệ thống trình diễnđang tồn tại tương thích với phiên bản cũ của định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO để xử lý các tệp tin đã được ghi nếu các định dạng đa phương tiện đó được hệ thống hỗ trợ.Tuy nhiên, phần lớn các tiêu chuẩn mã hóa đa phương tiện chỉ đề cập đến việc giải mã các luồng không lỗi, vì vậy cần phải đảm bảo nội dung chứa trong các rãnh đa phương tiện có thể được giải mã đúng. Các hệ thống trình diễncó thể sử dụng các rãnh chỉ dẫn tiếp nhận để xử lý sự suy giảm chất lượng gây nên bởi quá trình truyền thông tin, ví dụ: nội dung có thể không được giải mã đúng nằm trong các rãnh chỉ dẫn tiếp nhận. Việc lưu dữ liệu tại cả rãnh đa phương tiện và rãnh chỉ dẫn tiếp nhận có thể kiểu bỏ bằng cách chỉ lưu tại một rãnh và rãnh còn lại sẽ tham chiếu sang, cụ thể dữ liệu trong rãnh đa phương tiện sẽ tham chiếu sang rãnh chỉ dẫn tiếp nhận.

# Cấu trúc hộp

## Cấu trúc tệp tin và các hộp thông thường

### Hộp Dữ liệu Đa phương tiện (Media Data Box)

#### Định nghĩa

Loại hộp: ‘*mdat*’.

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp này chứa dữ liệu đa phương tiện. Trong các rãnh hình ảnh, hộp này chứa các khung hình ảnh. Một bản trình diễncó thể không có hoặc có nhiều Hộp Dữ liệu Đa phương tiện. Dữ liệu đa phương tiện thực tế theo sau trường kiểu, cấu trúc của trường này được miêu tả bởi siêu dữ liệu (xem bảng mẫu, 8.5 và hộp định vị hạng mục, 8.11.3).

Trong các bản trình diễnlớn, nhiều khi cần có nhiều dữ liệu trong hộp này hơn so với quy định kích thước 32 bit. Trong hợp này, các trường có kích thước lớn khác nhau (như đề cập trong 4.2) được sử dụng.

Tệp tin có thể chứa nhiều Hộp Dữ liệu Đa phương tiện. Tệp tin không chứa hộp này trong trường hợp tất cả dữ liệu đa phương tiện được nằm trong các tệp tin khác. Siêu dữ liệu tham chiếu đến dữ liệu đa phương tiện thông qua độ dịch chuyển tuyệt đối của nó bên trong tệp tin (8.7.5, Hộp Độ dịch chuyển Đoạn); nhờ đó có thể dễ dàng bỏ qua các mào đầu Hộp Dữ liệu Đa phương tiện và không gian trống, đồng thời các tệp tin không có cấu trúc hộp cũng có thể được tham chiếu và sử dụng.

#### Cú pháp

*aligned(8) class MediaDataBox extends Box(‘mdat’) {*

*bit(8) data[];*

*}*

#### Ngữ nghĩa

*data* là dữ liệu đa phương tiện.

### Hộp Không gian Trống (Free Space Box)

#### Định nghĩa

Loại hộp:‘*free*’,‘*skip*’.

Thuộc đối tượng: tệp tinhoặc hộp khác.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Thông tin chứa trong Hộp Không gian Trống thường không phù hợp và có thể bị kiểu bỏ mà không bị ảnh hưởng đến bản trình diễn. (Cần thận trọng khi xóa đối tượng này, vì có thể làm sai lệch độ dịch chuyển trong bảng mẫu, trừ khi đối tượng này ở phía sau tất cả dữ liệu đa phương tiện).

#### Cú pháp

*aligned(8) class FreeSpaceBox extends Box(free\_type) {*

*unsigned int(8) data[];*

*}*

#### Ngữ nghĩa

*free\_type*có thểlà ‘*free*’hoặc‘*skip*’

### Hộp Thông tin Tải xuống Liên tục (Progressive Download Information Box)

#### Định nghĩa

Loại hộp: ‘*pdin*’.

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Thông tin Tải xuống Liên tục hỗ trợviệc tải xuống liên tục một tệp tinISO. Hộp này chứa các cặp số (đến cuối hộp) gồm tốc độ tải tệp tinhiệu dụng tính theo đơn vị byte/s và độ trễ phát lại ban đầu tính bằng milli giây.

Bên nhận có thể ước tính tốc độ tải xuống hiện tại và từ đó ước tính độ trễ ban đầu phù hợp bằng cách nội suy tuyến tính giữa các cặp hoặc ngoại suy từ tham sốđầu hoặc cuối.

Vị trí của Hộp Thông tin Tải xuống Liên tục được khuyến nghị đặt ngay ở phần đầu của tệp tin để tăng hiệu quả sử dụng.

#### Cú pháp

*aligned(8) class ProgressiveDownloadinfoBox extends FullBox(‘pdin’, version = 0, 0) {*

*for (i=0; ; i++) { // to end of box*

*unsigned int(32)rate;*

*unsigned int(32)initial\_delay;*

*}*

*}*

#### Ngữ nghĩa

*rate* là tốc độ tải xuống được biểu thị theo byte/s.

*initial\_delay*là độ trễ được đề xuất để sử dụng khi phát một tệp tin, như vậy nếu việc tải xuống liên tục với tốc độ cho trước, tất cả dữ liệu trong tệp tin sẽ đến đúng thời điểm và việc phát lại không bị dừng.

## Cấu trúc của movie

### Hộp Movie (Hộp Movie)

#### Định nghĩa

Loại hộp: ‘*moov*’.

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Siêu dữ liệu của một bản trình diễnđược lưu trong mộtHộp Movieđơn nhất. Thông thường hộp này hay ở vị trí gần đầu hoặc cuối của tệp tin, mặc dù điều đó không bắt buộc.

#### Cú pháp

*aligned(8) class MovieBox extends Box(‘moov’){*

*}*

### Hộp Mào đầu Movie (Movie Header Box)

#### Định nghĩa

Loại hộp: ‘*mvhd*’.

Thuộc đối tượng: Hộp Movie (‘*moov*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này chứa thông tintổng quát, độc lập với đa phương tiện và thích hợp với toàn bộ bản trình diễn.

#### Cú pháp

*aligned(8) class MovieHeaderBox extends FullBox(‘mvhd’, version, 0) {*

*if (version==1) {*

*unsigned int(64)creation\_time;*

*unsigned int(64)modification\_time;*

*unsigned int(32)timescale;*

*unsigned int(64)duration;*

*} else { // version==0*

*unsigned int(32)creation\_time;*

*unsigned int(32)modification\_time;*

*unsigned int(32)timescale;*

*unsigned int(32)duration;*

*}*

*template int(32)rate = 0x00010000; // typically 1.0*

*template int(16)volume = 0x0100; // typically, full volume*

*const bit(16)reserved = 0;*

*const unsigned int(32)[2]reserved = 0;*

*template int(32)[9]matrix = { 0x00010000,0,0,0,0x00010000,0,0,0,0x40000000 };// Unity matrix*

*bit(32)[6]pre\_defined = 0;*

*unsigned int(32)next\_track\_ID;*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyênchỉ phiên bản của hộp (trong tiêu chuẩn này là 0 hoặc 1).

*creation\_time*có kiểu số nguyên chỉ thời gian tạo ra bản trình diễn (được tính theo giây, bắt đầu từ lúc nửa đêm ngày 1 tháng 1 năm 1904 theo thời gian quốc tế UTC).

*modification\_time*có kiểu số nguyênchỉ thời gian sử đổi bản trình diễngần nhất(được tính theo giây, bắt đầu từ lúc nửa đêm ngày 1 tháng 1 năm 1904 theo thời gian quốc tếUTC).

*timescale*có kiểu số nguyên chỉ thang thời gian của toàn bộ bản trình diễn; đây là số đơn vị thời gian trôi qua trong một giây. Ví dụ, một hệ thống phối hợp thời gian đo thời gian một phần sáu mươi của giây có thang thời gian 60.

*duration*có kiểu số nguyênchỉ độ dài của bản trình diễn(trong thang thời gian được chỉ định). Thuộc tính này được trích xuất từ các rãnh của bản trình diễn: giá trị của trường này tương ứng với khoảng thời gian của rãnh dài nhất trong bản trình diễn. Nếu không xác định được khoảng thời gian (duration) đó từ các rãnh thì khoảng thời gian này sẽ được gán toàn số 1.

*rate*có kiểu số đấu chấm tĩnh 16.16,chỉ tốc độ ưu tiên để trình diễn;1.0 (0x00010000) là tốc độ trình diễnbình thường.

*volume*cũng có kiểu số dấu chấm tĩnh 8.8, chỉmức âm lượng được ưu tiên lựa chọn khi phát. 1.0 (0x0100) chỉ mức âm lượng lớn nhất.

*matrix*chỉ ma trận biến đổi của hình ảnh; (u,v,w) tương ứng là (0,0,1), biểu diễn dưới dạng hexa (0,0,0x40000000).

*next\_track\_ID*có kiểu số nguyên khác 0, chỉ ID của rãnhkế tiếp được bổ sung vàobản trình diễn. Giá trị của next\_*track\_id* cần lớn hơn *track-ID* lớn nhất sử dụng. Nếu next\_*track\_id*toàn là 1 (32 bit maxint) và một rãnh đa phương tiệnmới được thêm vào thìcần tìm kiếm trong tệp tin một rãnh ID chưa sử dụng để gán cho rãnh đa phương tiện mới đó.

## Cấu trúc rãnh

### Hộp rãnh (Track Box)

#### Định nghĩa

Loại hộp:*‘trak’*.

Thuộc đối tượng: Hộp Movie (‘*moov*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một hoặc nhiều.

Hộp Rãnh là hộp chứa của mỗi rãnh đơn lẻtrong một bản trình diễn. Một bản trình diễncó thể có một hoặc nhiều rãnh. Các rãnh độc lập nhauvà mang thông tin về thời gian và không gian của rãnh. Mỗi rãnh sẽ chứa cácHộp Đa phương tiệncó liên quantới rãnh đó.

Các rãnh được sử dụng với hai mục đích: một là để chứa dữ liệu đa phương tiện (rãnh đa phương tiện) và hai là chứa thông tin đóng góicho các giao thức streaming (rãnh chỉ dẫn).

Trong tệp tin ISO sẽ có ít nhất một rãnh đa phương tiện, các rãnh đa phương tiệnnày dùng để tạo nên các rãnh chỉ dẫnsẽ nằm trong cùng một tệp tin, ngay cả khi dữ liệu đa phương tiện đó không được tham chiếu bởi các rãnh chỉ dẫn; nếu xóa tất cả các rãnh chỉ dẫn, toàn bộ bản trình diễnkhông có chỉ dẫn vẫn tồn tại.

#### Cú pháp

*aligned(8) class TrackBox extends Box(‘trak’) {*

*}*

### Hộp Mào đầu Rãnh (Track Header Box)

#### Định nghĩa

Loại hộp: ‘*tkhd*’.

Thuộc đối tượng: Hộp Rãnh (*‘trak’*).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này chỉ các đặc tính của một rãnh. Trong một rãnh chỉ tồn tại duy nhất mộtHộp Mào đầu Rãnh.

Trong trường hợp không códanh sách biên tập,việc trình diễnmột rãnhbắt đầu từphần đầu của toàn bộ bản trình diễn. Một bản biên tập trống được sử dụng dịch chuyểnthời gian bắt đầu của một rãnh.

Giá trị mặc định của các cờ trong mào đầu rãnh đối với các rãnh đa phương tiện là 7 (*track\_enabled*, *track\_in\_movie*, *track\_in\_preview*). Nếu cả hai cờ *track\_in\_movie* và *track\_in\_preview* không được thiết lập trên tất cả các rãnh của bản trình diễn thì tất cả các rãnh sẽ được xử lý giống như khi cả hai cờ đã được thiết lập. Các rãnh chỉ dẫn máy chủsẽ có *track\_in\_movie* và *track\_in\_preview*bằng 0, vì vậy chúng sẽ được bỏ qua trong các chế độ phát lại cục bộ và xem trước.

Với các nhãn ‘iso3’ hoặc các nhãn có cùng yêu cầu, chiều rộng và chiều cao của mào đầu rãnh được đo trên lưới vuông (đồng dạng). Dữ liệu rãnh hình ảnh được chuẩn hóa cho các kích thước (về mặt logic) trước mọi biến đổi hoặc sắp xếp do hệ thống sắp xếp hoặc tổng hợp. Các ma trận rãnh (và movie), nếu được sử dụng cũng hoạt động trong không gian đồng bộ này.

Trường khoảng thời gian nói đến ở đây không bao gồm khoảng thời gian của các phân mảnh movie kế tiếp sau đó, nếu có thì chỉ là khoảng thời gian của đa phương tiện chứa trong Hộp Movie. Hộp Mào đầu Mở rộng Movie có thể được sử dụng để lưu khoảng thời gian nói trên, bao gồm cả phân mảnh movie khi cần và khi có khả năng thực hiện.

#### Cú pháp

*aligned(8) class TrackHeaderBoxextends FullBox(‘tkhd’, version, flags){*

*if (version==1) {*

*unsigned int(64)creation\_time;*

*unsigned int(64)modification\_time;*

*unsigned int(32)track\_ID;*

*const unsigned int(32)reserved = 0;*

*unsigned int(64)duration;*

*} else { // version==0*

*unsigned int(32)creation\_time;*

*unsigned int(32)modification\_time;*

*unsigned int(32)track\_ID;*

*const unsigned int(32)reserved = 0;*

*unsigned int(32)duration;*

*}*

*const unsigned int(32)[2]reserved = 0;*

*template int(16) layer = 0;*

*template int(16) alternate\_group = 0;*

*template int(16)volume = {if track\_is\_audio 0x0100 else 0};*

*const unsigned int(16)reserved = 0;*

*template int(32)[9]matrix= { 0x00010000,0,0,0,0x00010000,0,0,0,0x40000000 };// unity matrix*

*unsigned int(32) width;*

*unsigned int(32) height;*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên, chỉ phiên bản của hộp(0 hoặc 1).

*flags*có kiểu số nguyên 24 bit, xác định:

*Track\_enabled*: cho biết rãnh này đã được kích hoạt hay chưa. Giá trị 0x000001 thể hiện rãnh được kích hoạt, giá trị 0 thể hiện rãnh chưa được kích hoạt.

*Track\_in\_movie*: cho biết rãnh được sử dụng trong bản trình diễn. Giá trị của cờ là 0x000002.

*Track\_in\_preview*: cho biết rãnh được sử dụng khi xem trước bản trình diễn. Giá trị của cờ là 0x000004.

*creation\_time*có kiểu số nguyên, chỉ thời gian tạo ra rãnh (tính theo giây, bắt đầu từ nửa đêm ngày 1 tháng 1 năm 1904 theo thời gian UTC).

*modification\_tine* có kiểu số nguyên, chỉ thời giansửa đổi mới đây nhất của rãnh (tính theo giây, bắt đầu từ nửa đêm ngày 1 tháng 1 năm 1904 theo thời gian UTC).

*track\_ID*có kiểu số nguyên, chỉ định danh duy nhất cho rãnh trên toàn bộ thời gian tồn tại của bản trình diễn. Các rãnh ID không bao giờ được dùng lại và khác 0.

*duration*có kiểu số nguyên,được dùng để chỉ ra khoảng thời gian của rãnh đó (trong thang thời gian được chỉ định trong Hộp Mào đầu Movie). Giá trị của trường này bằng tổng các khoảng thời gian hiệu chỉnh củacác rãnh. Nếu không có danh sách biên tập, khi đó khoảng thời gian bằng tổng các khoảng thời gian mẫu, được chuyển đổi theo thành thang thời gian trong Hộp Mào đầu Movie. Nếu khoảng thời gian của rãnh không xác định được bằng cách trên thì sẽ gán bởi toàn bộ các bit 1.

*layer* thể hiện thứ tự của các rãnh hình ảnh; các rãnhứng với các số thấp sẽđược trình diễn trước. Giá trị 0 là giá trị thông thường, -1 là rãnh trước rãnh 0,...

*alternate\_group*: có kiểu số nguyên chỉ một nhóm hoặc một tập hợp các rãnh. Nếu trường này bằng 0 nghĩa là rãnh này không có liên hệ gì tới các rãnh khác. Nếu trường này khác 0thì các rãnh có cùng giá trị này sẽ cùng chung một nhóm. Chỉ cần một rãnh trong nhóm được trình diễn tại một thời điểm nào đó là có thể biết được các đặc tính của nhóm như tốc độ bit, codec, ngôn ngữ, kích thước gói,...Một nhóm có thể có duy nhất một rãnh.

*volume* có kiểu số dấu chẫm tĩnh 8.8, chỉ mức âm lượng của rãnh. 1.0 (0x0100) chỉ mức âm lượng lớn nhất và là giá trị phổ biến. Giá trị của âm lượng không phù hợp với rãnh hình ảnh. Các rãnh có thể được tạo ra bằng cách kết hợp chúng dựa trên âm lượng, sau đó sử dụng Hôp Mào đầu Movie để thiết lập âm lượng chung, hoặc sử dụng tổ hợp âm thanh phức tạp hơn như là MPEG-4 BIFS.

*matrix* chỉ ma trận biến đổi của hình ảnh; (u,v,w) tương ứng là (0,0,1), biểu diễn dưới dạng hexa (0,0,0x40000000).

*widthvà height* chỉ kích thước hình ảnhtrình diễncủa rãnh thông qua các số dấu chấm tĩnh 16.16. Các giá trịnày không nhất thiết phải giống như kích thước điểm ảnh của hình ảnh được lưu trong miêu tả mẫu; tất cả hình ảnh trong chuỗi hình ảnh có khả năng thay trước khi chuyển đổi thông qua ma trận nêu trên. Kích thước điểm ảnh của hình ảnh là giá trị mặc định.

### Hộp Tham chiếu Rãnh (Track Reference Box)

#### Định nghĩa

Loại hộp: ‘*tref*’.

Thuộc đối tượng: Hộp Rãnh(*‘trak’*).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này cung cấp thông tin tham chiếu từ rãnh chứanày tới các rãnh khác trong bản trình diễn. Các tham chiếu được phân thành nhiều kiểu tham chiếu khác nhau. Tham chiếu “chỉ dẫn“ (hint) chứa các liên kếtcủa rãnh chỉ dẫntrỏ tới dữ liệu đa phương tiện. Tham chiếu miêu tả nội dung ‘*cdsc*’ liên kết thông tin mô tả hoặc các rãnh siêu dữ liệu trỏ tới nội dung mà nó miêu tả. Tham số “*hind”* chỉ các rãnh được tham chiếu có thể chứa dữ liệu đa phương tiện cần thiết cho việc giải mã các rãnh có chứa rãnh tham chiếu. Các rãnh được tham chiếu phải là các rãnh chỉ dẫn. Ví dụ: tham số “*hind”* được dùng để chỉ sự phụ thuộc giữa các rãnh chỉ dẫnghi trong dữ liệu phân lớp IP đa điểm trên RTP.

Trong hộp rãnh chỉ tồn tại duy nhất một Hộp Rãnh Tham chiếu.

Nếu hộp này không tồn tại thì rãnh sẽ không tham chiếu đến bất cứ rãnh nào khác. Mảng chứa thông tin tham chiếu được định cỡ để chứa đủ các hộp kiểu tham chiếu.

#### Cú pháp

*aligned(8) class TrackReferenceBox extends Box(‘tref’) {*

*}*

*aligned(8) class TrackReferenceTypeBox (unsigned int(32) reference\_type) extends*

*Box(reference\_type) {*

*unsigned int(32) track\_IDs[];*

*}*

#### Ngữ nghĩa

Hộp Tham chiếu Rãnh chứa các hộp kiểu tham chiếu rãnh.

*track\_ID*có kiểu số nguyên, cung cấp một tham chiếu từ các rãnh chứa trỏ đến các rãnh khác trong bản trình diễn.*track\_id* không bao giờ được sử dụng lại và khác 0.

*reference\_type* được thiết lậpvới các giá trị sau (hoặc cũng có thể là giá trị được đăng ký, hoặc từ một chỉ tiêu kỹ thuật):

* ‘*hint*’: khi các rãnh được tham chiếu chứa đa phương tiện gốc củacác rãnh chỉ dẫn.
* ‘*cdsc*’: khi rãnh này miêu tả rãnh được tham chiếu.
* *‘hind’*: khi rãnh này phụ thuộc vào rãnh chỉ dẫntham chiếu, ví dụ, *‘hind’* chỉ nên được dùng nếu có sử dụng rãnh chỉ dẫntham chiếu.
* ‘*vdep*’: rãnh này chứa thông tin bổ sung độ sâu hình ảnh cho cácrãnh tham chiếu hình ảnh.
* ‘*vplx*’ rãnh này chứa thông tin hình ảnh thị sai bổ trợ chorãnh tham chiếu hình ảnh.

### Hộp Nhóm Rãnh (Track Group Box)

#### Định nghĩa

Loại hộp: ‘*trgr*’.

Thuộc đối tượng: Hộp Rãnh(‘*trak*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này chỉ thị các nhóm của các rãnh, trong đó mỗi nhóm sẽchung đặc tính hoặc các rãnh trong nhóm có một mối liên hệđặc biệt với nhau. Hộp nàycó thể là hộp rỗng hoặc chứa nhiều hộp bên trong, kiểu của các hộp bên trong sẽ thể hiện những đặc tính cũng như mối liên hệ của các hộp. Các hộp bên trong có định danh riêng dùng để phân biệt rãnh cùng nhóm. Các rãnh này cùng kiểu với các hộp nằm trong Hộp Nhóm Rãnh và có các giá trị định danh giống nhau khi nằm trong các hộp chứa cùng nhóm rãnh.

Các nhóm rãnh không được dùng để chỉ các mối quan hệ có tính phụ thuộc lẫn nhau giữa các rãnh mà Hộp tham chiếu rãnh sẽ đảm nhiệm việc này.

#### Cú pháp

*aligned(8) class TrackGroupBox('trgr') {*

*}*

*aligned(8) class TrackGroupTypeBox(unsigned int(32) track\_group\_type) extends*

*FullBox(track\_group\_type, version = 0, flags = 0){*

*unsigned int(32) track\_group\_id;*

*// the remaining data may be specified for a particular track\_group\_type*

*}*

#### Ngữ nghĩa

*track\_group\_type* cho biết kiểu nhóm và sẽ được thiết lập giá trị sau (hoặc một giá trị được đăng kí, hoặc một giá trị được đề xuất từ chỉ tiêu kỹ thuật):

* ‘*msrc*’ chỉ ra rằng rãnh này nằm trong bản trình diễn đa nguồn. Các rãnh có cùng giá trị *track\_group\_ID* trong Hộp Kiểu Nhóm*track\_group\_type* ‘msrc’ được ánh xạ bắt từ nguồn giống nhau. Ví dụ, một bản ghi của một cuộc điện thoại có hình có thể gồm cả hình và tiếng của cả 2 bên chủ gọi và bị gọi, khi đó giá trị của *track\_group\_ID* ứng với rãnh hình và rãnh tiếng của bên chủ gọi sẽ khác giá trị *track\_group\_ID* của bên bị gọi.

Cặp *track\_group\_ID* và *track\_group\_type* nhận biết một nhóm rãnh trong tệp tin. Các rãnh có hộp kiểu nhóm rãnh cụ thể sẽ có cùng giá trị *track\_group\_ID* của cùng một nhóm rãnh.

## Cấu trúc rãnh đa phương tiện

### Hộp Đa phương tiện (MediaBox)

#### Định nghĩa

Loại hộp: ‘*mdia*’.

Thuộc đối tượng: Hộp Rãnh (*‘trak’*).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp Đa phương tiệnchứa tất cả các đối tượng có khai báo thông tin về dữ liệu đa phương tiệnnằm trong một rãnh.

#### Cú pháp

*aligned(8) class MediaBox extends Box(‘mdia’) {*

*}*

### Hộp Mào đầu Đa phương tiện (Media Header Box)

#### Định nghĩa

Loại hộp: ‘*mdhd*’.

Thuộc đối tượng: Hộp Đa phương tiện(‘*mdia*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp Mào đầu Đa phương tiện chỉ ra các thông tin tổng quan, không phụ thuộc vào dữ liệu đa phương tiện và phù hợp với những đặc tính của đa phương tiện trong một rãnh.

#### Cú pháp

*aligned(8) class MediaHeaderBox extends FullBox(‘mdhd’, version, 0) {*

*if (version==1) {*

*unsigned int(64)creation\_time;*

*unsigned int(64)modification\_time;*

*unsigned int(32)timescale;*

*unsigned int(64)duration;*

*} else { // version ==0*

*unsigned int(32)creation\_time;*

*unsigned int(32)modification\_time;*

*unsigned int(32)timescale;*

*unsigned int(32)duration;*

*}*

*bit(1) pad = 0;*

*unsigned int(5)[3] language; // ISO-639-2/T language code*

*unsigned int(16)pre\_defined = 0;*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên, chỉ phiên bản của hộp(0 hoặc 1).

*creation\_time* có kiểu số nguyên, chỉ thời gian tạo củađa phương tiện trong rãnh (tính theo giây, bắt đầu từ nửa đêm ngày 1 tháng 1 năm 1904 theo thời gian UTC).

*modification\_tine* có kiểu số nguyên, chỉ thời gian sửa đổi mới đây nhất của đa phương tiện trong rãnh (tính theo giây, bắt đầu từ nửa đêm ngày 1 tháng 1 năm 1904 theo thời gian UTC).

*timescale* có kiểu số nguyên, chỉ thang đo thời gian cho đa phương tiện; đây là số đơn vị thời gian trôi qua trong một giây. Ví dụ, một hệ thống phối hợp thời gian đo thời gian của một phần 60 giây có thang là 60.

*duration*có kiểu số nguyênchỉ khoảng thời gian của đa phương tiện (trong thang đo của thang thời gian). Nếu không xác định được khoảng thời gian thì hệ thống sẽ gán giá trị toàn bit 1.

*language* chỉ mã ngôn ngữ củađa phương tiện. Xem ISO 639-2/T để biết các mã ngôn ngữ 3 ký tự.

### Hộp Tham chiếu Xử lý (Handler Reference Box)

#### Định nghĩa

Loại hộp: ‘*hdlr*’.

Thuộc đối tượng: Hộp Đa phương tiện(‘*mdia*’) hoặc Hộp Meta (‘*meta*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này nằm trong Hộp Đa phương tiện, được dùng để khai báo kiểu đa phương tiện của rãnh vàthể hiện quá trình xử lý dữ liệu đa phương tiện có trên rãnh. Ví dụ, định dạng mà bộ giải mã dùng để giải mã hình ảnh cần được lưu trong rãnh hình ảnh, định dạng đó được nhận dạng thông qua việc xử lý của bộ xử lý hình ảnh. Thông tin lưu trữ định dạng đa phương tiện sẽ xác định kiểu đa phương tiện mà định đạng đó sử dụng.

Nếu hộp này khi xuất hiện trong Hộp Meta thì nó sẽ được dùng để khai báo cấu trúc hoặc định dạng của nội dung chứa trong hộp ‘*meta*’.

Các kiểu luồng siêu dữ liệusẽ có chung một bộ xử lý; định dạng cụ thể được xác định bởi thông tin đầu vào của mẫu, ví dụ như hình ảnh hoặc âm thanh.

Rãnh hình ảnhbổ trợ được mã hóa giống như rãnh hình ảnh, tuy nhiên, rãnh hình ảnh bổ trợ sử dụng kiểu xử lý khác và không nhằm mục đích để hiển thị hình ảnh (ví dụ nó chứa thông tin độ sâu, hoặc tínhđơn sắchoặc thông tin màu 2 chiều). Các rãnh hình ảnhbổ trợ thường liên kết với một rãnh hình ảnhthông qua một rãnh tham chiếu tương ứng.

Kiểu văn bản đa phương tiện sắp xếp theo thời gian cho bộ giải mãtương ứng biết rằng nó sẽ chỉ xử lý dữ liệu dạng văn bản. Kiểu phụ đề đa phương tiện cũng sẽ cho bộ giải mã tương ứng biết nó sẽ xử lý dữ liệu dạng văn bản và có thê có cả hình ảnh.

LƯU Ý: các luồng MPEG-7 là một kiểu luồng siêu dữ liệu, có bộ xử lý riêng được miêu tả trongphần định dạng tệp tin MP4 [ISO/IEC 14496-14].

LƯU Ý: các rãnh siêu dữ liệu liên kết với rãnh mà chúng đề cập đến thông qua kiểu tham chiếu rãnh ‘*cdsc*’. Các rãnh siêu dữ liệu sử dụng một mào đầu đa phương tiệntrống (‘*nmhd*’), như mô tả trong 8.4.5.5.

#### Cú pháp

*aligned(8) class HandlerBox extends FullBox(‘hdlr’, version = 0, 0) {*

*unsigned int(32)pre\_defined = 0;*

*unsigned int(32)handler\_type;*

*const unsigned int(32)[3]reserved = 0;*

*string name;*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*handler\_type*nếu xuất hiện trong Hộp Đa phương tiện sẽ có kiếu số nguyên, chứa một trong các giá trị sau đây hoặc một giá trị được lấy từ một chỉ tiêu kỹ thuật:

* ‘*vide*’:rãnh hình ảnh.
* ‘*soun*’:rãnh âm thanh.
* ‘*hint*’:rãnh chỉ dẫn.
* ‘*meta*’: rãnh siêu dữ liệu sắp xếp theo thời gian.
* ‘*auxv*’: rãnh phụ trợ hình ảnh.
* ‘text’: rãnh văn bản sắp xếp theo thời gian.
* ‘subt’: rãnh phụ đề.

*handler\_type*nếu xuất hiện trong Hộp Siêu dữ liệu sẽ chứa một giá trị thích hợp để xác định định dạng nội dung của Hộp Siêu dữ liệu. Giá trị ‘null’ có thể được sử dụng trong Hộp Siêu dữ liệu để thể hiện rằng nó chỉ được sử dụng để giữ các tài nguyên.

*name* là một chuỗi kí tự UTF-8 được kết thúc bởi kí tự trốngcho biết tên kiểu của rãnh mà bất kỳ ai cũng có thể đọc được(nhằm mục đích tìm lỗi và kiểm tra).

### Hộp Thông tin Đa phương tiện (Media Information Box)

#### Định nghĩa

Loại hộp:‘*minf*’.

Vật chứa:Hộp Đa phương tiện(‘*mdia*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này chứa tất cả các đối tượng mà các đối tượng đó chứa thông tin về đa phương tiện trong rãnh.

#### Cú pháp

*aligned(8) class MediaInformationBox extends Box(‘minf’) {*

*}*

### Hộp Mào đầu Thông tin Đa phương tiện (Media Information Header Box)

#### Định nghĩa

Mẫ hộp: ‘*vmhd*’, ‘*smhd*’, ’*hmhd*’, ‘*nmhd*’.

Thuộc đối tượng: Hộp Thông tin Đa phương tiện (‘*minf*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Mỗi kiểu rãnh sẽ có một mào đầu thông tin đa phương tiện riêng (ứng với kiểu xử lý đa phương tiện) được mô tả trong tài liệu này hoặc trong một chỉ tiêu kỹ thuật phù hợp.

Kiểu mào đầu đa phương tiện được xác định thông qua bộ xử lý đa phương tiện:

* Rãnh hình ảnh VideoMediaHeaderBox.
* Rãnh âm thanh SoundMediaHeaderBox.
* Rãnh siêu dữ liệu định thời NullMediaHeaderBox.
* Rãnh văn bản định thời NullMediaHeaderBox.
* Rãnh phụ đề SubtitleMediaHeaderBox.
* Rãnh chỉ dẫn HintMediaHeaderBox.

#### Hộp Mào đầu Đa phương tiện Hình ảnh (Video Media Header Box)

Mào đầu đa phương tiệnhình ảnh chứa thông tintrình diễntổng quát cho đa phương tiện hình ảnh, độc lập với việc mã hóa. Lưu ý rằng trường cờ có giá trị 1.

##### Cú pháp

*aligned(8) class VideoMediaHeaderBox extends FullBox(‘vmhd’, version = 0, 1) {*

*template unsigned int(16)graphicsmode = 0; // copy, see below*

*template unsigned int(16)[3]opcolor = {0, 0, 0};*

*}*

##### Ngữ nghĩa

*version*có kiểu số nguyênchỉ phiên bản của hộp.

*graphicsmode*chỉ dạng tổ hợp của rãnh hình ảnh được liệt kê: copy=0 sao chép đè lên hình ảnh hiện tại.

*opcolor* là bộ ba giá trị màu (đỏ, xanh lá cây, xanh dương) được sử dụng cho các chế độ đồ họa.

#### Hộp Mào đầu Đa phương tiện Âm thanh (Sound Media Header Box)

Mào đầu âm thanh đa phương tiện chứa thông tin bản trình diễntổng quát cho âm thanh, độc lập việc mã hóa. Đây là mào đầu được sử dụng cho tất cả các rãnh chứa âm thanh.

##### Cú pháp

*aligned(8) class SoundMediaHeaderBox extends FullBox(‘smhd’, version = 0, 0) {*

*template int(16) balance = 0;*

*const unsigned int(16)reserved = 0;*

*}*

##### Ngữ nghĩa

*version* có kiểu số nguyên chỉ phiên bản của hộp.

*balance* là số dấu chấm tĩnh 8.8 thể hiện việc điều chỉnh âm thanh trái phải trong không gian âm thanh nổi (stereo); 0 là trung tâm (giá trị thông thường); -1.0thể hiện âm thanh bên trái, 1.0 thể hiện âm thanh bên phải.

#### Hộp Mào đầu Đa phương tiện Chỉ dẫn (Hint Media Header Box)

Mào đầu đa phương tiệnchỉ dẫn chứa thông tin tổng quát của các rãnh chỉ dẫn, độc lập với giao thức.(PDU là đơn vị dữ liệu giao thức).

##### Cú pháp

*aligned(8) class HintMediaHeaderBox extends FullBox(‘hmhd’, version = 0, 0) {*

*unsigned int(16)maxPDUsize;*

*unsigned int(16)avgPDUsize;*

*unsigned int(32)maxbitrate;*

*unsigned int(32)avgbitrate;*

*unsigned int(32)reserved = 0;*

*}*

##### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*maxPDUsized* cho biết kích thước theo byte lớn nhất của PDU trong luồng chỉ dẫn.

*avgPDUsize* cho biết kích thước trung bình của một PDU tính trên toàn bộ bản trình diễn.

*maxbitrate* cho biết tốc độ bit lớn nhất tính theo số bit/giây.

*avgbitrate* cho biết tốc độ bit trung bình tính theo số bit/giây tính trên toàn bộ bản trình diễn.

#### Hộp Mào đầu Đa phương tiện Trống (Null Media Header Box)

Các luồng có mào đầu đa phương tiện không xác định sử dụng Hộp Mào đầu Đa phương tiệnTrống như mô tả trong phần này.

##### Cú pháp

*aligned(8) class NullMediaHeaderBox extends FullBox(’nmhd’, version = 0, flags) {*

*}*

##### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*flags*có kiểu số nguyên 24 bit (hiện có giá trị 0).

#### Hộp Mào đầu Đa phương tiện Phụ đề (Subtitle Media Header Box)

Hộp này chứa thông tin bản trình diễntổng quát của đa phương tiện phụ đề, độc lập với việc mã hóa và được sử dụng cho tất cả các rãnh chứa phụ đề.

##### Cú pháp

*aligned(8) class SubtitleMediaHeaderBox extends FullBox (‘sthd’, version = 0, flags = 0){*

*}*

##### Ngữ nghĩa

*version*có kiểu số nguyên cho biết phiên bản của hộp đó.

*flags*có kiểu số nguyên 24 bit (hiện có giá trị 0).

## Các bảng mẫu

### Hộp Bảng Mẫu(Sample Table Box)

#### Định nghĩa

Loại hộp: ‘*stbl*’.

Thuộc đối tượng: Hộp Thông tin Đa phương tiện (‘*minf*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Bảng mẫu chứa tất cả các chỉ số về thời gian và dữ liệu của các mẫu đa phương tiện trong một rãnh.Việc sử dụng các bảng cho phép định vị các mẫu theo theo thời gian, xác định kiểu mẫu (ví dụ xác định có phải là khung hình I hay không), đồng thời xác định kích thước, hộp chứa và độ dịch chuyển của mẫu trong hộp chứa.

Nếu rãnhchứaHộp Bảng Mẫukhông tham chiếu đến dữ liệu nào thì Hộp Bảng Mẫuđó không nhất thiết phải chứa bất kỳhộp con nào (đây không phải là rãnh đa phương tiện hữu dụng).

Nếu rãnh chứaHộp Bảng Mẫutham chiếu đến dữ liệu khác thì cần cócác hộp con sau:Hộp Miêu tả Mẫu, Hộp Kích thước Mẫu, Hộp Ánh xạ Mẫu vào đoạn, Hộp Độ dịch chuyển Đoạn. Ngoài ra, Hộp Miêu tả Mẫu cần chứa ít nhất một đầu vào mục. Việc sử dụng Hộp Miêu tả Mẫu là cần thiết bởi vì nó chứa trường chỉ số tham chiếu dữ liệu dùng để chỉ Hộp Tham chiếu Dữ liệu, mà Hộp Tham chiếu dữ liệu này lại dùng đề lấy các mấu đa phương tiện. Nếu không có Miêu tả Mẫu thì không thể xác định được các mẫu đa phương tiện được lưu ở đâu.

Phần phụ lục A.7 miêu tả việc truy nhập ngẫu nhiên sử dụng các cấu trúc được chỉ ra trong Hộp Bảng Mẫu.

#### Cú pháp

*aligned(8) class SampleTableBox extends Box(‘stbl’) {*

*}*

### Hộp Miêu tả Mẫu (Sample Description Box)

#### Định nghĩa

Loại hộp: ‘*stsd*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Bảng miêu tả mẫu cho biết thông tin chi tiết về kiểu mã hóa được sử dụng và các thông tin ban đầu cần thiết cho việc mã hóa.

Thông tin được lưu trong Hộp Miêu tả Mẫu sau *entry-count* có thể là đặc thù cho từng kiểu rãnh như được trình bày trong tài liệu này và cũng có thể có các biến thể trong cùng một kiểu rãnh (ví dụ các kiểu mã hóa khác nhau có thể sử dụng các thông tin cụ thể khác nhau sau một số trường thông tin chung, kể cả trong cùng một rãnh hình ảnh).

Các kiểu định dạng thông tin đầu vào mục mẫu được xác định bởi các bộ xử lý đa phương tiện:

* Rãnh hình ảnh VisualSample*Entry*
* Rãnh âm thanh AudioSampleEntry
* Rãnh siêu dữ liệu định thời MetaDataSampleEntry
* Rãnh văn bản định thời PlainTextSampleEntry
* Rãnh phụ đề SubtitleSampleEntry
* Rãnh chỉ dẫn Định dạng mụcđặc thù cho giao thức với tên tương ứng.

Đối với các rãnh chỉ dẫn, miêu tả mẫu chứa dữ liệu khai báo thích hợp cho giao thức streaming được sử dụng và định dạng của rãnh chỉ dẫn. Các giao thức khác nhau có định nghĩa miêu tả mẫu khác nhau.

Có thể có nhiều miêu tả được sử dụng trong một rãnh.

Các trường ‘protocol’ và ‘codingname’ là những định danh được đăng ký để nhận dạng giao thức streaming hoặc bộ giải mã định dạng nén. Một giao thức (protocol) hoặc tên mã (codingname) cụ thể có thể có phần mở rộng tùy chọn hoặc bắt buộc đối với miêu tả mẫu (ví dụ các tham số thiết lập codec). Tất cả các phần mở rộng đó sẽ nằm trong các hộp; các hộp này xuất hiện sau các trường được yêu cầu. Các hộp không được nhận dạng sẽ bị bỏ qua.

Nếu trường ‘format’ của SampleEntry không được nhận dạng thì bản thân miêu tả mẫu và các mẫu đa phương tiện liên quan đều không được giải mã.

LƯU Ý: định nghĩa của các đầu vào mục mẫu quy định thứ tự cụ thể của các hộp, trong các tiêu chuẩn dẫn xuất cũng thường tuân theo thứ tự này. Để tương thích tối đa, bộ ghi cần xây dựng các tệp tin tương tứng theo thứ tự trong cả tiêu chuẩn và các phần được kế thừa, còn các bộ đọc cần chuẩn bị trước khả năng chấp nhận thứ tự hộp bất kỳ.

Các trường *samplerate*,*samplesize* và*channelcount* lưu định dạng trình diễn âm thanhđầu ra mặc định của đa phương tiện. Thang thời gian của một rãnhâm thanh cần lựa chọn phù hợp với tần số lấy mẫu, hoặc là bội số nguyên của nó, để cho phép các mẫu được định thời cơ bản xác. ChannelCount là giá trị lớn hơn không và thể hiện số kênh âm thanh lớn nhất có thể đươc phát đi. ChannelCount bằng 1 thể hiện âm thanh đơn sắc (mono), bằng 2 thể hiện âm thanh nổi (stereo). Nếu ChannelCount có giá trị lớn hơn 2 thì bộ mã hóa cần được cấu hình để nhận dạng được các kênh âm thanh đó.

Khi muốn thể hiện việc lấy mẫu âm thanh lớn hơn giá trị lưu trong trường tần số lấy mẫu thì có thể sử dụng:

* AudioSampleEntryV1, với yêu cầu phải kèm theo Hộp Miêu tả Mẫu phiên bản 1;
* HộpTần số Lấy mẫu, hộp này có thể chỉ xuất hiện trong AudioSampleEntryV1 và khi xuất hiện, nó sẽ thay giá trị trong trường tần số lấy mẫu bởi tần số lấy mẫu thực tế;
* Khi xuất hiện HộpTần số Lấy mẫu, thang thời gian đa phương tiện sẽ giống như tần số lấy mẫu hoặc bằngphần nguyên, hay bằng bội số của tần số lấy mẫu;
* Trường tần số lấy mẫu trong mục mẫu sẽ chứa một giá trị dịch chuyển trái 16 bit (như đối với AudioSampleEntry) thích hợp với thang thời gian đa phương tiện, hoặc phần nguyên hay bằng bội số của thang thời gian.

AudioSampleEntryV1 sẽ chỉ được sử dụng khi cần thiết, nếu không thì để tương thích tối đa, AudioSampleEntry sẽ được sử dụng. AudioSampleEntryV1 không xuất hiện trong Hộp Miêu tả Mẫu với phiên bản 0.

Tất cả các trường *TextSubtitleSampleEntry*, *TextMetaDataSampleEntry*, hoặc *SimpleTextSampleEntry*đều chứa một kiểu MIME và có thể được sử dụng để xác định các luồng sử dụng kiểu MIME. Kiểu MIME sẽ được dùng nếu các nội dung của một tập mẫu, bắt đầu với một mẫu đồng bộ và kết thúc tại mẫu ngay trước mẫu đồng bộ kế tiếp, được liên kết với nhau, kết quả thu được đáp ứng những yêu cầu giải mã của các tài liệu dạng MIME. Các mẫu không đồng bộ chỉ được sử dụng nếu định dạng đó thể hiện hành vi của ‘tiến trình giải mã’ và thời gian mẫu thể hiện khi nào kết quả của tiến trình giải mã nên hiện diện (theo kiểu đa phương tiện).

LƯU Ý: các mẫu trong một rãnh chỉ bao gồm toàn các mẫu đồng bộ, vì thế, sẽ là một tài liệu tương thích với kiểu MIME đó.

Trong cácrãnh hình ảnh, trường *frame\_count* phải có giá trị là 1, trừ khi tiêu chuẩn của định dạng đa phương tiện này qui định giá trị và cho phép các giá trị lớn hơn. Tiêu chuẩn này phải ghi cả cách các khung hình ảnh riêng lẻ được tìm thấy (thông tin kích thước của chúng) và thời điểm chúng được thiết lập. Việc xác định thời điểm cần đơn giản như chia khoảng thời gian mẫu cho *frame\_count* để tính ra khoảng thời gian của khung.

Lưu ý: cho dù biến count có tới 32 bit, số lượng các mục dữ liệu thường ít hơn rất nhiều và bị hạn chế bởi chỉ số tham chiếu trong bảng mẫu chỉ là 16 bit.

Một tùy chọn BitRateBox có thể xuất hiện tại cuối của mọi MetaDataSampleEntry để biểu thị thông tin tốc độ bit của một luồng. Thông tin này có thể được sử dụng cho cấu hình bộ đệm. Trong trường hợp của siêu dữ liệu XML nó có thể được sử dụng để lựa chọn định dạng biểu diễn bộ nhớ thích hợp (DOM, STX).

Chiều rộng (*width*) và chiều cao (*height*) trong đầu vào mục mẫu hình ảnh sẽ là dữ liệu để tính số điểm ảnh mà bộ mã hóa sẽ phân phối; điều này cho phép cấp phát các bộ đệm. Vì đây là các số đếm nên chúng không quan tâm đến tỉ lệ các điểm ảnh.

Tỉ lệ các điểm ảnh và khẩu độ rõ ràng của hình ảnh có thể được chỉ định cụ thể sử dụng các hộp đầu vào mục mẫu ‘pasp’ và ‘clap’. Có cả hai tùy chọn, nếu tồn tại, chúng sẽ thay thế các khai báo (nếu có) trong cấu trúc riêng để mã hóa giải mã hình ảnh, mà cấu trúc này cần được khảo sát nếu các hộp này không tồn tại. Để tương thích tối đa, các hộp này cần tuân theo cấu trúc bất cứ hộp nào đã được định nghĩa hoặc được yêu cầu theo các tiêu chuẩn kỹ thuật.

Trong Hộp PixelAspectRatioBox, hSpacing và vSpacing có cùng đơn vị, nhưng các đơn vị này không được chỉ rõ: duy nhất vấn đề tỉ lệ. hSpacing và vSpacing có thể hoặckhông thể giảm các mục và chúng có thể giảm tới 1/1. Cả hai phải là số dương.

Chúng được xác định theo phương diện tỉ lệ của một điểm ảnh, với đơn vị tùy ý. Nếu một điểm ảnh xuất hiện theo độ rộng H và độ cao V, khi đó hSpacing/vSpacing là bằng H/V. Điều đó nghĩa là một ô vuông trên màn hình với độ cao là n điểm ảnh cần tới n\*vSpacing/hSpacing điểm ảnh theo chiều rộng để xuất hiện hình vuông.

LƯU Ý:Khi điều chỉnh tỉ lệ theo phương diện điểm ảnh, thông thường, kích thước nằm ngang của hình ảnhđược định cỡ, nếu cần thiết (ví dụ. nếu hệ thống hiển thị cuối cùng có tỉ lệ phương diện điểm ảnh khác nhau từ cùng nguồn hình ảnh).

LƯU Ý: Điều này được khuyến nghị cho các điểm ảnh gốc, bao gồm cả việc biến đổi, được thực hiện song song nếu có thể. Nếu kết quả từ việc thực hiện biến đổi ‘cơ bản xác’ tỉ lệ theo phương diện điểm ảnh là một lưới ô vuông, để chuẩn hóa kích thước rãnh, các thành phần hoặc vị trí (ví dụ. rãnh và/hoặc ma trận phim ảnh) và chuẩn hóa các đặc trưng hiển thị là một ma trận đơn vị, sau đó không cần thực hiện lấy mẫu lại. Đặc biệt, hình ảnh không nên lấy mẫu lại nhiều hơn một lần trong quá trình kết xuất nếu có thể.

Có 4 giá trị danh nghĩa trong *CleanApertureBox*.Các tham số này được biểu diễn như một phân số N/D. Phân số này có thể ở dạng tối giản hoặc không. Cặp tham số *fooN* và *fooD* được biểu diễn bởi tham số *foo*.Đối với *horizOff* và *vertOff*, D (mẫu số) phải là số dương và N (tử số) có thể dương hoặc âm. Đối với *cleanApertureWidth* và *cleanApertureHeight*, cả N và D phải là số dương.

LƯU Ý: Những giá trị này được biểu diễn ở dạng phân số là do một vài lý do sau đây. Thứ nhất, trong một hệ thống độ cơ bản xác của chiều rộng sau khi hiệu chỉnh tỉ lệ theo phương diện điểm ảnh là nguyên, không phải đếm số điểm ảnh trước khi hiệu chỉnh. Thứ hai, nếu video được thay đổi kích thước trong toàn bộ khẩu độ, sự biểu diễn cơ bản xác cho khẩu độ rõ không phải là nguyên. Cuối cùng, bởi vì đây là biểu diễn sử dụng trung tâm và độ dịch chuyển, một phép chia hai là cần thiết và vì vậy một phần hai giá trị có thể xuất hiện.

Việc xem xét kích thước điểm ảnh được xác định bởi chiều rộng(*width*) và chiều cao (*height)* VisualSampleEntry. Nếu tọa độ trung tâm của hình ảnh tại *pcX* và *pcY*, thì *horizOff* và *vertOff* được xác định như sau:

*pcX = horizOff + (width- 1)/2*

*pcY = vertOff+ (height - 1)/2;*

Thường thì, *horizOff* và *vertOff* bằng 0, vì vậy hình ảnh được định vị ở trung tâm khung hình.

Các điểm ảnh xa nhất về phía trái/phía phải và phía trên/phía dưới đường khẩu độ rõ ràng nằm tại:

*pcX ± (cleanApertureWidth - 1)/2*

*pcY ± (cleanApertureHeight - 1)/2;*

Định dạng đầu ra âm thanh (các trường *samplerate*, *samplesize* và *channelcount*) trong đầu vào mục mẫu cần được xem xét duy nhất cho các bộ mã hóa giải mã không ghi cấu hình đầu ra. Nếu các bộ mã hóa giải mã âm thanh có thông tin cuối cùng về định dạng đầu ra, nó sẽ lấy thông tin đó; trong trường hợp các trường *samplerate*, *samplesize* và *channelcount*trong đầu vào mục mẫu bị bỏ qua thì có thể lựa chọn các giá trị phù hợp (ví dụ, tần số lấy mẫu khả thi cao nhất).

Trong một hộp, đầu vào mục *URIMetaSampleEntry* chứa URI xác định dạng của siêu dữ liệu và dữ liệu khởi tạo tùy chọn. Định dạngcủa cả các mẫu và dữ liệu được khởi tạo được xác định bởi toàn bộ hoặc một phần của URI.

Thông thường, một hộp tỉ lệ bit tùy chọn có thể được sử dụng trong đầu vào mục *URIMetaSampleEntry*.

Có thể có trường hợp URI xác định một định dạng của siêu dữ liệu mà cho phép có nhiều hơn một ‘stated fact’ trong phạm vi mỗi mẫu. Tuy nhiên, tất cả các mẫu siêu dữ liệu trong định dạng này là ‘các khung hình I*’*hiệu quả, xác định toàn bộ các siêu dữ liệu cho khoảng thời gian của các khung hình I đó. Điều này có nghĩa là tập hợp đầy đủ các siêu dữ liệu tại mọi thời điểm củamột rãnh, nằm trong (a) các mẫu cùng thời gian của các rãnh (nếu có) miêu tả rãnh này, cộng với (b)siêu dữ liệu rãnh (nếu có), siêu dữ liệu movie (nếu có) và siêu dữ liệu tệp tin (nếu có).

Nếu cần siêu dữ liệu thay đổi từng bước, thì nền tảng MPEG-7 sẽ cung cấp khả năng đó.

Thông tin về định dạng URI đối với một số hệ thống siêu dữ liệu được trình bày trong phụ lục G.

Thông tin về màu có thể được cung cấp trong một hoặc nhiều Hộp Thông tin Màu*(ColourInformationBoxes)* đặt trong *VisualSampleEntry*. Các hộp cần được đặt theo thứ tự trong đầu vào mục mẫu bắt đầu với các hộp độ cơ bản xác cao nhất (và có khả năng là khó xử lý nhất) tới hộp có độ cơ bản xác thấp nhất. Việc này chỉ có tính khuyến nghị, không bắt buộc, liên quan tới việc dựng hình và chuyển đổi màu và không có tính quy phạm; đầu đọc có thể chọn hình thức phù hợp nhất. Hộp Thông tin Màuvới kiểu màu không nhận dạng được có thể bị bỏ qua.

Nếu được sử dụng, thông tin ICC có thể có một hạn chế theo mã ‘rICC’, cho phép xử lý đơn giản hơn. Thông tin đó sẽ là thuộc một trong hai lớp Monochrome hoặc Three-Component Matrix-Based của thông tin đầu vào, như được định nghĩa bởi ISO 15076-1. Nếu thông tin này thuộc lớp khác, khi đó chỉ thị ‘prof’phải được sử dụng.

Nếu thông tin màu được cung cấp trong cả hai hộp này và trong luồng bit hình ảnh, hộp này được ưu tiên và thay thế thông tin trong luồng bit.

LƯU Ý: khi thông tin ICC được chỉ rõ, SMPTE RP 177 “Derivation of Basic Television Color Equations “có thể được sử dụng nếu cần tạo ra ma trận chuyển đổi màu cơ bảntừ Y'CbCr sang R'G'B' miêu tả bởi thông tin ICC.

#### Cú pháp

*aligned(8) abstract class SampleEntry (unsigned int(32) format) extends Box(format){*

*const unsigned int(8)[6] reserved = 0;*

*unsigned int(16) data\_reference\_index;*

*}*

*class HintSampleEntry() extends SampleEntry (protocol) {*

*unsigned int(8) data [];*

*}*

*class BitRateBox extends Box(‘btrt’){*

*unsigned int(32) bufferSizeDB;*

*unsigned int(32) maxBitrate;*

*unsigned int(32) avgBitrate;*

*}*

*class MetaDataSampleEntry(codingname) extends SampleEntry (codingname) {*

*}*

*class XMLMetaDataSampleEntry() extends MetaDataSampleEntry (’metx‘) {*

*stringcontent\_encoding; // optional*

*stringnamespace;*

*string schema\_location; // optional*

*BitRateBox (); // optional*

*}*

*class TextMetaDataSampleEntry() extends MetaDataSampleEntry (‘mett’) {*

*stringcontent\_encoding; // optional*

*string mime\_format;*

*BitRateBox (); // optional*

*}*

*aligned(8) class URIBox extends FullBox(‘uri ’, version = 0, 0) {*

*string theURI;*

*}*

*aligned(8) class URIInitBox extends FullBox(‘uriI’, version = 0, 0) {*

*unsigned int(8) uri\_initialization\_data[];*

*}*

*class URIMetaSampleEntry() extends MetaDataSampleEntry (’urim‘) {*

*URIbox the\_label;*

*URIInitBox init;// optional*

*MPEG4BitRateBox (); // optional*

*}*

*// Visual Sequences*

*class PixelAspectRatioBox extends Box(‘pasp’){*

*unsigned int(32) hSpacing;*

*unsigned int(32) vSpacing;*

*}*

*class CleanApertureBox extends Box(‘clap’){*

*unsigned int(32) cleanApertureWidthN;*

*unsigned int(32) cleanApertureWidthD;*

*unsigned int(32) cleanApertureHeightN;*

*unsigned int(32) cleanApertureHeightD;*

*unsigned int(32) horizOffN;*

*unsigned int(32) horizOffD;*

*unsigned int(32) vertOffN;*

*unsigned int(32) vertOffD;*

*}*

*class ColourInformationBox extends Box(‘colr’){*

*unsigned int(32) colour\_type;*

*if (colour\_type == ‘nclx’) /\* on-screen colours \*/ {*

*unsigned int(16) colour\_primaries;*

*unsigned int(16) transfer\_characteristics;*

*unsigned int(16) matrix\_coefficients;*

*unsigned int(1)full\_range\_flag;*

*unsigned int(7)reserved = 0;*

*} else if (colour\_type == ‘rICC’){*

*ICC\_profile; // restricted ICC profile*

*} else if (colour\_type == ‘prof’) {*

*ICC\_profile; // unrestricted ICC profile*

*}*

*}*

*class VisualSampleEntry(codingname) extends SampleEntry (codingname){*

*unsigned int(16) pre\_defined = 0;*

*const unsigned int(16) reserved = 0;*

*unsigned int(32)[3]pre\_defined = 0;*

*unsigned int(16)width;*

*unsigned int(16)height;*

*template unsigned int(32)horizresolution = 0x00480000; // 72 dpi*

*template unsigned int(32)vertresolution= 0x00480000; // 72 dpi*

*const unsigned int(32)reserved = 0;*

*template unsigned int(16)frame\_count = 1;*

*string[32]compressorname;*

*template unsigned int(16)depth = 0x0018;*

*int(16)pre\_defined = -1;*

*// other boxes from derived specifications*

*CleanApertureBox clap; // optional*

*PixelAspectRatioBoxpasp;// optional*

*}*

*// Audio Sequences*

*aligned(8) class SamplingRateBox extends FullBox(‘srat’) {*

*unsigned int(32) sampling\_rate;*

*}*

*class AudioSampleEntry(codingname) extends SampleEntry (codingname){*

*const unsigned int(32)[2] reserved = 0;*

*template unsigned int(16) channelcount = 2;*

*template unsigned int(16) samplesize = 16;*

*unsigned int(16) pre\_defined = 0;*

*const unsigned int(16) reserved = 0 ;*

*template unsigned int(32) samplerate = { default samplerate of media}<<16;*

*aligned(8) class SamplingRateBox extends FullBox(‘srat’) {*

*unsigned int(32) sampling\_rate;*

*}*

*class AudioSampleEntryV1(codingname) extends SampleEntry (codingname){*

*const unsigned int(16) audioentryversion = 1;*

*const unsigned int(16) reserved = 0;*

*const unsigned int(32) reserved = 0;*

*template unsigned int(16) channelcount = 2;*

*template unsigned int(16) samplesize = 16;*

*unsigned int(16) pre\_defined = 0;*

*const unsigned int(16) reserved = 0 ;*

*template unsigned int(32) samplerate = {suitable rate from timescale << 16};*

*SamplingRateBox(); // optional but normally present*

*}*

*// Timed Text Sequences*

*class PlainTextSampleEntry(codingname) extends SampleEntry (codingname) {*

*}*

*class SimpleTextSampleEntry(codingname) extends PlainTextSampleEntry (codingname) {*

*string content\_encoding; // optional*

*string mime\_format;*

*BitRateBox (); // optional*

*}*

*// Subtitle Sequences*

*class SubtitleSampleEntry(codingname) extends SampleEntry (codingname) {*

*}*

*class XMLSubtitleSampleEntry() extends SubtitleSampleEntry (’stpp‘) {*

*string namespace;*

*string schema\_location; // optional*

*string auxiliary\_mime\_types;*

*// optional, required if auxiliary resources are present*

*BitRateBox (); // optional*

*}*

*class TextSubtitleSampleEntry() extends SubtitleSampleEntry (‘sbtt’) {*

*string content\_encoding; // optional*

*string mime\_format;*

*BitRateBox (); // optional*

*}*

*aligned(8) class SampleDescriptionBox (unsigned int(32) handler\_type) extends FullBox('stsd', version, 0){*

*int i ;*

*unsigned int(32) entry\_count;*

*for (i = 1 ; i <= entry\_count ; i++){*

*switch (handler\_type){*

*case ‘soun’: // for audio tracks*

*AudioSampleEntry();*

*break;*

*case ‘vide’: // for video tracks*

*VisualSampleEntry();*

*break;*

*case ‘subt’: // for subtitle tracks*

*SubtitleSampleEntry();*

*break;*

*case ‘text’: // for plain text tracks*

*TextSampleEntry();*

*case ‘hint’: // Hint track*

*HintSampleEntry();*

*break;*

*case ‘meta’: // Metadata track*

*MetadataSampleEntry();*

*break;*

*}*

*}*

*}*

*}*

#### Ngữ nghĩa

*version* luôn nhận giá trị 0 trừ khi hộpcó chứa AudioSampleEntryV1, khi đó *version*sẽ nhậngiá trị 1.

*entry\_count* có kiểu số nguyên cho biết số lượng các mục.

*SampleEntry*là thông tin đầu vào mẫu thích hợp.

*data\_reference\_index* là số nguyên chứa chỉ số tham chiếu dữ liệu sử dụng để thu thập dữ liệu liên quan tới các mẫu sử dụng miêu tả mẫu này. Dữ liệu tham chiếu được lưu trong các Hộp Tham chiếu Dữ liệu. Khoảng chỉ số được tính từ một đến số lượng của các tham chiếu dữ liệu.

*ChannelCount* là số lượng kênh, 1 (mono) hoặc 2 (stereo).

*SampleSize* là các bit, có giá trị mặc định là 16.

*SampleRate* là tần số lấy mẫu khi không tồn tại SamplingRateBox; khi có SamplingRateBox thì *SampleRate*sẽ có kiểu số nguyên (32 bit, được biểu diễn bởi số dấu chấm động 16.16) với giá trị là bội số hoặc ước số của tần số lấy mẫu thực tế.

*sampling\_rate* là tần số lấy mẫu thực tế của âm thanh, được biểu diễn bởi số nguyên 32 bit.

*resolution* là số dấu chấm tĩnh 16.16 biểu diễn độ phân giải của hình ảnh theo đơn vị số điểm ảnh trên một inch (2,54 cm).

*frame\_count*cho biết có bao nhiêu khung của hình ảnh nén được lưu trong mỗi mẫu. Mặc định là 1 đối với mẫu có một khung hìnhvà có thể có giá trị lớn hơn 1 đối với mẫu có nhiều khung hình.

*Compressorname* là một tên dùng cho mục đích cung cấp thông tin. Nó được định dạng trong một trường 32 byte cố định, với byte đầu tiên chỉ thị số byte được hiển thị, tiếp theo là số byte dữ liệu có khả năng hiển thị, sau đó thêm các byte đệm để có tổng cộng 32 byte (bao gồm cả kích thước byte). Trường này có thể nhận giá trị 0.

*depth*thực hiện một trong các giá trị sau

0x0018- các hình ảnh có màu sắc không có mẫu ký tự.

*width and height*là độ rộng và chiều cao cực đại của hình ảnh của luồng được miêu tả bởi miêu tả mẫu theo số điểm ảnh.

*hSpacing, vSpacing*: xác định mối liên hệ giữa độ rộng và chiều cao của một điểm ảnh.

*cleanApertureWidthN, cleanApertureWidthD*là phân số xác định cơ bản xác độ rộng khẩu độ tính theo số điểm ảnh của hình ảnh.

*cleanApertureHeightN,cleanApertureHeightD*:là phân số xác định cơ bản xác chiều cao khẩu độ theo số điểm ảnh của hình ảnh.

*horizOffN, horizOffD:* là phân số xác định độ lệnh phương ngang của trung tâm khẩu rõ trừ đi (*width*-1)/2, thường là 0.

*vertOffN, vertOffD*: là phân số xác định độ lệnh theo phương thẳng đứng của trung tâm khẩu rõ trừ đi (*height*-1)/2, thường là 0.

*content\_encoding* là một chuỗi các ký tự UTF-8 kết thúc bởi ký tự trống (null) và cung cấp một kiểu MIME xác định mã hóa nội dung của siêu dữ liệu định thời. Nó được xác định giống như đối với IteminfoEntry trong tiêu chuẩn này. Nếu không xuất hiện (một chuỗi rỗng được cung cấp) các siêu dữ liệu định thời sẽ không được mã hóa. Một ví dụ cho trường này là ‘ứng dụng/zip’. Lưu ý rằng hiện không tồn tại các kiểu MIME của BiM[ISO/IEC 23001-1] và TeM [ISO/IEC15938-1]. Vì vậy các kiểu MIME thử nghiệm ‘ứng dụng/x-BiM’ và ‘văn bản/x-TeM’ cần được sử dụng để nhận dạng các cơ chế mã hóa này.

*namespace* là một trường được kết thúc bởi một kí tự trống, bao gồm một danh sách được biểu diễn bởi các kí tự UTF-8 và phân tách nhau bởi khoảng trống của một hoặc nhiều không gian tên XML mà văn bản mẫu phải tuân theo. Khi sử dụng cho siêu dữ liệu, trường này cần thiết để nhận dạng kiểu của siêu dữ liệu, ví dụ kiểu gBSD hoặc AQoS [MPEG-21-7] và để giải mã sử dụng các cơ chế mã hóa nhận biết XML như BiM.

*schema\_location* là một trường tùy chọn kết thúc bởi kí tự trống, bao gồm một danh sách có các thành phần được biểu diễn bởi các kí tự UTF-8 và phân tách nhau bởi khoảng trống, chứa các URL cho các sơ đồ XML mà văn bản mẫu phải tuân theo. Nếu tồn tại một *namespace* và một lược đồ thì trường này sẽ là URL của lược đồ. Nếu có nhiều hơn một *namespace* thì cú pháp của trường này sẽ tuân thủ theo thuộc tính xsi:schemaLocation được định nghĩa bởi [XML].Khi sử dụng cho siêu dữ liệu, trường này cần thiết để giải mã siêu dữ liệu định thời sử dụng các cơ chế mã hóa nhận biết XML như BiM.

*mime\_format* cung cấp một kiểu MIME, biểu diễn bởi các kí tự UTF-8, kết thúc bởi ký tự trống, kiểu MIME này xác định định dạng nội dung của các mẫu. Ví dụ cho trường này bao gồm cả ‘text/html’ và ‘text/plain’.

*bufferSizeDB* cho biết kích thước của bộ đệm giải mã đối với luồng cơ sở tính theo byte.

*maxBitrate* cho biết tốc độ bit lớn nhất tính theo đơn vị bit/s đối với cửa sổ một giây bất kỳ.

*avgBitrate* cho biết tốc bit trung bình, tính theo đơn vị bit/s đối với toàn bộ bản trình diễn.

*theURI* là định dạng URI theo các quy tắc trong 6.2.4;

*uri\_initialization\_data* là một dữ liệu không rõ ràng mà hình thức được định nghĩa trong các tài liệu về dạng URI.

*colour\_type*: cho biết các kiểu thông tin màu. Đối *colour\_type* ‘nclx’: các trường này cơ bản xác bốn byte định nghĩa cho PTM\_COLOR\_INFO() trong A.7.2 của ISO/IEC 29199-2 nhưng lưu ý rằng toàn bộ cờ phạm vi đầy đủ ở một vị trí bit khác.

*ICC\_profile*: thông tin ICC được xác định trong ISO 15076-1 hoặc ICC.1:2010.

*auxiliary\_mime\_types* cho biết kiểu đa phương tiện của tất cả các tài nguyên phụ trợ, như các hình ảnh và các phông chữ, nếu xuất hiện, được lưu trữ như các mẫu nhỏ phụ đề. Nếu có nhiều hơn một *mime\_type*, khi đó trường này sẽ là một danh sách không gian được phân tách. Trường này biểu diễn bởi các ký tự UTF-8, kết thúc bởi ký tự trống.

Tất các các chuỗi sẽ kết thúc bởi giá trị trống, ngay cả khi không được sử dụng. Giá trị “tùy chọn “có nghĩa là có ít nhất một byte trống.

Cả *namespace* và *schema\_location* được sử dụng để nhận biết nội dung văn bản XML và để khai báo “nhãn “hoặc thông tin thích hợp. Nhiều định danh *namespace*thể hiệnrãnh phù hợp với đặc điểm kỹ thuật được biểu diễn bởi mỗi định danh, một số có thể nhận dạng được nhiều đặc điểm hiện tại. Một bộ giải mã cần có khả năng giải mã tất cả các *namespace* để có thể giải mã và hiển thị cơ bản xác đa phương tiện liên kết với các đầu vào mục mẫu.

LƯU Ý: ngoài ra, các định danh *namespace* có thể biểu các ràng buộc về năng lực, như giới hạn về kích thước tài liệu, kích thước phông, tốc độ vẽ,…cũng như các ràng buộc về cú pháp như tính năng không cho phép hoặc bị kiểu bỏ.

### Hộp ưu tiên suy giảm (Degradation Priority Box)

#### Định nghĩa

Kiểu hộp : ‘*stdp*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này chứa độ suy giảm ưu tiên của mỗi mẫu. Giá trị này được lưu trong bảng, mỗi hộp ứng với mỗi mẫu. Kích thước của bảng, *sample\_count*được lấy từ sample\_count trong Hộp Kích thức Mẫu (‘*stsz*’). Các đặc tả kỹ thuật suy ra từ hộp này xác định nghĩacơ bản xác và giới hạn có thể được chấp nhận của trường ưu tiên.

#### Cú pháp

*aligned(8) class DegradationPriorityBox extends FullBox(‘stdp’, version = 0, 0) {*

*int i;*

*for (i=0; i <sample\_count; i++) {*

*unsigned int(16)priority;*

*}*

*}*

#### Ngữ nghĩa

*version* có kiểu số nguyên chỉ rõ phiên bản của hộp.

*priority*có kiểu số nguyên cho biết mức độ giảm tính ưu tiên của mỗi mẫu.

### Hộp tần số lấy mẫu (Sample Scale Box)

Hiện phần này chưa được xây dựng.

## Cấu trúc thời gian của rãnh

### Hộp Thời gian Mẫu (Time to Sample Boxs)

#### Định nghĩa

Thời gian tổng hợp (Composition Time - CT) và thời gian giải mã (Decoding Time - DT) của các mẫu được chỉ ra trong Hộp Thời Gian Mẫu, hộp này có hai kiểu khác nhau. Thời gian giải mã được xác định từ Hộp Thời Gian Giải Mã Mẫu, cho biết các khoảng thời gian delta giữa các lần giải mã kế tiếp. Thời gian tổng hợp được xác định từ Hộp Thời Gian MẫuTổng Hợp là các khoảng thời gian trênh lệch so với thời gian giải mã. Nếu thời gian tổng hợp và thời gian giải mã giống nhau cho mọi mẫu trong rãnh, thì chỉ cần có Hộp Thời Gian Giải Mã Mẫu; Hộp Thời Gian MẫuTổng Hợp là không cần thiết.

Các Hộp Thời Gian Mẫu phải cung cấp khoảng thời gian khác không cho tất cả các mẫu, ngoại trừ có thể có ngoại lệ với mẫu cuối. Khoảng thời gian trong khối ‘*stts’* là các số dương khác không, ngoại trừ đối với đầu vào mục cuối cùng có thể là không. Quy tắc này được suy ra từ quy tắc không có hai nhãn thời gian giống nhau trong một luồng. Phải thận trọng khi thực hiện bổ sung các mẫu vào một luồng, sao cho mẫu kết thúc trước đó phải có khoảng thời gian khác không để tuân theo quy tắc này. Nếu không xác định được khoảng thời gian của mẫu cuối cùng thì sử dụng một giá trị nhỏ tùy ý và một hiệu chỉnh ‘dwell’.

Trong ví dụ sau đây, có một chuỗi các khung hình I, P và B, mỗi khung hình có thời gian giải mã delta là 10. Các mẫu được lưu trữ như sau, với các giá trị được chỉ định cho thời gian giải mã delta và độ dịch chuyển thời gian tổng hợp (thực tế CT và DT được đưa ra để tham chiếu). Xảy ra việc sắp xếp lại bởi vì các khung hình P đã được dự đoán phải được giải mã trước các khung B dự đoán hai chiều. Giá trị của DT của một mẫu luôn bẳng tổng của các delta của các mẫu trước đó. Lưu ý rằng tổng tất cả của các delta giải mã là khoảng thời gian của đa phương tiệntrong rãnh này.

Bảng 2 – Ví dụ về GOP đóng

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **GOP** | /-- | -- | -- | -- | -- | -- | --\ | /-- | -- | -- | -- | -- | -- | --\ |
|  | I1 | P4 | B2 | B3 | P7 | B5 | B6 | I8 | P11 | B9 | B10 | P14 | B12 | B13 |
| **DT** | 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 | 110 | 120 | 130 |
| **CT** | 10 | 40 | 20 | 30 | 70 | 50 | 60 | 80 | 110 | 90 | 100 | 140 | 120 | 130 |
| **Delta**  **giải mã** | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| **Độ dịch chuyển tổng** | 10 | 30 | 0 | 0 | 30 | 0 | 0 | 10 | 30 | 0 | 0 | 30 | 0 | 0 |

Bảng 3 – Ví dụ về GOP mở

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **GOP** | /-- | -- | -- | -- | -- | --\ | /- | -- | -- | -- | --- | --\ |
|  | I3 | B1 | B2 | P6 | B4 | B5 | I9 | B7 | B8 | P12 | B10 | B11 |
| **DT** | 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 | 110 |
| **CT** | 30 | 10 | 20 | 60 | 40 | 50 | 90 | 70 | 80 | 120 | 100 | 110 |
| **Delta**  **giải mã** | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| **Độ dịch chuyển tổng** | 30 | 0 | 0 | 30 | 0 | 0 | 30 | 0 | 0 | 30 | 0 | 0 |

#### Hộp Thời gian Giải mãMẫu (Decoding Time to Sample Box)

##### Định nghĩa

Loại hộp: ‘*stts*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này bao gồm một phiên bản rút gọn của một bảng cho phép đánh chỉ số từ thời gian giải mã tới số lượng mẫu. Các bảng khác chứa kích thước mẫu và các con trỏ được đánh số từ số lượng mẫu. Mỗi đầu vào mục của bảng cho biết số lượng của các mẫu liên tiếp với delta thời gian giống nhau và delta của các mẫu này. Bằng cách thêm vào các delta một ánh xạ đầy đủ thời gian và mẫu có thể được hình thành.

Hộp Ánh xạ Thời gian mã hóa và mẫu chứa thời gian mã hóa delta: DT(n+1) =DT(n) + *STTS*(n) trong đó*STTS*(n) là (không bị nén) bảng đầu vào mục cho mẫu n.

Các đầu vào mục mẫuđược sắp xếp theo các nhãn thời gian giải mã; do vậy delta không âm.

Trục DT có gốc tọa độ không: DT(i) = SUM(từ j=0 đến i-1 của delta(j)) và tổng của tất cả các delta cho biết độ dài của đa phương tiện trong rãnh (không được ánh xạ tới toàn bộ phạm vi thời gian và không được xem xét bất kỳ danh sách biên tập).

Hộp Danh sách Biên tậpcho biết giá trị CT khởi tạo nếu nó khác rỗng (khác không).

##### Cú pháp

*aligned(8) class TimeToSampleBox extends FullBox(’stts’, version = 0, 0) {*

*unsigned int(32)entry\_count;*

*int i;*

*for (i=0; i < entry\_count; i++) {*

*unsigned int(32)sample\_count;*

*unsigned int(32)sample\_delta;*

*}*

*}*

Ví dụ với bảng 2, đầu vào mục có thể là:

|  |  |
| --- | --- |
| ***sample\_count*** | ***sample\_delta*** |
| 14 | 10 |

##### Ngữ nghĩa

*version* là số nguyên được chỉ thị phiên bản của hộp.

*entry\_count* là số nguyên cho biết số lượng đầu vào mục của bảng.

*sample\_count* là số nguyên chỉ số lượng các mẫu liên tiếp có cùng khoảng thời gian mẫu.

*sample\_delta* là số nguyên cho biết khoảng cách thời gian delta giữa các mẫu theo thang thời gian của đa phương tiện.

#### Hộp Thời gian Tổng hợp Mẫu (Composition Time to Sample Box)

##### Định nghĩa

Loại hộp: ‘*ctts*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này cho biết độ dịch chuyển giữa thời gian giải mã và thời gian tổng hợp. Trong phiên bản 0 của hộp này thời giải mã phải ít hơn thời gian tổng hợp và các độ dịch chuyển này được thể hiện như các số không dấu, đó là CT(n) = DT(n) + *CTTS*(n) trong đó*CTTS*(n) (không bị nén) là bảng đầu vào mục cho mẫu n. Trong phiên bản 1 của hộp này, từ mốc thời gian tổng hợpcó thể suy ra mốc thời gian giải mã và ngược lại, độ dịch chuyển là sô có dấu. Khuyến nghị đối với các nhãn thời gian tổng hợp được tính là chì có duy một nhãn có giá trị 0.

Đối với hai phiên bản của hộp, mỗi mẫu có duy nhất một giá trị nhãn thời gian tổng hợp, có nghĩa là nhãn thời gian cho hai mẫu sẽ không bao giờ giống nhau.

Có thể không có khung hình nào được tổng hợp tại thời điểm 0; việc xử lý này không xác định (hệ thống có thể hiển thị khung hình đầu tiên trong thời gian dài hơn, hoặc hiển thị đầy màu sắc phù hợp).

Khi phiên bản 1 của hộp này được sử dụng, CompositionToDecodeBox có thể cũng xuất hiện trong bảng mẫu liên quan tới các mốc thời gian tổng hợp và mốc thời giải mã. Khi khả năng tương thích ngược hoặc khả năng tương thích với tập hợp các bộ đọc không xác định trước được yêu cầu thìsử dụng phiên bản 0 của hộp này khi có thể. Trong cả hai phiên bản của hộp, nhưng đặc biệt trong phiên bản 0, nếu muốn đa phương tiện bắt đầutại rãnh thời gian 0 và mẫu đa phương tiện đầu tiên không có thời gian tổng hợpbằng 0 thì một danh sách biên tập có thể được sử dụng để ‘di chuyển’ đa phương tiện đóvề thời điểm 0.

Bảng ánh xạ thời gian tổng hợp và mẫu là tùy chọn và chỉ xuất hiện nếu DT và CT khác nhau đối với mọi mẫu.

Các rãnh chỉ dẫn không được sử dụng hộp này.

Ví dụ trong bảng 2

|  |  |
| --- | --- |
| **samle\_count** | **sample\_offset** |
| 1 | 10 |
| 1 | 30 |
| 2 | 0 |
| 1 | 30 |
| 2 | 0 |
| 1 | 10 |
| 1 | 30 |
| 2 | 0 |
| 1 | 30 |
| 2 | 0 |

##### Cú pháp

*aligned(8) class CompositionOffsetBox extends FullBox(‘ctts’, version = 0, 0) {*

*unsigned int(32)entry\_count;*

*int i;*

*if (version==0) {*

*for (i=0; i < entry\_count; i++) {*

*unsigned int(32)sample\_count;*

*unsigned int(32)sample\_offset;*

*}*

*}*

*else if (version == 1) {*

*for (i=0; i < entry\_count; i++) {*

*unsigned int(32)sample\_count;*

*signed int(32)sample\_offset;*

*}*

*}*

*}*

##### Ngữ nghĩa

*version* là số nguyên chỉ phiên bản của hộp.

*entry\_count*  là số nguyên cho biết số lượng đầu vào mục.

*sample\_count* là số nguyên cho biết số lượng các mẫu liên tiếp có độ dịch chuyển cho trước.

*sample\_offset* là số nguyên cho biết độ dịch chuyển giữa CT và DT, như vậy CT(n) = DT(n) + *CTTS*(n).

#### Hộp Tổng hợp và Giải mã (Composition to Decode Box)

##### Định nghĩa

Loại hộp: ‘*cslg*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’) hoặc Hộp Thuộc tính Mở rộng rãnh (‘trep’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Khi sử dụng các độ dịch chuyển tổng hợp có dấu, hộp này có thể được sử dụng để tạo mối liên hệ giữa các mốc thời gian tổng hợp và mốc thời gian giải mã, khắc phục một sốthời điểm không rõ ràng do các độ dịch chuyển tổng hợp có dấu tạo ra.

Lưu ý rằng tất cả các trường này áp dụng cho toàn bộ đa phương tiện (không chỉ là thành phần được lựa chọn bởi các hiệu chỉnh). Khuyến nghị rằng mọi hiệu chỉnh, tường minh hay ngầm định, không lựa chọn bất kỳ phần nào của mốc thời gian tổng hợp không ánh xạ tới một mẫu. Ví dụ, nếu thời gian tổng hợp nhỏ nhất là 1000, khi đó hiệu chỉnh mặc định từ 0 tới khoảng thời gian đa phương tiện còn lại là khoảng thời gian từ 0 tới 1000 không được liên kết với mẫu đa phương tiện nào. Hành vi của thiết bị đọc và đối tượng được tổng hợp trong khoảng thời này không xác định được trong các trường hợp đó. Khuyến nghị gán giá tính toán nhỏ nhất CTS bằng không, hoặc bằng giá trị đầu của hiệu chỉnh đầu tiên.

Khoảng thời gian tổng hợp của mẫu cuối cùng trong một rãnh có thể (thường là) không xác định hoặc không rõ ràng; trường dành cho thời gian tổng hợp cuối cùng có thể được sử dụng để làm rõ sự không xác định vừa nêuvà thiết lập khoảng thời gian tổng hợpcơ bản xác cho rãnh khi biết thời gian bắt đầu tổng hợp.

Khi Hộp Tổng hợp và Giải mãnằm trong Hộp Bảng Mẫu thì nó sẽ lưu thời gian tổng hợp và thời gian mã hóa liên quan đến các mẫu nằm trong Hộp Movie, không bao gồm các phân mảnh movie kế tiếp.

Khi Hộp Tổng hợp và Giải mã nằm trong Hộp Thuộc tính Mở rộng Rãnh thì nó sẽ lưu thời gian tổng hợp và thời gian giải mã liên quan đến các mẫu trong tất cả các phân mảnh movie trong Hộp Movie.

##### Cú pháp

*class CompositionToDecodeBox extends FullBox(‘cslg’, version=0, 0) {*

*signed int(32) compositionToDTSShift;*

*signed int(32) leastDecodeToDisplayDelta;*

*signed int(32) greatestDecodeToDisplayDelta;*

*signed int(32) compositionStartTime;*

*signed int(32) compositionEndTime;*

*}*

##### Ngữ nghĩa

*compositionToDTSShift:* nếu giá trị này được thêm vào các thời gian tổng hợp (như được tính toán bởi các độ dịch chuyển CTS từ DTS), sau đó cho tất cả các mẫu, CTS của chúng được đảm bảo lớn hơn hoặc bằng DTS của chúng và mô hình bộ đệm được chỉ định thông quathông tin/mức độ chỉ thị được sử dụng; nếu *leastDecodeToDisplayDelta*dương hoặc bằng 0 thì trường này có thể bằng 0; nếu không ít nhất nó bằng (-leastDecodeToDisplayDelta).

*leastDecodeToDisplayDelta*: độ dịch chuyểntổng hợp nhỏ nhấttrong hộp CompositionTimeToSample của rãnh đó.

*greatestDecodeToDisplayDelta:*độ dịch chuyển tổng hợp lớn nhất trong hộp CompositionTimeToSample của rãnh đó.

*compositionStartTime:* thời gian tổng hợp nhỏ nhất (CTS) cho mọi mẫu đa phương tiện của rãnh đó.

*compositionEndTime:* thời gian tổng hợp cộng với khoảng thời gian tổng hợp của mẫu với thời gian tổng hợp lớn nhất (CTS) trong đa phương tiện của rãnh đó; nếu trường này có giá trị 0, thời gian tổng hợp cuối là không xác định.

### Hộp Mẫu Đồng bộ (Sync Sample Box)

#### Định nghĩa

Loại hộp: ‘*stss*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’)

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này cung cấp một đánh dấu rút gọn của các mẫu đồng bộ trong luồng. Bảng được sắp xếp theo đúng thứ tự tăng dần của số lượng mẫu.

Nếu hộpmẫu đồng bộ không xuất hiện, mỗi mẫu sẽ là một mẫu đồng bộ.

#### Cú pháp

*aligned(8) class SyncSampleBox extends FullBox(‘stss’, version = 0, 0) {*

*unsigned int(32)entry\_count;*

*int i;*

*for (i=0; i < entry\_count; i++) {*

*unsigned int(32)sample\_number;*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*entry\_count* có kiểu số nguyêncho biết số lượng các đầu vào mục trong bảng sau. Nếu *entry\_count* bằng không có nghĩa là không có các mẫu đồng bộ trong luồng và bảng sau là trống.

*sample\_number* cho biết số lượng của các mẫu đồng bộ trong luồng.

### Hộp Mẫu Đồng bộ Vùng tối (Shadow Sync Sample Box)

#### Định nghĩa

Loại hộp: ‘*stsh*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Bảng đồng bộ vùng tối cung cấp một tập hợp các tùy chọn của các mẫu đồng bộ có thể được sử dụng khi tìm kiếm hoặc sử dụng cho các mục đích tương tự. Khi trình diễn thông thường chúng được bỏ qua.

Mỗi mục trong ShadowSyncTable bao gồm một cặp số mẫu. Đầu vào mục đầu tiên (*shadowed\_sample\_number*) cho biết số lượng mẫu mà một đồng bộ vùng tối sẽ được xác định. Mẫu đó luôn luôn là một mẫu không đồng bộ (ví dụ như một khác biệt khung hình). Số mẫu thứ hai (*sync-sample-number*) cho biết số lượng mẫu của mẫu đồng bộ (ví dụ khung hình cơ bản) có thể được sử dụng khi cần một mẫu đồng bộ tại*shadowed-sample-number*, hoặc trước*shadowed-sample-number*.

Các đầu vào mục trong ShadowSyncBox sẽ được lưu trữ dựa trên trường *shadowed\_sample\_number*.

Các mẫu đồng bộ vùng tối thường được đặt trong vùng của rãnh mà rãnh đó không được xuất hiện trong khi phát bình thường (đầu ra được chỉnh sửa theo cách thức của một danh sách biên tập), mặc dù đây không phải là một yêu cầu. Bảng đồng bộ vùng tối có thể được bỏ qua và rãnh sẽ vẫn phát (và tìm kiếm) cơ bản xác nếu nó được bỏ qua (mặc dù có lẽ không tối ưu).

ShadowSyncSample thay thế, chứ không phải bổ sung, các mẫu vùng tối (ví dụ mẫu tiếp được gửi là *shadowed\_sample\_number*+1). Mẫu đồng bộ vùng tối được xử lý như là nó xuất hiện tại thời điểm của mẫu mà nó che lấp, với khoảng thời gian của mẫu mà nó che lấp.

Việc chỉ dẫn và truyền dẫn có thể trở nên phức tạp hơn nếu một mẫu vùng tối cũng được sử dụng như thành phần của phát lại thông thường, hoặc được sử dụng nhiều hơn một lần. Trong trường hợp rãnh chỉ dẫncầntách riêng các đồng bộ vùng tối thì tất cả có thể nhận dữ liệu đa phương tiện của chúng từ một đồng bộ vùng tối trong rãnh đa phương tiện với các nhãn thời gian khác nhau ví dụ như trong các mào đầu.

#### Cú pháp

*aligned(8) class ShadowSyncSampleBox extends FullBox(‘stsh’, version = 0, 0) {*

*unsigned int(32)entry\_count;*

*int i;*

*for (i=0; i < entry\_count; i++) {*

*unsigned int(32)shadowed\_sample\_number;*

*unsigned int(32)sync\_sample\_number;*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*entry\_count* có kiểu số nguyên cho biết số lượng của các đầu vào mục.

*shadowed\_sample\_number* cho biết số lượng của một mẫu mà trong đó có một mẫu đồng bộ thay thế.

*sync\_sample\_number* cho biết số lượng của mẫu đồng bộ thay thế.

### Hộp các Mẫu Độc lập và Dùng một lần (Independent and Disposable Samples Box)

#### Định nghĩa

Loại hộp: ‘*sdtp*’.

Hộpchứa: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Bảng tùy chọn này trả lời 3 câu hỏi liên quan đến sự phụ thuộc của mẫu:

1. Mẫu này có phụ thuộc vào các mẫu khác nhau (ví dụ. đó là một ảnh –I) hay không?
2. Không có các mẫu khác nào phụ thuộc vào mẫu này?
3. Có phải mẫu này bao gồm nhiều (độ dư) kiểu mã hóa của dữ liệu tại cùng thời điểm (có thểvới nhiều sự phụ thuộc khác nhau)?

Trong trường hợp không có bảng này:

1. Bảng mẫu đồng bộ (một phần) trả lời câu hỏi đầu tiên; trong nhiều bộ mã hóa giả mã hình ảnh, các ảnh –Icơ bản là các điểm đồng bộ.
2. Sự phụ thuộc của các mẫu khác nhau vào mẫu này chưa xác định được.
3. Sự tồn tại của mã hóa dư chưa xác định được.

Khi thực hiện chế độ ‘trick’, ví dụ như tua nhanh (fast-forward),có thể sử dụng phần đầu tiên của thông tin để định vị các mẫu có thể giải mã độc lập. Tương tự như vậy, khi thực hiện truy nhập ngẫu nhiên,có thể cần phải xác định vị trímẫu đồng bộ trước đó hoặc điểm khôi phục truy nhập ngẫu nhiên và việc tua đi từ mẫu đồng bộ hoặc điểm bắt đầu pre-roll của điểm khôi phục truy nhập ngẫu nhiên tới điểm mong muốn. Trong khi đang tua đi, các mẫu không phụ thuộc không cần thiết phải ghi lại hoặc giải mã.

Giá trị của ‘*sample\_is\_depended\_on*’ là độc lập với sự tồn tại của mã hóadư. Tuy nhiên, mã hóadư có thể phụ thuộc khác nhau đối với mã hóa cơ bản,nếu các mã hóadư là khả dụng, giá trị của ‘*sample\_depends\_on’*chỉ lưu mã hóa cơ bản.

Một mẫu đầu tiên (thường là một ảnh trong chuỗi hình ảnh) xác định mối liên hệ với mẫu tham chiếu, đó là mẫu ưu tiên trực tiếp được đánh dấu như “*sample\_depends\_on*“không phụ thuộc (một ảnh I). Một mẫu đầu có cả thời gian tổng hợp trước mẫu tham chiếu và cũng có thể phụ thuộc việc giải mã trên một mẫu trước mẫu tham chiếu. Do vậy, ví dụ, nếu việc phát lại và giải mã bắt đầu tại mẫu tham chiếu, cácmẫu này được đánh dấu như mẫu đầu tiên vì vậy sẽ không còn cần thiết và có thể không được giải mã. Do đó, mẫu đứng đầu không được đánh dấu không phụ thuộc.

Đối với các rãnh có *handler\_type* không phải là ‘vide’, ‘soun’, ‘hint’ hoặc ‘auxv’, nếu một mẫu khác có sample\_depends\_on=2hoặc được đánh dấunhư một “mẫu đồng bộ“đã được xử lý thì trừ khi có các quy định khác, không thì một mẫu có sample\_depends\_on=2 và sample\_has\_redundancy=1 có thể bị kiểu bỏ và khoảng thời gian của nó được cộng thêm một khoảng thời gian trước đó để duy trì thời gian của các mẫu tiếp theo.

Kích thước của bảng, *sample\_count*, được lấy từ giá trị *sample\_count* trong Hộp Kích thướcMẫu hoặc Hộp Kích thước Mẫu rút gọn (‘*stz2*’).

#### Cú pháp

*aligned(8) class SampleDependencyTypeBox extends FullBox(‘sdtp’, version = 0, 0) {*

*for (i=0; i <sample\_count; i++){*

*unsigned int(2) is\_leading;*

*unsigned int(2) sample\_depends\_on;*

*unsigned int(2) sample\_is\_depended\_on;*

*unsigned int(2) sample\_has\_redundancy;*

*}*

*}*

#### Ngữ nghĩa

*is\_leading* nhận một trong bốn giá trị sau:

0: đặc tính đầu của mẫu này chưa xác định được;

1:mẫu này là mẫu đứng đầu, đó là mẫu có sự phụ thuộc trước ảnh -I được tham chiếu (và do đó không thể giải mã được);

2: mẫu này không phải là một mẫu đứng đầu;

3: mẫu này là một mẫu đứng đầu không có sự phụ thuộc trước ảnh –I được tham chiếu (và do đó có thể giải mã được);

*sample\_depends\_on*nhận một trong bốn giá trị sau;

0: sự phụ thuộc của mẫu này chưa được xác định;

1:mẫu này phụ thuộc vào mẫu khác (không phải là ảnh - I);

2:mẫu này không phụ thuộc vào mẫu khác (ảnh - I);

3: được dành riêng.

*sample\_is\_depended\_on*nhận một trong bốn giá trị sau:

0: sự phụ thuộc của các mẫu khác đối với mẫu này chưa được xác định;

1:các mẫu khác có thể phụ thuộc vào mẫu này (không sẵn có);

2: không có các mẫu khác nhau phụ thuộc vào một mẫu này (sẵn có);

3: được dành riêng.

*sample\_has\_redundancy*nhận một trong bốn giấ trị sau:

0: nó không biết liệu có mã hóa dư trong mẫu này hay không;

1: cómã hóa dư trong mẫu này;

2:không có mã hóa dư trong mẫu này;

3:được dành riêng.

### Hộp Biên tập (Edit Box)

#### Định nghĩa

Loại hộp: ‘*edts*’

Thuộc đối tượng: Hộp Rãnh (‘*trak*’)

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Biên tập ánh xạ mốc thời gian bản trình diễnvới mốc thời gian đa phương tiện lưu trữ trong tệp tin. Hộp Biên tập là hộp chứa các danh sách biên tập.

Hộp Biên tập là tùy trọn. Trong trường hợp không có hộp này, sẽ tồn tại một ánh xạ một - một ẩn của các mốc thời gian này,bản trình diễncủa một rãnhsẽ bắt đầu tại điểm đầu của bản trình diễn. Một hiệu chỉnh rỗng được sử dụng để dịch chuyển thời gian bắt đầu của một rãnh.

#### Cú pháp

*aligned(8) class EditBox extends Box(‘edts’) {*

*}*

### Hộp Danh sách Biên tập(Edit List Box)

#### Định nghĩa

Loại hộp: ‘*elst*’

Thuộc đối tượng: hộp hiệu chỉnh(‘*edts*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này chứa một ánh xạ mốc thời gian rõ ràng. Mỗi đầu vào mục xác định thành phần mốc thời gian của rãnh: bằng cách ánh xạ phần mốc thời gian đa phương tiện, hoặc bằng chỉ thị thời gian ‘rỗng’, hoặc bằng cách xác định một ‘dwell’, tại đó một điểm thời gian duy nhất trong đa phương tiện được lưu giữ trong một khoảng thời gian.

LƯU Ý: các hiệu chỉnh là không bị giới hạn đối với thời gian mẫu. Điều đó nghĩa là khi thực hiện hiệu chỉnh, có thể cần (a) sao lưu một điểm đồng bộ, pre-roll từ đó và sau đó (b) cần cẩn trọng với khoảng thời gian của mẫu đầu tiên – nó có thể có bị cắt nếu hiệu chỉnh trong khoảng thời gian của nó. Nếu đây là âm thanh, khung này có thể cần được giải mã và sau đó cuối cùng thực hiện tách lớp.Tương tự như vậy, khoảng thời gian của mẫu cuối cùng trong hiệu chỉnh có thể cần tách lớp.

Độ chênh lệch ban đầu của các rãnh (các luồng) được biểu diễn bởi một hiệu chỉnh rỗng khởi tạo. Ví dụ, để phát một rãnh từ đầu trong 30 giây, nhưng tại giây thứ 10 củabản trình diễn, chúng có một sanh sách hiệu chỉnh sau đây:

Entry-count = 2

Segment-duration = 10 giây

Media-Time = -1

Media-Rate = 1

Segment-duration = 30 giây (có thể là toàn bộ độ dài của rãnh)

Media-Time = 0 giây

Media-Rate = 1

Một hiệu chỉnh khác rỗng có thể được đưa vào một phần của mốc thời gian đa phương tiện mà không xuất hiện trong khởi tạo movie và chỉ xuất hiện trong các phân đoạn movie tiếp theo. Đặc biệt trong movie khởi tạo rỗng của tệp tin movie phân mảnh (khi không có các mẫu xuất hiện), *segment\_duration* của hiệu chỉnh này có thể bằng không, sau đó hiệu chỉnh này cung cấp độ chệnh lệch giữa thời gian tổng hợpđa phương tiệnvà thời gian trình diễn movie, của movie đó và các phân đoạn movie kế tiếp. Một hiệu chỉnh nên được sử dụng để thiết lập một thời gian bản trình diễn0 đối với mẫu xuất hiện đầu tiên sử dụng các độ dịch chuyểntổng hợp.

Ví dụ, nếu thời gian tổng hợp của khung thứ nhất là 20, khi đó hiệu chỉnh ánh xạ thời gian đa phương tiện từ 20 trở về thời gian movie 0 trở đi, sẽ đọc:

Entry-count = 1

Segment-duration = 0

Media-Time = 20

Media-Rate = 1

#### Cú pháp

*aligned(8) class EditListBox extends FullBox(‘elst’, version, 0) {*

*unsigned int(32)entry\_count;*

*for (i=1; i <= entry\_count; i++) {*

*if (version==1) {*

*unsigned int(64) segment\_duration;*

*int(64) media\_time;*

*} else { // version==0*

*unsigned int(32) segment\_duration;*

*int(32)media\_time;*

*}*

*int(16) media\_rate\_integer;*

*int(16) media\_rate\_fraction = 0;*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên cho biết phiên bản của hộp (0 hoặc 1).

*entry\_count* có kiểu số nguyênchỉ số lượng đầu vào mục.

*segment\_duration*có kiểu số nguyên chỉ khoảng thời gian của một đoạn hiệu chỉnh theo đơn vị của thang thời gian trong Hộp Mào đầu Movie.

*media\_time*có kiểu số nguyên chứa thời điểm bắt đầu đoạn hiệu chỉnh trong đa phương tiện (theo đơn vị thang thời gian đa phương tiện, trong thời gian tổng hợp). Nếu trường này được thiết lập giá trị -1 thì đó là hiệu chỉnh rỗng. Hiệu chỉnh cuối trong một rãnh sẽ không bao giờ là hiệu chỉnh rỗng. Mọi sự khác biệt giữa khoảng thời gian trong Hộp Mào đầu Movie và khoảng thời gian của rãnh được biểu thị ngầm như một hiệu chỉnh rỗng ở cuối.

*media\_rate* cho biết tốc độ tương đối để phát đa phương tiện tương ứng với đoạn hiệu chỉnh. Nếu giá trị này bằng 0, khi đó hiệu chỉnh này là ‘dwell’: đa phương tiện tại *media\_time* đượcdùng cho khoảng thời gian của đoạn. Trong các trường hợp khác, trường sẽ có giá trị 1.

## Câu trúc sắp xếp dữ liệu của rãnh

### Hộp Thông tin Dữ liệu (Data Information Box)

#### Định nghĩa

Loại hộp:‘*dinf*’.

Thuộc đối tượng: Hộp Thông tin Đa phương tiện(‘*minf*’) hoặc Hộp Meta (‘*meta*’).

Hình thức sử dụng: - Bắt buộc (đối với hộp ‘*minf*’).

- Không bắt buộc (đối với hộp ‘*meta*’).

Số lượng: một.

Hộp Thông tin Dữ liệu bao gồm các đối tượng khai báo vị trí của thông tin đa phương tiện trong một rãnh.

#### Cú pháp

*aligned(8) class DataInformationBox extends Box(‘dinf’) {*

*}*

### Hộp Tham chiếu Dữ liệu (Data Reference Box)

#### Định nghĩa

Loại hộp:‘url ‘, ‘urn ‘, ‘*dref*’.

Thuộc đối tượng:Hộp Thông tin Dữ liệu (‘*dinf*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Đối tượng tham chiếu dữ liệu bao gồm một bảng các tham chiếu dữ liệu (thường là URL) khai báo vị trí của dữ liệu đa phương tiện được sử dụng trong bản trình diễn. Chỉ số tham chiếu dữ liệu trong miêu tả mẫu gắn các mục trong bảng tham chiếu với các mẫu trong rãnh.Một rãnh có thể được chia thành nhiều nguồn theo cách này.

Nếu cờ được thiết lập chỉ rằng dữ liệu nằm trong cùng tệp tin như hộp này thì không có chuỗi nào (kể cả chuỗi rỗng) được lưu trong trường đầu vào mục.

DataEntryBox trong DataReferenceBox sẽ là DataEntryUrnBox hoặc DataEntryUrlBox.

LƯU Ý: Mặc dù biến *count*là 32 bit nhưng số lượng bit thường ít hơn nhiều, trên thực tế chỉ số tham chiếu trong bảng mẫu bị hạn chế chỉ 16 bit.

Khi một tệp tin có dữ liệu chỉ mục có cờ được thiết lập nghĩa là dữ liệu đa phương tiện nằm trong cùng một tệp tin đó, dữ liệu được chia thành các phân đoạn để truyền, giá trị của cờ không thay đổi khi tệp tin nàyđược ghép lại sau khi truyền.

#### Cú pháp

*aligned(8) class DataEntryUrlBox (bit(24) flags)extends FullBox(‘url ’, version = 0, flags) {*

*string location;*

*}*

*aligned(8) class DataEntryUrnBox (bit(24) flags)extends FullBox(‘urn ’, version = 0, flags) {*

*string name;*

*string location;*

*}*

*aligned(8) class DataReferenceBoxextends FullBox(‘dref’, version = 0, 0) {*

*unsigned int(32) entry\_count;*

*for (i=1; i <= entry\_count; i++) {*

*DataEntryBox(entry\_version, entry\_flags) data\_entry;*

*}*

*}*

#### Ngữ nghĩa

*version* có kiểu số nguyên chỉ phiên bản của hộp.

*entry\_count* có kiểu số nguyên thể hiện số đầu vào mục thực tế.

*entry\_version*có kiểu số nguyên chỉ phiên bản của định dạng đầu vàomục.

*entry\_flags* là cờ kiểu số nguyên 24 bit, cờ có giá trị bằng x000001 có nghĩa là dữ liệu đa phương tiện nằm trong cùng một tệp tin giống nhưHộp Moviechứa cả dữ liệu tham chiếu.

*data\_entry* là URL hoặc URN. Tên là một URN và nằm trong đầu vào mục URN. Vị trí là một URL nằm trong đầu vào mục URL và tùy chọn đầu vào mục URN, trong đó từ vị trí có thể biết được tài nguyên với tên cho trước. Mỗi tên là là một chuỗi ký tự UTF-8 kết thúc bởi một chuỗi rỗng. Nếu cờ *self-contained* được thiết lập thì dạng URL được sử dụng và khi đó không có chuỗi nào tồn tại; hộp kết thúc bởi trường *entry\_flags*. Kiểu của URL xác định dịch vụ truyền tệp tin (ví dụ kiểu URL là http, ftp…)và các dịch vụ tốt cũng cho phép truy nhập ngẫu nhiên. Các URL tương đối được cấp quyền và liên quan với tệp tinchứa Hộp Movie có chứa dữ liệu tham chiếu.

### Các Hộp Kích thước Mẫu (Sample Size Boxes)

#### Định nghĩa

Loại hộp:‘*stsz*’, ‘*stz2*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này bao gồm số lượng mẫu và bảng tra kích thước theo đơn vị bytes của mỗi mẫu. Cơ chế này cho phép dữ liệu đa phương tiện tự tách khung Tổng số mẫu thường xuyên được cập nhập vào biến *sample count*.

Có hai dạng Hộp Kích thước Mẫu. Dạng thứ nhất có trường thể hệ kích thước mẫu là kích thược cố định là 32 bit; nó cho phép xác định kích thước của tất cả các mẫu trong rãnh. Dạng thứ hai là dạng có trường có kích thước nhỏ hơn, để tiết kiệm bộ nhớ khi kích thước ít thay đổi. Thực tế phải tồn tại hộp ở một trong 2 dạng nêu trên, tuy nhiên dạng thứ nhất được ưa chuộng hơn do khả năng mở rộng là lớn nhất.

LƯU Ý: kích thước mẫu nói chung không được phép bằng 0, nó phải có giá trị và được định nghĩa trong hệ thống mã hóa, được định nghĩa bởi đầu vào mục mẫu.

#### Hộp Kích thước Mẫu (Sample Size Box)

##### Cú pháp

*aligned(8) class SampleSizeBox extends FullBox(‘stsz’, version = 0, 0) {*

*unsigned int(32) sample\_size;*

*unsigned int(32) sample\_count;*

*if (sample\_size==0) {*

*for (i=1; i <= sample\_count; i++) {*

*unsigned int(32) entry\_size;*

*}*

*}*

*}*

##### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*sample\_size*có kiểu số nguyên chỉ kích thước mẫu mặc định. Nếu tất cả các mẫu có cùng kích thước, trường này sẽ chứa giá trị kích thước đó. Nếu trường này có giá trị 0, khi đó các mẫu có kích thước khác nhau và những kích thước này nằm trong bảng kích thước mẫu. Nếu trường này có giá trị khác 0 thì nó thể hiện hằng số kích thước mẫu và không cần mảng giá trị nào sau đó.

*sample\_count*có kiểu số nguyên chỉ số mẫu trong rãnh; nếu *sample-size*bằng 0 thì *sample\_count* cũng cơ bản là số mục có trong bảng mẫu.

*entry\_size*có kiểu số nguyên xác định kích thước của mẫu, được đánh số theo chỉ số của mẫu.

#### Hộp Kích thước Mẫu Rút gọn (Compact Sample Size Box)

##### Cú pháp

*aligned(8) class CompactSampleSizeBox extends FullBox(‘stz2’, version = 0, 0) {*

*unsigned int(24) reserved = 0;*

*unisgned int(8) field\_size;*

*unsigned int(32) sample\_count;*

*for (i=1; i <= sample\_count; i++) {*

*unsigned int(field\_size) entry\_size;*

*}*

*}*

##### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*field\_size*có kiểu số nguyên chỉ kích thước tính theo bit của các đầu vào mục trong bảng mẫu; nó sẽ nhận một trong các giá trị: 4, 8 hoặc 16. Nếu giá trị là 4, khi đó mỗi byte bao gồm 2 giá trị: entry[i]<<4 + entry[i+1]; nếu kích thước không chiếm đủ số nguyên của byte, byte cuối cùng sẽ chèn vào các bit 0.

*sample\_count*có kiểu số nguyên xác định số các mục bảng.

*entry\_size*có kiểu số nguyên xác định kích thước của mẫu, được đánh số theo chỉ số của mẫu.

### Hộp Ánh xạ Mẫu vàoĐoạn (Sample To Chunk Box)

##### Định nghĩa

Loại hộp:‘*stsc*’

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Các mẫu trong dữ liệu đa phương tiệnnhóm lại thành các đoạn. Các đoạn có thể có kích thước khác nhau và các mẫu trong một đoạn có thể có kích thước khác nhau. Bảng này có thể được dùng tìm đoạn có chứa một mẫu, vị trí của mẫu và miêu tả mẫu có liên quan.

Bảng được mã hóa ngắn gọn. Mỗi đầu vào mục cho biết chỉ số của đoạn đầu tiên của một nhóm các đoạn có cùng đặc tính. Có thể tính được tổng số đoạn đã chạy bằng cách trừ đi một sau khi biết tổng số đoạn từ đoạn trước đó. Cũng có thể tính tổng số mẫu bằng cách nhân số đoạn với tỉ lệ thích hợp số mẫu / 1 đoạn.

##### Cú pháp

*aligned(8) class SampleToChunkBoxextends FullBox(‘stsc’, version = 0, 0) {*

*unsigned int(32) entry\_count;*

*for (i=1; i <= entry\_count; i++) {*

*unsigned int(32) first\_chunk;*

*unsigned int(32) samples\_per\_chunk;*

*unsigned int(32) sample\_description\_index;*

*}*

*}*

##### Ngữ nghĩa

*version có* kiểu số nguyên chỉ phiên bản của hộp.

*entry\_count* có kiểu số nguyên xác định số các đầu vào mục.

*first\_chunk*có kiểu số nguyên thể hiện chỉ số của đoạn đầu tiên của nhóm các đoạn có cùng số mẫu / 1 đoạnvà *sample-description-index*, chỉ số của đoạn đầu tiên trong một rãnh có giá trị 1(trường *first\_trunk* trong bản ghi đầu tiên của hộp này có giá trị 1, thể hiện mẫu đầu tiên nằm trong đoạn đầu tiên).

*samples\_per\_chunk*có kiểu số nguyên xác định số mẫu trong mỗi đoạn.

*sample\_description\_index*có kiểu số nguyên xác định chỉ số của đầu vào mục mẫumiêu tả các mẫu trong đoạn này. Chỉ số nằm trong khoảng từ 1 đến số các đầu vào mục mẫu có trong Hộp Mô tả Mẫu.

### Hộp Độ dịch chuyển Đoạn (Chunk Offset Box)

#### Định nghĩa

Loại hộp:‘*stco*’, ‘*co64*’

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một tại một thời điểm.

Bảng độ dịch chuyển đoạnthể hiện chỉ số của mỗi đoạn trong tệp tin. Có hai dạng độ dịch chuyển khác nhaulà 32 bit hoặc 64 bit. Độ dịch chuyển 64 bit rất hữu dụng khi quản lý các bản trình diễn lớn. Ít nhất một trong số các dạng này sẽ tồn tại trong các biến đơn lẻ của bảng mẫu.

Độ dịch chuyển cơ bản là độ dịch chuyển của tệp tin, không phải là độ dịch chuyển của hộp trong tệp tin (ví dụ như Hộp Dữ liệu Đa phương tiện). Điều này cho phép tham chiếu đến dữ liệu đa phương tiện trong các tệp tin không có cấu trúc hộp. Cũng cần lưu ý khi xây dựng một tệp tin ISO độc lập với siêu dữ liệu (Hộp Movie) nằm ở phần đầu vì kích thước củaHộp Movie sẽ ảnh hưởng đến độ dịch chuyểnđoạn so với dữ liệu đa phương tiện.

#### Cú pháp

*aligned(8) class ChunkOffsetBoxextends FullBox(‘stco’, version = 0, 0) {*

*unsigned int(32) entry\_count;*

*for (i=1; i <= entry\_count; i++) {*

*unsigned int(32) chunk\_offset;*

*}*

*}*

*aligned(8) class ChunkLargeOffsetBoxextends FullBox(‘co64’, version = 0, 0) {*

*unsigned int(32) entry\_count;*

*for (i=1; i <= entry\_count; i++) {*

*unsigned int(64) chunk\_offset;*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp.

*entry\_count* có kiểu số nguyên xác định số đầu vào mục.

*chunk\_offset*có kiểu số nguyên 32 hoặc 64 bit cho biết độ lệch của phần đầu của một đoạn trong tệp tinđa phương tiện chứa nó.

### Hộp CácBit Đệm (Padding Bits Box)

#### Định nghĩa

Loại hộp:‘*padb*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Trong một số luồng, các mẫu đa phương tiện không chiếm tất cả các bit của các byte như đã được cung cấp bởi kích thước mẫu, và được chèn thêm các bit đệm ở cuối byte sau cùng. Trong một vài trường hợp, cần phải ghi lại số lượng các bit đệm đã sử dụng. Bảng này cung cấp thông tin đó.

#### Cú pháp

*aligned(8) class PaddingBitsBox extends FullBox(‘padb’, version = 0, 0) {*

*unsigned int(32) sample\_count;*

*int i;*

*for (i=0; i < ((sample\_count + 1)/2); i++) {*

*bit(1) reserved = 0;*

*bit(3) pad1;*

*bit(1) reserved = 0;*

*bit(3) pad2;*

*}*

*}*

#### Ngữ nghĩa

*sample\_count* – số lượng các mẫu có trong rãnh, nó phải trùng với số lượng trong các bảng khác.

*pad1* – có giá trị từ 0 đến 7, chỉ số bit tại phần cuối của mẫu (i\*2)+1.

*pad2* – có giá trị từ 0 đến 7, chỉ số bit tại phần cuối của mẫu (i\*2)+2.

### Hộp Thông tin Mẫu con (Sub-sample Information Box)

#### Định nghĩa

Loại hộp:‘*subs*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’) hoặc Hộp Phân mảnh Rãnh (‘*traf*’).

Tính bắt buộc: không bắt buộc.

Số lượng: 0 hoặc 1.

Hộp này có tên là Hộp Thông tin Mẫu con, được thiết kế để chứa thông tin mẫu con. Một mẫu con là một dải liên tục các byte của một mẫu. Một định nghĩa cụ thể về mẫu con được cung cấp cho một hệ thống mã hóa (ví dụ cho ISO / IEC 14496-10, Mã hóa hình ảnh tiên tiến). Trong trường hợp không có định nghĩa cụ thể như vậy, hộp này sẽ không được áp dụng cho các mẫu trong hệ thống mã hóa đó.

Nếu *subsample\_count* bằng 0 trong bất kỳ đầu vào mục nào thì những mẫu này sẽ không có thông tin mục con và không có mảng theo sau. Bảng này được mã hóa đơn giản; bảng thể hiện mẫu có cấu trúc mẫu con bằng cách ghi lại điểm khác biệt trong *sample-number*giữa mỗi đầu vào mục. Đầu vào mục đầu tiên của bảng ghi số mẫu của mẫu đầu tiên chứa thông tin mẫu con.

LƯU Ý: có thể kết hợp *subsample\_priority* và *discardable* khi *subsample\_priority* nhỏ hơn một giá trị nhất định và *discardable* nhận giá trị là 1. Tuy nhiên, vì các hệ thống khác nhau sử dụng các kiểu đo khác nhau cho các giá trị ưu tiên, để phân biệt chúng an toàn cần có một giải pháp rõ ràng để kiểu bỏ mẫu con.

#### Cú pháp

*aligned(8) class SubSampleInformationBoxextends FullBox(‘subs’, version, 0) {*

*unsigned int(32) entry\_count;*

*int i,j;*

*for (i=0; i < entry\_count; i++) {*

*unsigned int(32) sample\_delta;*

*unsigned int(16) subsample\_count;*

*if (subsample\_count> 0) {*

*for (j=0; j <subsample\_count; j++) {*

*if(version == 1){*

*unsigned int(32) subsample\_size;*

*}*

*else{*

*unsigned int(16) subsample\_size;*

*}*

*unsigned int(8) subsample\_priority;*

*unsigned int(8) discardable;*

*unsigned int(32) reserved = 0;*

*}*

*}*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp (0 hoặc 1 trong trường hợp này).

*entry\_count* có kiểu số nguyên xác định số đầu vào mục.

*sample\_delta*có kiểu số nguyên xác định số mẫu có cấu trúc mẫu con. Nó được mã bởi vì có sự khác biệt giữa số mẫu mong muôn và số mẫu được chỉ định trong đầu vào mục trước dó. Nếu đầu vào mục hiện tại là đầu vào mục đầu tiên thì giá trị thể hiện số mẫu của mẫu đầu tiên chứa thông tin mục con, nghĩa là giá trị đó thể hiện sự khác biệt giữa số mẫu và 0.

*subsample\_count*có kiểu số nguyên xác định số lượng của mẫu con có trong mẫu hiện tại. Nếu không có cấu trúc mẫu con nào thì giá trị này nhận giá trị 0.

*subsample\_size*có kiểu số nguyên xác định kích thước theo byte của mẫu con hiện tại.

*subsample\_priority*có kiểu số nguyên xác định ưu tiên suy giảm của mỗi mẫu con. Các giá trị *subsample\_priority*lớn hơn thể hiện các mẫu con quan trọng và có tác động lớn hơn đến chất lượng giải mã.

*discardable* bằng 0 có nghĩa là mẫu con cần có để giải mã mẫu hiện tại, trong khi giá trị bằng 1 có nghĩa là mẫu con không cần thiết với việc giải mã mẫu hiện tại nhưng có thể được sử dụng để nâng cao, ví dụ mẫu concó các bản tin thông tin nâng cao bổ sung (SEI).

### Hộp Kích cỡ Thông tin Bổ trợ Mẫu (Sample Auxiliary Information Sizes Box)

#### Định nghĩa

Loại hộp:‘*saiz*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’) hoặc Hộp Phân mảnh Rãnh ('*traf*').

Hình thức sử dụng: không bắt buộc.

Số lương: không hoặc nhiều.

Thông tin bổ trợ mẫu của mỗi mẫu có thể được lưu tại bất cứ đâu trong cùng tệp tinnhư là dữ liệu mẫu; đối với tệp tinđa phương tiện độc lập, điển hình là Hộp Dữ liệu Đa phương tiện hoặc hộp trong một chỉ tiêu kỹ thuật phái sinh. Thông tin này được lưu hoặc (a) là trong nhiều đoạn cùng với số lượng mẫu mỗi đoạn, cũng như số lượng các đoạn,khớp với việc chia đoạn của dữ liệu mẫu cơ sở hoặc (b) là trong một đoạn duy nhất đối với tất cả các mẫu trong bản mẫu movie (hoặc trong một phân mảnh movie). Thông tin bổ trợ mẫu của tất cả các mẫu trong một đoạn đơn lẻ (hoặc một đoạn hoạt động) được lưu liên tiếp (tương tự như dữ liệu mẫu).

Thông tin bổ trợ mẫu thường được lưu trữ trong cùng một tệp tin với mẫu mà nó có liên quan vì chúng có cùng cấu trúc dữ liệu tham chiếu (‘*dref*’). Tuy nhiên dữ liệu này có thể ở bất cứ đâu trong tệp tin vì vậy cần dùngđộ dịch chuyển thông tin bổ trợ (‘*saio*’) để xác định vị trí của dữ liệu.

Thông tin bổ trợ mẫu được cấp quyền hoặc được yêu cầu xác định bởi các nhãn hoặc định dạng mã hóa được sử dụng. Định dạngcủa thông tin bổ trợ mẫu được xác định bởi *aux\_info\_type*. Nếu *aux\_info\_type* và *aux\_info\_type\_parameter* bị bỏ qua thì giá trị của *aux\_info\_type* hoặc là (a) trong trường hợp nội dung thay đổi, ví dụ như nội dung cục bộ, *scheme\_type* bao gồm Hộp Thông tin Lược đồ Bảo vệ hoặc là (b) kiểu đầu vào mục mẫu. Giá trị mặc định của *aux\_info\_type\_parameter*bằng 0. Một số giá trị của *aux\_info\_type*chỉ được dùng đối với các kiểu rãnh cụ thể. Một rãnh có thể có nhiều luồng thông tin bổ trợ mẫu với các kiểu khác nhau. Các kiểu được đăng ký tại cơ quan đăng ký có thẩm quyền.

Trong khi *aux\_info\_type* xác định định dạng của thông tin bổ trợ, một vài luồng của thông tin bổ trợ có cùng định dạng có thể được sử dụng khi giá trị *aux\_info\_type\_parameter* của chúng khác nhau. Ngữ nghĩa của *aux\_info\_type\_parameter* đối với một *aux\_info\_type*cụ thể phải được xác định cùng với giá trị cụ thể của *aux\_info\_type* và định dạng thông tin bổ trợ.

Hộp này cung cấp kích thước của thông tin bổ trợ cho từng mẫu.Thí dụ với hộp này, cần phải đồng bộ SampleAuxiliaryInformationOffsetsBox với các giá trị của *aux\_info\_type*và *aux\_info\_type\_parameter* để cung cấp thông tin dịch chuyển cho những thông tin bổ trợ này.

#### Cú pháp

*aligned(8) class SampleAuxiliaryInformationSizesBoxextends FullBox(‘saiz’, version = 0, flags){*

*if (flags & 1) {*

*unsigned int(32) aux\_info\_type;*

*unsigned int(32) aux\_info\_type\_parameter;*

*}*

*unsigned int(8) default\_sample\_info\_size;*

*unsigned int(32) sample\_count;*

*if (default\_sample\_info\_size == 0) {*

*unsigned int(8) sample\_info\_size[ sample\_count ];*

*}*

*}*

#### Ngữ nghĩa

*aux\_info\_type*có kiểu số nguyênchỉ kiểu của thông tin bổ trợ mẫu. Hầu hết trong hộp này sẽ xuất hiện trường hợp giá trị của *aux\_info\_type*bằng*aux\_info\_type\_parameter*.

*aux\_info\_type\_parameter*xác định các “luồng“thông tin bổ trợ có cùng giá trị với *aux\_info\_type* và liên quan tới cùng mộtrãnh. Ngữ nghĩa của *aux\_info\_type\_parameter*được xác định bởi giá trị của *aux\_info\_type*.

*default\_sample\_info\_size*có kiểu số nguyên xác định kích thước của thông tin bổ trợ mẫu trong trường hợp mà tất cả các thông tin bổ trợ mẫu được chỉ định có cùng kích thước. Nếu kích thước khác nhau thì trường này sẽ có giá trị 0.

*sample\_count*có kiểu số nguyên xác định số lượng thông tin bổ trợ mẫu có kích thước xác định. Khi một Hộp Kích thước Thông tin Bổ trợMẫu xuất hiện trongHộp Bảng Mẫuphải bằng hoặc ít hơn *sample\_count* trong Hộp Kích thước Mẫu hoặc Hộp Kích thước Mẫu rút gọn.Khi một Hộp Kích thước Thông tin Bổ trợ Mẫu xuất hiện trong một Hộp Phân mảnh Rãnh thì nó phải bằng hoặc ít hơn tổng số đầu vào mục *sample\_count*trong cácHộp Xử lý Phân mảnh Rãnh củaphân mảnh rãnh. Nếu ít hơn số lượng mẫu thì thông tin bổ trợ được cung cấp cho các mẫu khởi tạo và các mẫu còn lại không có thông tin bổ trợ liên quan.

*sample\_info\_size* xác định kích thước của thông tin bổ trợ tính theo byte. Kích thước này có thể bằng 0 trong trường hợp các mẫu không có thông tin hỗ trợ liên quan.

### Hộp Độ dịch chuyểnThông tin Bổ trợMẫu (Sample Auxiliary Information Offsets Box)

#### Định nghĩa

Loại hộp:‘*saio*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’) hoặc Hộp Phân mảnh Rãnh ('*traf*').

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Để biết về thông tin bổ trợ mẫu, xem định nghĩa của Hộp Kích thước Thông tin Bổ trợ Mẫu.

Hộp này cung cấp thông tin về vị trí cho thông tin bổ trợ mẫu tương tự như độ dịch chuyển đoạn của dữ liệu mẫu.

#### Cú pháp

*aligned(8) class SampleAuxiliaryInformationOffsetsBoxextends FullBox(‘saio’, version, flags){*

*if (flags & 1) {*

*unsigned int(32) aux\_info\_type;*

*unsigned int(32) aux\_info\_type\_parameter;*

*}*

*unsigned int(32) entry\_count;*

*if* (*version == 0*) *{*

*unsigned int(32) offset[ entry\_count ];*

*}*

*else {*

*unsigned int(64) offset[ entry\_count ];*

*}*

*}*

#### Ngữ nghĩa

*aux\_info\_type* và *aux\_info\_type\_parameter* được định nghĩa như trongSampleAuxiliaryInformationSizesBox.

*entry\_count* xác định số lượng đầu vào mục. Khi Hộp Độ dịch chuyển Thông tin Bổ trợ Mẫuxuất hiện trong mộtHộp Bảng Mẫuthì *entry\_count* phải bằng 1 hoặc bằng giá trị của trường *entry\_count* trong Hộp Độ dịch chuyển Đoạn hoặc Hộp Độ dịch chuyển Lớn Đoạn. Khi Hộp Độ dịch chuyển Thông tin Bổ trợ Mẫu xuất hiện trongHộp Phân mảnh Rãnhthì *entry\_count* phải bằng 1 hoặc bằng số Hộp Xử lý Phân mảnh Rãnh trongHộp Phân mảnh Rãnh.

*offset*xác định vị trí của thông tin bổ trợ mẫutrong tệp tin ứng với mỗi đoạn hoặcxử phân mảnh rãnh. Nếu *entry\_count* bằng 1 thì thông tin bổ trợ mẫu của tất cả các đoạn hoặc các xử lýliên tiếp nhau trong tệp tin trong đoạn hoặc thứ tự xử lý. Trong Hộp Bảng Mẫu, độ dịch chuyển là tuyệt đối. TrongHộp Phân mảnh Rãnh, những giá trị này là tương đối với độ dịch chuyển cơ sở được thiết lập bởiHộp Mào đầu Phân mảnhRãnh (‘*tfhd*’) trong cùngphân mảnh rãnh (xem 8.8.14).

## Phân mảnh movie

### Hộp Movie Mở rộng (Movie Extends Box)

#### Định nghĩa

Loại hộp:‘*mvex*’.

Thuộc đối tượng: Hộp Movie (‘*moov*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp nàychỉ cho thiết bị đọc biết có thể có Hộp Phân mảnh Movie trong tệp tin. Để biết tất cả các mẫu trong các rãnh, các Hộp Phân mảnh Movienày phải được tìm thấy và rà soát đúng thứ tự, thông tin của các Hộp Phân mảnh Movie sẽ được bổ sung để giúp việc tìm ra chúng trong Hộp Movie.

Phần giới thiệu về phân mảnh movienằm trong Phụ lục A.

#### Cú pháp

*aligned(8) class MovieExtendsBox extends Box(‘mvex’){*

*}*

### Hộp Mào đầu Movie Mở rộng (Movie Extends Header Box)

#### Định nghĩa

Loại hộp:‘*mehd*’.

Thuộc đối tượng: Hộp Movie Mở rộng (‘*mvex*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Mào đầu Movie Mở rộng là một hộp tùy chọn,hộp này cung cấp toàn bộ khoảng thời gian, bao gồm các phân mảnh của một movie bị phân mảnh. Nếu hộp này không được thiết lập, toàn bộ khoảng thời gian phải được tính toán bằng cách kiểm tra từng phân mảnh.

#### Cú pháp

*aligned(8) class MovieExtendsHeaderBox extends FullBox(‘mehd’, version, 0) {*

*if (version==1) {*

*unsigned int(64) fragment\_duration;*

*} else { // version==0*

*unsigned int(32) fragment\_duration;*

*}*

*}*

#### Ngữ nghĩa

*fragment\_duration*có kiểu số nguyên xác định độ dài của bản trình diễn của toàn bộ movie bao gồm các phân đoạn (tính theo thang thời gian quy định trong Hộp Mào đầu Movie). Giá trị của trường này tương ứng vớikhoảng thời gian của rãnh dài nhất, bao gồm các phân mảnh movie. Nếu một tệp tin MP4 được tạo trong thời gian thực, ví dụ như streaming trực tuyến, nó không giống nhưviệc biết trước *fragment\_duration* và hộp này có thể bỏ qua.

### Hộp Rãnh Mở rộng (Track Extends Box)

#### Định nghĩa

Loại hộp:‘*trex*’.

Thuộc đối tượng: Hộp Movie Mở rộng (‘*mvex*’).

Hình thức sử dụng: bắt buộc.

Số lượng: Cần duy nhất 1 hộp cho mỗi rãnh trong Hộp Movie.

Hộp này thiết lập giá trị mặc định được sử dụng bởi phân mảnh movie. Bằng cách thiết lập giá trị mặc định theo cách này, không gian bộ nhớ (vùng nhớ) và độ phức tạp được giảm bớt trong mỗiHộp Phân mảnh Rãnh.

Các trường cờ mẫutrong các phân mảnh mẫu (*default\_sample\_flags* ở đây và trongHộp Mào đầu Phân mảnh Rãnh,*sample\_flags*và*first\_sample\_flags* trongHộp Xử lý Phân mảnh Rãnh) được mã hóa 32 bit. Nó có cấu trúc như sau:

*bit(4) reserved=0;*

*unsigned int(2) is\_leading;*

*unsigned int(2) sample\_depends\_on;*

*unsigned int(2) sample\_is\_depended\_on;*

*unsigned int(2) sample\_has\_redundancy;*

*bit(3) sample\_padding\_value;*

*bit(1) sample\_is\_non\_sync\_sample;*

*unsigned int(16) sample\_degradation\_priority;*

Cờ *sample\_is\_non\_sync\_sample*cung cấp thông tin giống như trong bảng mẫu đồng bộ [8.6.2]. Khi thiết lập giá trị 0 cho cờ này của một mẫu, điều đó có nghĩa mẫu này không nằm trong phân mảnh movie và được đánh dấu với một đầu vào mục trong bảng đầu vào mục mẫu (hoặc nếu tất cả các mẫu là mẫu đồng bộ thì không cần bảng mẫu đồng bộ).

*sample\_padding\_value*xác định bảng các bit chèn.

*sample\_degradation\_priority*xác định bảng ưu tiên suy giảm.

#### Cú pháp

*aligned(8) class TrackExtendsBox extends FullBox(‘trex’, 0, 0){*

*unsigned int(32) track\_ID;*

*unsigned int(32) default\_sample\_description\_index;*

*unsigned int(32) default\_sample\_duration;*

*unsigned int(32) default\_sample\_size;*

*unsigned int(32) default\_sample\_flags*

*}*

#### Ngữ nghĩa

*track\_id* xác định rãnh; đó là rãnh ID của một rãnh trongHộp Movie.

Các trường *default\_* này thiết lập giá trị mặc định được sử dụng trong các phân phân mảnh rãnh.

### Hộp Phân mảnh Movie (Movie Fragment Box)

#### Định nghĩa

Loại hộp:‘*moof*’

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Các phân mảnh movie mở bản trình diễn theo thời gian. Chúng cung cấp thông tin mà trước đó đã có trongHộp Movie. Thông thườngcác mẫu thực tế nằm trong các Hộp Dữ liệu Đa phương tiện, nếu chúng nằm trong cùng tệp tin. Chỉ số dữ liệu tham chiếu nằm trong miêu tả mẫu nên nó có thể xây dựng các bản trình diễn mở rộng, trong đó dữ liệu đa phương tiện nằm trong các tệp tin chứ không phải tệp tin chứa Hộp Movie.

Hộp Phân mảnh Movielà hộp có mức nhất (tức là ngang hàngvới Hộp Movie và các Hộp Dữ liệu Đa phương tiện). Nó chứa Hộp Mào đầu Phân mảnh Movievà một hoặc nhiềuHộp Phân mảnh Rãnh.

LƯU Ý: khôngnhất thiết bất kỳ phân đoạn movie cụ thể nào đều mở rộng tất cả các rãnh trong mào đầu movie, và không có hạn chế về vị trí của dữ liệu đa phương tiện tham chiếu bởi phân mảnh movie. Tuy nhiên, các thông số kỹ thuật dẫn xuất có thể có những hạn chế này.

#### Cú pháp

*aligned(8) class MovieFragmentBox extends Box(‘moof’){*

*}*

### Hộp Mào đầu Phân mảnh Movie (Movie Fragment Header Box)

#### Định nghĩa

Loại hộp:‘*mfhd*’

Thuộc đối tượng: Hộp Phân mảnh Movie(‘*moof*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Phần mào đầu của phân mảnh movie có trường số thứ tự như là một cách kiểm tra an toàn. Trường này thường bắt đầu bằng 1 vàtăng dần theo số phân mảnh movie trong tệp tinvà theo thứ tự xuất hiện. Điều này cho phép thiết bị đọc xác minh tính toàn vẹn của dãy số, nếu các phân mảnh không đúng thứ tự sẽ gây ra lỗi về cấu trúc.

LƯU Ý: khôngbắt buộc các số thứ tự phải liên tiếp nhau, chỉ cần số thứ tự của phân mảnh sau phải lớn hơn số thứ tự của phân mảnh trước.

#### Cú pháp

*aligned(8) class MovieFragmentHeaderBoxextends FullBox(‘mfhd’, 0, 0){*

*unsigned int(32) sequence\_number;*

*}*

#### Ngữ nghĩa

*sequence\_number* là số thứ tự của phân mảnh, tăng theo thứ tự.

### Hộp Phân mảnh Rãnh (Track Fragment Box)

#### Định nghĩa

Loại hộp:‘*traf*’.

Thuộc đối tượng: Hộp Phân mảnh Movie(‘*moof*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Trong phân mảnh movie có một tập các phân mảnhrãnh, 0 hoặc nhiều ứng với mỗi rãnh. Các phân mảnh rãnh không chứa hoặc chứa nhiều rãnh hoạt động, mỗi rãnh lưu một chuỗi liên tục các mẫu. Trong những cấu trúc này, một số trường là tùy chọn hoặc có giá trị mặc định.

Có thể thêm thời gian rỗng (empty time) vào rãnh sử dụng những cấu trúc này cũng như thêm các mẫu. Ví dụ thời gian rỗng có thể được sử dụng trong các rãnh âm thanh thể hiện khoảng im lặng.

#### Cú pháp

*aligned(8) class TrackFragmentBox extends Box(‘traf’){*

*}*

### Hộp Mào đầu Phân mảnh Rãnh (Track Fragment Header Box)

#### Định nghĩa

Loại hộp:‘*tfhd*’.

Thuộc đối tượng: Hộp Phân mảnh Rãnh (‘*traf*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Mỗi phân mảnh movie có thể không thêm hoặc thêm nhiều phân mảnh vào mỗi rãnh; một phân mảnh rãnh có thể không thêm hoặc thêm nhiều các mẫuhoạt động liên tiếp. Mào đầu phân mảnh rãnh thiết lập thông tin và các giá trị mặc định sử dụng cho cácmẫu hoạt động đó.

Các cờ sau được định nghĩa trong *tf\_flags*:

*0x000001base-data-offset-present*: chỉ sự hiện diện của trường *base-data-offset*. Nó cung cấp một neo rõ ràng cho độ dịch chuyển dữ liệu trong mỗi rãnh hoạt động (xem dưới đây). Nếu không được cung cấp, độ dịch chuyển dữ liệu cơ bản của rãnh đầu tiên trong phân mảnh movie sẽ là vị trí của byte đầu tiên của Hộp Phân mảnh Moviekèm theo, và cho các phân mảnh rãnh thứ hai và phân mảnh rãnh kế tiếp, giá trị mặc định nằm ở phần cuối của dữ liệu định nghĩa bởi phân mảnh trước. Tất cả các phân mảnhkế thừa (inheriting)các độ dịch chuyển theo cách nàyphải sử dụng cùng dữ liệu tham chiếu (ví dụ dữ liệu của những rãnh này phải ở trong cùng một tệp tin).

*0x000002sample-description-index-present*: chỉ ra sự có mặt của trường này, đồng thời trong phân mảnh này,giá trị vừa nêu sẽ thay thếgiá trị mặc định thiết lập trongHộp Mở rộng Rãnh.

*0x000008 default-sample-duration-present.*

*0x000010 default-sample-size-present.*

*0x000020 default-sample-flags-present.*

*0x010000 duration-is-empty*: chỉ ra khoảng thời gian trong *default-sample-duration*hoặc bởi khoảng thời gian mặc định trong Hộp Mở rộng Rãnh là khoảng rỗng, ví dụ như khi không có mẫunào trong khoảng thời gian. Sẽ có lỗi khi tạo một bản trình diễn có cả các danh sách biên tập chỉnh sửa trongHộp Movie và các phân mảnhkhoảng thời gian rỗng.

*0x020000 default-base-is-moof*: nếu *base-data-offset-present*bằng 0 thì *base-data-offset* của phân mảnh rãnh này là vị trí của byte đầu tiên của Hộp Phân mảnh Moviekèm theo. Việc hỗ trợ của cờ*default-base-is-moof* là do yêu cầu của nhãn ‘iso5’ và có thể không sử dụng trong các nhãn hoặc các nhãn tương đương trước ‘iso5’.

LƯU Ý:việc sử dụng cờ *default\_base\_is\_moof* phá vỡ khả năng tương thích với các nhãn trước đó của định dạng tệp tin này, vì nó thiết lập một điểm neo cho tính toán độ dịch chuyển khác với trước đó. Vì vậy, cờ*default\_base\_is\_moof*không được thiết lập khi các nhãn của phiên bản cũ tồn tại trong Hộp Kiểu Tệp tin.

#### Cú pháp

*aligned(8) class TrackFragmentHeaderBoxextends FullBox(‘tfhd’, 0, tf\_flags){*

*unsigned int(32) track\_ID;*

*// all the following are optional fields*

*unsigned int(64) base\_data\_offset;*

*unsigned int(32) sample\_description\_index;*

*unsigned int(32) default\_sample\_duration;*

*unsigned int(32) default\_sample\_size;*

*unsigned int(32) default\_sample\_flags*

*}*

#### Ngữ nghĩa

*base\_data\_offset*độ dịch chuyển cơ bản sử dụng khi tính toán độ dịch chuyển dữ liệu.

### Hộp Xử lý Phân mảnh Rãnh (Track Fragment Run Box)

#### Định nghĩa

Loại hộp:‘*trun*’.

Thuộc đối tượng: Hộp Phân mảnh Rãnh (‘*traf*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

TrongHộp Phân mảnh Rãnhcó thể có nhiềuhoặc không có Hộp xử lý Rãnh nào. Nếu cờ *duration-is-empty* được thiết lập trong *tf\_flags*, khi đó không có rãnh hoạt động. Một rãnh hoạt độngghi lại một tập liên tục các mẫu của một rãnh.

Số trường tùy chọn được xác định bởi số bit được thiết lập trong byte thấp của cờ và kích thước của bản ghi các bit thiết lập trong byte thứ hai của cờ. Thủ tục này được tuân theo cho phép định nghĩa các trường mới.

Nếu độ dịch chuyển dữ liệu không tồn tại, khi đó dữ liệu cho hoạt động này bắt đầu ngay sau dữ liệu của hoạt động trước, hoặc tại *base-data-offset*xác định bởi mào đầuphân mảnh rãnh nếu đây là hoạt động đầu tiên của phân mảnh rãnh. Nếu độ dịch chuyển dữ liệu xuất hiện, nó tương ứng với *base-data-offset* thiết lập trong mào đầu phân mảnhrãnh.

Các cờ sau được định nghĩa:

*0x000001 data-offset-present*.

*0x000004 first-sample-flags-present:*giá trị này sẽ thay thế các giá trị cờ mặc định chỉ đối với mẫu đầu tiên. Vì thế có thể ghi lại một nhóm các khung mà khung đầu tiên là quan trọng và sau đó là các khung khác mà không cần khai báo cờ cụ thể cho mọi mẫu. Nếu cờ và trường này được sử dụng thì *sample-flag* sẽ không xuất hiện.

*0x000100 sample-duration-present:* chỉ ra rằng mỗi mẫu có khoảng thời gian của riêng nó, nếu không sẽ sử dụng giá trị mặc định.

*0x000200sample-size-present*:mỗi mẫu có kích thước riêng của nó, nếu không sẽ sử dụng giá trị mặc định.

*0x000400 sample-flags-present:* mỗi mẫu có cờ của riêng nó, nếu không sẽ sử dụng giá trị mặc định.

*0x000800 sample-composition-time-offsets-present:* mỗi mẫu có một độ dịch chuyển thời gian tổng hợp (ví dụ như sử dụng hình ảnh I/P/B trong MPEG).

Các giá trị độ dịch chuyển tổng hợp trong Hộp Tổng hợp Thời gian và Mẫu và trong hộp rãnh hoạt động có thể có dấu hoặc không có dấu. Các khuyến nghị được cho hộp tổng hợp thời gian và mẫu liên quan đến việc sử dụng các độ dịch chuyển có dấu cũng được áp dụng ở đây.

#### Cú pháp

*aligned(8) class TrackRunBoxextends FullBox(‘trun’, version, tr\_flags) {*

*unsigned int(32) sample\_count;*

*// the following are optional fields*

*signed int(32) data\_offset;*

*unsigned int(32) first\_sample\_flags;*

*// all fields in the following array are optional*

*{*

*unsigned int(32) sample\_duration;*

*unsigned int(32) sample\_size;*

*unsigned int(32) sample\_flags*

*if (version == 0)*

*{ unsigned int(32) sample\_composition\_time\_offset; }*

*else*

*{ signed int(32) sample\_composition\_time\_offset; }*

*}[ sample\_count ]*

*}*

#### Ngữ nghĩa

*sample\_count*số lượng mẫu được chèn vào hoạt động; cũng là số dòng trong bảng sau (số dòng có thể rỗng).

*data\_offset* được chèn vào ngầm hoặc rõ vào *data\_offset* đã thiết lập trong mào đầu phân mảnhrãnh.

*first\_sample\_flags* cung cấp một bộ cờ cho duy nhất mẫu đầu tiên của hoạt động này.

### Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie (Movie Fragment Random Access Box)

#### Định nghĩa

Loại hộp:‘*mfra*’

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie (‘*mfra*’) cung cấp một bảng, bảng này có thể hỗ trợ thiết bị đọc tìm kiếm các mẫu đồng bộ trong tệp tin sử dụng các phân mảnhmovie. Nó bao gồm một phân mảnh Hộp Truy nhập Ngẫu nhiên Rãnh cho mỗi rãnh mà thông tin được cung cấp (có thể không phải tất cả các rãnh). Nó thường được đặt vào cuối hoặc gần cuối của tệp tin, hộp cuối cùng trong Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie cung cấp một bản sao của độ dài của trường trong Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie. Các thiết bị đọc có thể tìm hộp này bằng cách kiểm tra 32 bit cuối của tệp tin, hoặc quét ngược từ cuối của tệp tin để tìmHộp Độ dịch chuyển Truy nhập Ngẫu nhiên Phân mảnh Movie và sử dụng thông tin kích thước trong đó để tìm ra điểm bắt đầu của Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie.

Hộp này cho biếtmẫu đồng bộnằm ở đâu; bản thân phân mảnh movie được xác định. Khuyến nghị đối với thiết bị đọc nên lưu ý đến cả vị trí và việc sử dụng hộp nàykhi điều chỉnh tệp tin sau khi nó được tạo có thể kết xuất ra cảcác con trỏ hoặc khai báo mẫu đồng bộ không đúng.

#### Cú pháp

*aligned(8) class MovieFragmentRandomAccessBoxextends Box(‘mfra’){*

*}*

### Hộp Truy nhập Ngẫu nhiên Phân mảnh Rãnh (Track Fragment Random Access Box)

#### Định nghĩa

Loại hộp:‘*tfra*’.

Thuộc đối tượng: Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie(‘*mfra*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một ứng với mỗi rãnh.

Mỗi đầu vào mục bao gồm vị trí và thời gian trình diễn của mẫu đồng bộ. Lưu ýkhông phải mọi mẫu đồng bộ trong rãnh đều cần được liệt kê trong bảng.

Thiếu hộp này không có nghĩa là mọi mẫu đều là mẫu đồng bộ. Thông tin truy nhập ngẫu nhiên trong ‘*trun*’, ‘*traf*’ và ‘*trex*’ có thể được thiết lập ngay cả khi có sự xuất hiện của hộp này.

#### Cú pháp

*aligned(8) class TrackFragmentRandomAccessBoxextends FullBox(‘tfra’, version, 0) {*

*unsigned int(32) track\_ID;*

*const unsigned int(26) reserved = 0;*

*unsigned int(2) length\_size\_of\_traf\_num;*

*unsigned int(2) length\_size\_of\_trun\_num;*

*unsigned int(2) length\_size\_of\_sample\_num;*

*unsigned int(32) number\_of\_entry;*

*for(i=1; i <= number\_of\_entry; i++){*

*if(version==1){*

*unsigned int(64) time;*

*unsigned int(64) moof\_offset;*

*}*

*else{*

*unsigned int(32) time;*

*unsigned int(32) moof\_offset;*

*}*

*unsigned int((length\_size\_of\_traf\_num+1) \* 8) traf\_number;*

*unsigned int((length\_size\_of\_trun\_num+1) \* 8) trun\_number;*

*unsigned int((length\_size\_of\_sample\_num+1) \* 8) sample\_number;*

*}*

*}*

#### Ngữ nghĩa

*track\_ID* có kiểu số nguyênxác định *track\_id*.

*length\_size\_of\_traf\_num* chỉ ra độ dài theo byte của trường *traf\_number* trừ đi 1.

*length\_size\_of\_trun\_num* chỉ ra độ dài theo byte của trường *trun\_number* trừ đi 1.

*length\_size\_of\_sample\_num* chỉ ra độ dài theo byte của trường *sample\_number* trừ 1.

*number\_of\_entry*có kiểu số nguyêncho biết số lượng các đầu vào mục của rãnh này. Nếu giá trị này bằng 0 có nghĩa là mọi mẫu đều là mẫu đồng bộ và không có bảng đầu vào mục phía sau.

*time*có kiểu số nguyên 32 hoặc 64 bit chỉ thời gian trình diễn của mẫu đồng bộ tính theo đơn vị định nghĩa trong ‘*mdhd*’của rãnh liên kết.

*moof\_offset*có kiểu số nguyên 32 hoặc 64 bit chỉ độ dịch chuyểncủa ‘*moof*’ đã dùng trong đầu vào mục này. Độ dịch chuyểncơ bảnlà *byte-offset* giữa phần bắt đầu của tệp tin và phần bắt đầu của ‘*moof*’.

*traf\_number* chỉ số lượng ‘*traf*’ có chứa mẫu đồng bộ, mỗi *‘moof’* được đánh số bắt đầu từ 1(‘*traf*’ đầu tiên được đánh số thứ tự 1).

*trun\_number* chỉ số lượng ‘*trun*’ có chứa mẫu đồng bộ, mỗi *‘traf’*được đánh số bắt đầu từ 1.

*sample\_number* chỉ số mẫu đồng bộ, mỗi ‘*trun*’ được đánh số bắt đầu từ 1.

### Hộp Độ dịch chuyểnTruy nhập Ngẫu nhiên Phân mảnh Movie (Movie Fragment Random Access Offset Box)

#### Định nghĩa

Loại hộp:‘*mfro*’.

Thuộc đối tượng: Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie (‘*mfra*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này cung cấp một bản sao của trường độ dài của Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie đi kèm.Nó được đặt cuối cùng ở trong hộp đó, vì vậy trường kích thước cũng nằm ở phần cuối cùng của Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie. Khi Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie cũng nằm trong phần cuối của tệp tin này thì việc định vị trở nên dễ dàng hơn. Trường kích thước ở đây phải có giá trị cơ bản xác. Tuy nhiên, sự xuất hiện của Hộp Truy nhập Ngẫu nhiên Phân mảnh Movie và vị trí của nó có nằm ở cuối của tệp tin này hay không cũng không được đảm bảo chắc chắn.

#### Cú pháp

*aligned(8) class MovieFragmentRandomAccessOffsetBox extends FullBox(‘mfro’, version, 0) {*

*unsigned int(32) size;*

*}*

#### Ngữ nghĩa

*size*có kiểu số nguyên chỉ số byte của hộp ‘*mfra*’. Trường này thường được đặt ở phần cuối của hộp để hỗ trợ trong việc tìm kiếm hộp ‘*mfra*’ khi thiết bị đọc tìm kiếm từ phía cuối của tệp tin.

### Hộp Thuộc tính Mở rộng Rãnh (Track Extension Properties Box)

#### Định nghĩa

Loại hộp: `trep’

Thuộc đối tượng: Movie Extends Box (‘*mvex*’)

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều (với mỗi rãnh là không hoặc một)

Hộp này có thể được dùng để lưu trữ hoặc tổng hợp các đặc tính của rãnh trong các phân mảnh movie phía sau. Nó có thể chứa các hộp nhỏ với số lượng bất kỳ.

#### Cú pháp

*class TrackExtensionPropertiesBox extends FullBox(‘trep’, 0, 0) {*

*unsigned int(32) track\_id;*

*// Any number of boxes may follow*

*}*

#### Nghữ nghĩa

*track\_id*chỉ rãnh mà các tính năng mở rộng của nónằm trong hộp này.

### Hộp Thuộc tính Chuỗi Khởi động Thay thế (Alternative Startup Sequence Properties Box)

#### Định nghĩa

Loại hộp: ‘assp’

Thuộc đối tượng:Hộp Thuộc tính Mở rộng Rãnh (‘trep’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này thể hiện các thuộc tính của các nhóm mẫu chuỗi khởi động thay thế trong các phân mảnh rãnh kế tiếp của rãnh nằm trong Hộp Thuộc tính Mở rộng Rãnh.

Phiên bản 0 của Hộp Thuộc tính Chuỗi Khởi động Thay thế sẽ được sử dụng nếu phiên bản 0 của Hộp Ánh xạ Nhóm và Mẫu được dùng cho nhóm mẫu chuỗi khởi động thay thế. Phiên bản 1 của Hộp Thuộc tính Chuỗi Khởi động Thay thế sẽ được sử dụng nếu phiên bản 1 của Hộp Ánh xạ Nhóm và Mẫu được dùng cho nhóm mẫu chuỗi khởi động thay thế.

#### Cú pháp

*class AlternativeStartupSequencePropertiesBox extends FullBox(‘assp’, version, 0) {*

*if (version == 0) {*

*signed int(32) min\_initial\_alt\_startup\_offset;*

*}*

*else if (version == 1) {*

*unsigned int(32) num\_entries;*

*for (j=1; j <= num\_entries; j++) {*

*unsigned int(32) grouping\_type \_parameter;*

*signed int(32) min\_initial\_alt\_startup\_offset;*

*}*

*}*

*}*

#### Nghữ nghĩa

*min\_initial\_alt\_startup\_offset*: Không có giá trị sample\_offset[1] nào của đầu vào mô tả nhóm mẫu được tham chiếu nằm trong nhóm mẫu chuỗi khởi động thay thế nhỏ hơn min\_initial\_alt\_startup\_offset. Với phiên bản 0, nhóm mẫu chuỗi khởi động thay thế này sử dụng phiên bản 0 của Hộp Ánh xạ Nhóm và Mẫu. Với phiên bản 1 thì sẽ sử dụng phiên bản 1 của Hộp Ánh xạ Nhóm và Mẫu với ràng buộc được chỉ ra trong *grouping\_type \_parameter*.

*num\_entriesindicates*thể hiện số nhóm mẫu khởi động thay thế có trong hộp này.

*grouping\_type \_parameter*chỉ một nhóm mẫu thay thế mà đầu vào lặp sử dụng.

### Hộp Thời gian Giải mã Phân mảnh Rãnh (Track Fragment Decode Time Box)

#### Định nghĩa

Loại hộp:`*tfdt*’

Thuộc đối tượng:Hộp Phân mảnh Rãnh (‘*traf*’)

Hình thức sử dụng: không bắt buộc.

Số lượng:Không hoặc một.

Hộp Thời gian Giải mã Đa phương tiện Cơ sở Phân mảnh Rãnh (TrackFragment Base Media Decode Time Box) cung cấp thời gian giải mã tuyệt đối, dựa trên trục thời gian đa phương tiện. Điều này có thể hữu ích, chẳng hạn như khi truy nhập ngẫu nhiên vào một tệp tin, nó không nhất thiết phải tổng hợp tổng thời gian của tất cả các phân mảnh trước đó để tìm giá trị này (trong đó các khoảng thời gian mẫu là cac giá trị delta trong Hộp Mã hóa Thời gian và Mẫu và các *sample\_duration*của rãnh hoạt động trước đó).

Hộp Thời gian Giải mã Đa phương tiện Cơ sở Phân mảnh Rãnh, nếu có, sẽ được bố trí sauHộp Mào đầu Phân mảnh Rãnh và trướcHộp Xử lý Phân mảnh Rãnh đầu tiên.

LƯU Ý: thời gian mã hóa cơ bản là thời gian của đa phương tiện, được thiết lập trước khi ánh xạ thời gian của đa phương tiện với thời gian trình diễn, ví dụ như danh sách biên tập hoặc cấu trúc tương tự.

#### Cú pháp

*aligned(8) class TrackFragmentBaseMediaDecodeTimeBox extends FullBox(‘tfdt’, version, 0) {*

*if (version==1) {*

*unsigned int(64) baseMediaDecodeTime;*

*} else { // version==0*

*unsigned int(32) baseMediaDecodeTime;*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên chỉ phiên bản của hộp (0 hoặc 1).

*baseMediaDecodeTime*có kiểu số nguyên bằng có giá trị bằng tổng của các khoảng thời gian giải mã trước đó trong đa phương tiện với thang đo của đa phương tiện đó. Nó không bao gồm các mẫu được thêm vào trong phân mảnh rãnh đi kèm.

### Hộp Chỉ định Mức (Level Assignment Box)

#### Định nghĩa

Loại hộp:`*leva*’.

Thuộc đối tượng:Hộp Movie Mở rộng (‘*mvex*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Các mức (level) xác định các tập con của tệp tin. Các mẫu ánh xạ tới mức n có thể phụ thuộc vào bất kỳ mẫu nào ở các mức m, trong đó m<=n và không phụ thuộc vào bất kỳ mẫu nào ở các mức p, trong đó p>n. Ví dụ, các mức có thể được xác định dựa theo mức độ thời gian(chẳng hạn như *temporal\_id* của SVC hoặc MVC).

Các mức không được chỉ rõ ra trong movie đầu tiên. Khi Hộp Chỉ định Mức được dùng, nó áp dụng cho tất cả các phân mảnh movie kế tiếp sau movie ban đầu.

Đối với bối cảnh của Hộp Chỉ định Mức, một phần nhỏ được xác định bao gồm một hoặc nhiều Hộp Phân mảnh Movie và các Hộp Dữ liệu Đa phương tiện có liên quan, có thể chỉ bao gồm phần khởi tạo của Hộp Dữ liệu Đa phương tiện.

Trong một phần nhỏ, dữ liệu của mỗi mức sẽ xuất hiện liên tục kế tiếp nhau. Dữ liệu của các mức trong một phần nhỏ sẽ xuất hiện theo thứ tự tăng dần theo giá trị của mức. Tất cả dữ liệu trong một phần nhỏ sẽ được gán cho các mức.

LƯU Ý: Trong ngữ cảnh của DASH (ISO/IEC 23009-1), mỗi phân đoạn con được đánh chỉ số trong Hộp Chỉ số Phân đoạn con là một phần nhỏ.

Hộp Chỉ định Mức thể hiện ánh xạ các đặc tính, ví dụ như các lớp có khả năng mở rộng, vào các mức. Mỗi đặc tính có thể được chỉ ra thông qua một rãnh, một rãnh con trong một rãnh, hoặc một nhóm mẫu trong một rãnh.

Khi *padding\_flag* bằng 1 thì có nghĩa mỗi phần nhỏ thích hợp có thể được tạo ra bằng cách ghép các mức nguyên dương bất kỳ trong một phân nhỏ và chèn thêm vào cuối Hộp Dữ liệu Đa phương tiện các byte toàn số 0 cho đến khi đầy kích thước của hộp như ghi ở phần mào đầu của Hộp Dữ liệu Đa phương tiện cuối cùng. Ví dụ, *padding\_flag* có thể thiết lập giá trị 1 khi thỏa mãn các điều kiện sau:

* Mỗi phần nhỏ chứa một hoặc một số các rãnh AVC, SVC hoặc MVC [ISO/IEC 14496-15] của cùng một luồng bit hình ảnh.
* Các mẫu trong một rãnh của một phần nhỏ liền kề nhau và đúng thứ tự giải mã trong Hộp Dữ liệu Đa phương tiện.
* Các mẫu của mức AVC, SVC, hoặc MVC đầu tiên chứa các đơn vị NAL bao gồm các đơn vị NAL mã hóa hình ảnh từ các mức khác nhau của cùng một phần nhỏ đó.

#### Cú pháp

*aligned(8) class LevelAssignmentBox extends FullBox(‘leva’, 0, 0) {*

*unsigned int(8) level\_count;*

*for (j=1; j <= level\_count; j++) {*

*unsigned int(32) track\_id;*

*unsigned int(1) padding\_flag;*

*unsigned int(7) assignment\_type;*

*if (assignment\_type == 0) {*

*unsigned int(32) grouping\_type ;*

*}*

*else if (assignment\_type == 1) {*

*unsigned int(32) grouping\_type ;*

*unsigned int(32) grouping\_type \_parameter;*

*}*

*else if (assignment\_type == 2) {} // no further Cú pháp elements needed*

*else if (assignment\_type == 3) {} // no further Cú pháp elements needed*

*else if (assignment\_type == 4) {*

*unsigned int(32) sub\_track\_id;*

*}*

*// other assignment\_type values are reserved*

*}*

*}*

#### Ngữ nghĩa

*level\_count* chỉ số lượng các mứctrong mỗi phần nhỏ. *level\_count* lớn hơn hoặc bằng 2.

*track\_id* cho vòng lặp đầu vào mục j xác định cácđịnh danh rãnh của rãnh mức j.

*padding\_flag* bằng 1 chỉ ra rằng một phần nhỏ thích hợp có thể được hình thànhbằng cách liên kết các mức nguyên dương bất kỳ trong một phần nhỏ và chèn vào cuối Hộp Dữ liệu Đa phương tiệncác byte 0cho đến khi đầy kích thước như được chỉ ra trong mào đầu của Hộp Dữ liệu Đa phương tiện cuối cùng. *padding\_flag* bằng 0 nghĩa là không đảm bảo chắc chắn.

*assignment\_type* chỉ cơ chế sử dụng để chỉ cụ thể việc chỉ định cho một mức.

*assignment\_type*với giá trị lớn hơn 4 được lưu trữ, trong khi các giá trị khác được quy định như sau. Trình tự của *assignment\_type* được giới hạn là một tập hợp các số không hoặc nhiều kiểu 2 hoặc 3, tiếp đến là không có kiểu hoặc các nhiều kiểu cụ thể.

* 0: các nhóm mẫu được sử dụng để xác định các mức, ví dụ như, các mẫu ánh xạ tới các chỉ số miêu tả nhóm mẫu khác nhau của một mẫu cụ thể nằm ở các mức khác nhau trong một rãnh xác định;các rãnh khác không bị ảnh hưởng và dữ liệu của nó phải nằm trong một mức cụ thể.
* 1: tương tự như đối với *assignment\_type*bằng 0 ngoại trừ việc chỉ định bởi một nhóm mẫu có tham số.
* 2, 3: chỉ định mức bởi rãnh (xem Hộp Chỉ số Phân đoạn con để thấy sự khác nhau giữa các quá trình xử lý các mức này).
* 4:mức tương ứng chứa các mẫu của một rãnh con. Các rãnh con được xác định thông qua các Hộp Rãnh con; các rãnh khác không bị ảnh hưởng và tất cả dữ liệu phải nằm trong một mức cụ thể.

*grouping\_type*  và *grouping\_type \_parameter*: nếu tồn tại thì thể hiện các nhóm mẫu được sử dụng để ánh xạ các đầu mục miêu tảnhóm mẫu trong Hộp Mô tả Nhóm Mẫuvào các mức.Mức n chứa các mẫu được ánh xạ vào đầu vào mục miêu tảnhóm mẫu có chỉ số n trong Hộp Mô tả Nhóm Mẫu cùng giá trị*grouping\_type*  và*grouping\_type \_parameter,* nếu có, của các mức được cung cấp trong hộp này.

*sub\_track\_id*thể hiện rãnh con xác định bởi *sub\_track\_id*trong vòng lặp đầu vào mục j được ánh xạ vào mức j.

### Thông tin bổ trợ mẫu trong các phân mảnh movie

Khi thông tin bổ trợ mẫu(xem 8.7.8 và 8.7.9) được dùng trong Hộp Phân mảnh Movie thì các độ dịch chuyển trong Hộp Độ dịch chuyển Thông tin Bổ trợ Mẫu được thể hiện như *data\_offset* trongHộp Xử lý Phân mảnh Rãnh,có nghĩa là chúng liên quan đến bất kỳ độ dịch chuyển dữ liệu cơ sở nào được thiết lập chophân mảnh rãnh đó. Nếu các địa chỉphân mảnh movie được sử dụng (không có độ dịch chuyển dữ liệu cơ sở trong mào đầu phân mảnh rãnh)và các thông tin phụ trợ được dùng thì cờ *default\_base\_is\_moof* phải được thiết lập trong các cờ của mào đầuphân mảnh rãnh đó.

Nếu chỉ có một độ dịch chuyển thì thông tin bổ trợ mẫu trong các phân mảnhrãnh hoạt động trong phân mảnh đó sẽ được lưu liên tục, nếu không sẽ có cơ bản xác một độ dịch chuyển cho mỗi rãnh hoạt động.

Nếu trường *default\_sample\_info\_size* khác không thuộc một trong các hộp trên thì kích thước của thông tin bổ trợ không đổi đối với các mẫu được xác định.

Ngoài ra, nếu :

* Hộp nàytồn tại trong Hộp Movie và
* *default\_sample\_info\_size* khác không tronghộp này khi nằm trong Hộp Movie và
* Hộp kích thước thông tin bổ trợ mẫukhông tồn tại trong phân mảnh movie.

thì thông tin bổ trợ có kích thước không đổi đối với tất cả các mẫu trong các phân mảnh movie và không cần thiết phải lặp lại hộp này trong phân mảnh movie đó.

## Cấu trúc nhóm mẫu

### Giới thiệu

Phần này chỉ định một cơ chế chung cho đại diện một phân vùng của các mẫu trong một rãnh. Việc nhóm mẫulà việc gán mỗi mẫu trong một rãnhlà thành viên của một nhóm mẫu dựa trên sự phân kiểu nhóm. Một nhóm mẫu trong quá trình nhóm không bị giới hạn phải là các mẫu liên tục và có thể chứa các mẫu không liền nhau.Vì có thể có nhiều hơn một nhóm mẫuđối với các mẫu trong một rãnh nên mỗi nhóm có một trường kiểu để chỉ kiểu của nhóm. Ví dụ, một tệp tin có thể chứa hai nhóm mẫu trong cùng một rãnh: một dựa trên việc gán mẫu vào các lớp và một cho các chuỗi con.

Các nhóm mẫu được đại diện bởi hai cấu trúc dữ liệu có liên kết: (1) Hộp SampleToGroup thể hiện việc gán các mẫuvào các nhóm mẫu; (2) Hộp SampleGroupDescription chứađầu vào mục nhóm mẫucủa mỗi nhóm mẫumiêu tả các thuộc tính của nhóm. Có thể có nhiều hộp SampleToGroup và SampleGroupDescription dựa trên các tiêu chí nhóm khác nhau. Chúng được phân biệt bởi một trường kiểu dùng để chỉ ra kiểu của nhóm.

Một nhóm có kiểu cụ thể có thể sử dụng một tham số trong ánh xạ mẫu vào nhóm; nếu như vậy, ý nghĩa của tham số đó phải được ghi lại cùng nhóm đó. Một ví dụ về điều này làcó thể ghi lại các điểm đồng bộ trong một tập của một số luồng hình ảnh; định nghĩa của nhóm có thể là “Is an I-frame”,tham số nhóm có thể là định danh của từng luồng. Bởi vì hộp ánh xạ mẫu vào nhóm chỉ xuất hiện một lần đối với mỗi luồng, vì vậy nó được thu gọn và thông báo cho thiết bị đọc biết mỗi luồng riêng biệt.

Một ví dụ của việc sử dụng các bảng là thực hiện việc ấn định các mẫu vào các lớp. Trong trường hợp này mỗi nhóm mẫu đại diện cho một lớp, trong đó hộp SampleToGroup miêu tảrõ một mẫu nằm trong lớp nào.

### Hộp Ánh xạ Mẫu vào Nhóm (Sample to Group Box)

#### Định nghĩa

Loại hộp: ‘*sbgp*’.

Thuộc đối tượng: Hộp Bảng Mẫu(‘*stbl*’) hoặc Hộp Phân mảnh Rãnh (‘*traf*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không có hoặc nhiều.

Bảng này có thể được sử dụng để tìm nhóm có một mẫutrong đó và miêu tả liên quan của nhóm mẫu đó.Bảng này được mã hoá với mỗi đầu vào mục ứng với chỉ số của mẫu đầu tiên của các mẫu hoạt động trong cùng bộ miêu tả nhóm mẫu. Định danh miêu tả nhóm mẫu là một chỉ số đề cập đến một Hộp Mô tả Nhóm Mẫu, hộp này chứa các đầu vào mục mục miêu tả đặc tính của từng nhóm mẫu.

Có thể có nhiều hộp này nếu có nhiều hơn một nhóm mẫu của các mẫu trong một rãnh. Mỗi trường hợp của Hộp Ánh xạ Nhóm vào Mẫu có một kiểu mã để phân biệt các nhóm mẫu khác nhau. Trong một rãnh, phải có ít nhất một trường hợp của hộp này với một kiểu nhóm cụ thể. SampleGroupDescription tương ứng sẽ có giá trị giống nhau đối với một kiểu nhóm.

Phiên bản 1 hộp này chỉ nên được sử dụng nếu cần một tham số kiểu nhóm.

#### Cú pháp

*aligned(8) class SampleToGroupBox extends FullBox(‘sbgp’, version, 0){*

*unsigned int(32) grouping\_type ;*

*if (version == 1) {*

*unsigned int(32) grouping\_type \_parameter;*

*}*

*unsigned int(32) entry\_count;*

*for (i=1; i <= entry\_count; i++) {*

*unsigned int(32) sample\_count;*

*unsigned int(32) group\_description\_index;*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyênchỉ phiên bản của hộp (0 hoặc 1).

*grouping\_type* có kiểu số nguyên xác định các kiểu (i.e. tiêu chí được sử dụng để tạo thành các nhóm mẫu) của nhóm mẫu và liên kết nó với bảng miêu tảnhóm mẫu với cùng một giá trị cho kiểu nhóm. Hầu hết đều sẽ tồn tại một kiểu hộp này với cùng giá trị của *grouping\_type* (và nếu được sử dụng *grouping\_type \_parameter*) sẽ tồn tại trong một rãnh.

*grouping\_type \_parameter* là một dấu hiệu kiểu phụ của nhóm.

*entry\_count* có kiểu số nguyên cung cấp cho số lượng đầu vào mục trong bảng.

*sample\_count*có kiểu số nguyên cho biết số mẫu liên tiếp với cùng một miêu tả nhóm mẫu Nếu tổng của *sample\_count* trong hộp này nhỏ hơn tổng *sample\_count* thì thiết bị đọc nên mở rộng nó với một đầu vào mục liên quan tới các mẫu còn lại không thuộc nhóm nào. Sẽ có lỗi xảy ra khi tổng tỏng hộp này lại lớn hơn *sample\_count* được lưu, khi đó thiết bị đọc sẽ không biết xử lý như thế nào.

*group\_description\_index*có kiểu số nguyên cho biết chỉ số của đầu vào mục nhóm mẫu miêu tả các mẫu trong nhóm này. Khoảng giá trị của chỉ số dao động từ 1 đến số lượng các đầu vào mục nhóm mẫu trong Hộp SampleGroupDescription, hoặc giá trị 0 để chỉ ra rằng mẫu này không là thành viên của nhóm có cùng kiểu này.

### Hộp Mô tả Nhóm Mẫu (Sample Group Description Box)

#### Định nghĩa

Loại hộp: ‘*sbgp*’.

Thuộc đối tượng:Hộp Bảng Mẫu(‘*stbl*’) hoặc Hộp Phân mảnh Rãnh (‘*traf*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không có hoặc nhiều hơn, mỗi Hộp Ánh xạ Mẫu vào Nhóm có một Hộp Mô tả Nhóm Mẫu.

Bảng miêu tả này cung cấp các thông tin về các đặc tính của các nhóm mẫu. Thông tin miêu tả là bất kỳ thông tin nào khác cần thiết để xác định hoặc miêu tảnhóm mẫu.

Có thể có nhiều trường hợp của hộp này nếu có nhiều hơn một nhóm mẫu của các mẫu trong một rãnh. Mỗi trường hợp của Hộp Miêu tả Nhóm Mẫu có một kiểu mã để phân biệt các nhóm mẫu khác nhau. Trong một rãnh,hầu hêt phải có ít nhất một trường hợp của hộp này với một kiểu nhóm cụ thể. SampleToGroup có liên kết sẽ thể hiện cùng một giá trị đối với kiểu nhóm đó.

Thông tin được lưu trữ trong hộp miêu tả nhóm mẫusau *entry-count*. Một kiểu đầu vào mục trừu tượng được định nghĩa và nhóm mẫu sẽ xác định các kiểu dẫn xuất thể hiện miêu tảcủa mỗi nhóm mẫu. Đối với các rãnh hành ảnh, một VisualSampleGroupEntry trừu tượng được sử dụng với các kiểu tương tự như các rãnh âm thanh và các rãnh chỉ dẫn.

#### Cú pháp

*// Sequence Entry*

*abstract class SampleGroupDescriptionEntry (unsigned int(32) grouping\_type ){*

*}*

*abstract class VisualSampleGroupEntry (unsigned int(32) grouping\_type ) extends*

*SampleGroupDescriptionEntry (grouping\_type ) {*

*}*

*abstract class AudioSampleGroupEntry (unsigned int(32) grouping\_type ) extends*

*SampleGroupDescriptionEntry (grouping\_type ) {*

*}*

*abstract class HintSampleGroupEntry (unsigned int(32) grouping\_type ) extends*

*SampleGroupDescriptionEntry (grouping\_type ) {*

*}*

*abstract class SubtitleSampleGroupEntry (unsigned int(32) grouping\_type ) extendsSampleGroupDescriptionEntry (grouping\_type ){*

*}*

*abstract class TextSampleGroupEntry (unsigned int(32) grouping\_type ) extends SampleGroupDescriptionEntry (grouping\_type ){*

*}*

*aligned(8) class SampleGroupDescriptionBox (unsigned int(32) handler\_type) extends FullBox('sgpd', version, 0){*

*unsigned int(32) grouping\_type ;*

*if (version==1) {*

*unsigned int(32) default\_length;*

*}*

*unsigned int(32) entry\_count;*

*Int i;*

*for (i = 1 ; i <= entry\_count ; i++){*

*if (version==1) {*

*if (default\_length==0) {*

*unsigned int(32) description\_length;*

*}*

*}*

*switch (handler\_type){*

*case ‘vide’: // for video tracks*

*VisualSampleGroupEntry (grouping\_type );*

*break;*

*case ‘soun’: // for audio tracks*

*AudioSampleGroupEntry(grouping\_type );*

*break;*

*case ‘subt’: // for subtitle tracks*

*SubtitleSampleGroupEntry(grouping\_type );*

*break;*

*case ‘text’: // for text tracks*

*TextSampleGroupEntry(grouping\_type );*

*break;*

*case ‘hint’: // for hint tracks*

*HintSampleGroupEntry(grouping\_type );*

*break;*

*}*

*}*

*}*

#### Ngữ nghĩa

*version*có kiểu số nguyên chỉ định các phiên bản của hộp.

*grouping\_type* có kiểu số nguyên xác định Hộp SampleToGroupứng với miêu tảnhóm mẫu này.

*entry\_count* có kiểu số nguyêncho biết số lượng các đầu vào mục trong bảng.

*default\_length* cho biết độ dài của mỗi đầu vào mục nhóm (nếu độ dài không đổi), hoặc không (0) nếu đầu vào mục nhóm biến đổi.

*description\_length* cho biết độ dài của từng đầu vào mục nhóm, trong trường hợp có sự biến đổi từ đầu vào mục này sang đầu vào mục khác cũng như sự biến đổi của *default\_length*thì *description\_length* bằng 0.

### Đại diện của các cấu trúc nhóm trong các phân mảnh movie

Việc hỗ trợ các cấu trúc nhóm mẫu bên trong các phân mảnh movie được cung cấp bởi việc sử dụng Hộp SampleToGroupmà hộp này nằm trongHộp Phân mảnh Rãnh ('*traf*'). Định nghĩa, cú pháp và ngữ nghĩa của hộp nàyđược chỉ ra trong 8.9.2.

Hộp SampleToGroup có thể được sử dụng để tìm nhóm mà một mẫu trong một phân mảnh rãnhnằm trong nhóm đó và miêu tảliên quan của nhóm mẫu đó. Bảng này được mã hoá với mỗi mục nhập cho biết chỉ số của mẫu đầu tiên của hoạt động của các mẫu với bộ miêu tả cùng một nhóm mẫu. ID mô tả nhóm mẫu là một chỉ số liên quan đến Hộp SampleGroupDescription, hộp này chứa các đầu vào mục miêu tả các đặc tính của từng nhóm mẫu và hiện diện trong Hộp Bảng Mẫu.

Có thể có nhiều trường hợp của Hộp SampleToGroup nếu có nhiều hơn một nhóm mẫucủa các mẫu trong một phân mảnh rãnh. Mỗi trường hợp của Hộp SampleToGroup có một kiểu mã để phân biệt các nhóm mẫu khác nhau. SampleGroupDescription liên quan sẽ chỉ ra cùng một giá trị cho một kiểu nhóm.

Tổng số mẫu trong bất kỳ Hộp SampleToGroup trongphân mảnh rãnh phải trùng với tổng số mẫu trong tất cả cácphân mảnh rãnhhoạt động. Mỗi Hộp SampleToGroup ghi lại một nhóm biệt củacác mẫu giống nhau.

Việc tồn tại nhiều Hộp SampleGroupDescription hay không tồn tại hộp này có thể được chỉ ra trong Hộp Phân mảnh Rãnh. Các định nghĩa này được bổ sung vào định nghĩa về bảng mẫu của rãnh trongHộp Movie. Các định nghĩa về nhóm trong một phân mảnh movie cũng có thể được tham chiếu và được sử dụng trong cùngphân mảnh movie.

Với Hộp SampleToGroup trong cácphân mảnh movie, các chỉ số miêu tả nhóm của các nhóm trong cùng một phân mảnh bắt đầu tại 0x10001, ví dụ giá trị chỉ số 1 ứng với giá trị 1 trong 16bit đầu. Điều này có nghĩa là phải có ít hơn 65536 định nghĩa nhóm cho rãnh nàyvà kiểu nhóm trong bảng mẫu đó trong Hộp Movie.

Khi thay đổi kích thước của các phân mảnh movie hoặc kiểu bỏ chúng, những định nghĩa nhóm vềfragment-local sẽ cần phải được gắn với các định nghĩa trongHộp Movie hoặc gắn với những phân mảnh movie mới, và số chỉ số trong Hộp SampleToGroupphải được điều chỉnh cho phù hợp. Một số khuyến cáo đã chỉ ra, trong quá trình này, định nghĩa giống hệt nhau (và do đó trùng lặp) không được thực hiện trong bất kỳ hộp SampleGroupDescription nào, nhưng các trùng lặp cần được hợp nhất và các chỉ số cần được điều chỉnh cho phù hợp.

## Dữ liệu người dùng

### Hộp Dữ liệu Người dùng (User Data Box)

#### Định nghĩa

Loại hộp: ‘*udta*’.

Thuộc đối tượng: Hộp Movie (‘*moov*’) hoặc Hộp Rãnh (‘*trak*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này chứa các đối tượng dùng để khai báo thông tin của người dùng liên quan đến hộp chứa và dữ liệu của người dùng (bản trình diễn hoặc rãnh).

Hộp Dữ liệu Người dùng chứa thông tin tham khảo dữ liệu người dùng. Thông tin người dùng được định dạng gồm một tập các hộp với các kiểu hộp rất cụ thể khai báo nội dung cơ bản xác thông tin liên quan đến người dùng.

Trong tiêu chuẩn này chỉ đề cập đến thông tin về bản quyền.

#### Cú pháp

*aligned(8) class UserDataBox extends Box(‘udta’) {*

*}*

### Hộp Bản quyền (Copyright Box)

#### Định nghĩa

Loại hộp: ‘*cprt*’

Thuộc đối tượng: Hộp Dữ liệu Người dùng (‘*udta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp Bản quyền chứa tuyên bố về bản quyền của toàn bộ bản trình diễnnếu nó nằm trongHộp Movie, hoặc nếu nằm trong rãnh thì thể hiện bản quyền về toàn bộ rãnh đó. Có thể có nhiều Hộp Bản quyền sử dụng các mã ngôn ngữ khác nhau.

#### Cú pháp

*aligned(8) class CopyrightBox extends FullBox(‘cprt’, version = 0, 0) {*

*const bit(1) pad = 0;*

*unsigned int(5)[3] language; // ISO-639-2/T language code*

*string notice;*

*}*

#### Ngữ nghĩa

*language* là mã ngôn ngữ. Các mã này nằm trong chuẩn ISO 639-2 / T, mỗi mã bao gồm ba ký tự.Mỗi ký tự được tạo nên từ các giá trị khác nhau nằm trong khoảng giữa giá trị ASCII của cơ bản ký tự đó và 0x60. Mã này là các chữ gồm 3 chữ thường nên luôn có giá trị dương.

*notice* là một chuỗi ký tự UTF-8 hoặc UTF-16 kết thúc bởi ký tự trống, thể thông báo về bản quyền. Nếu sử dụng UTF-16, chuỗi sẽ bắt đầu với BYTE ORDER MARK (0xFEFF) để phân biệt với chuỗi sử dụng UTF-8. Dấu hiệu này không xuất hiện ở phần cuối của chuỗi.

### Hộp Chọn lựa Rãnh (Track Selection Box)

#### Giới thiệu

Một ví dụ điển hình được lưu trong tệp tinchứa một nhóm thay thế cho mỗi kiểuđa phương tiện: một cho hình ảnh, một cho âm thanh, vv. Tệp tinnày có thể bao gồm một vài rãnh hình ảnh, mặc dù, tại thời điểm bất kỳ chỉ một trong số đó được trình chiếu. Điều này đạt được bằng cách gán tất cả các rãnh hình ảnhvào cùng một nhóm thay thế. (Xem 8.3.2 về định nghĩa của các nhóm thay thế).

Tất cảcác rãnh trong một nhóm thay thế đều là những ứng cử viên cho việc lựa chọn đa phương tiện, những nó không có nghĩa là có thể chuyển giữa các rãnh trong một phiên. Ví dụ có thể chuyển giữa các rãnh hình ảnh có tốc độ bit khác nhau và giữ kích thước khung hình không đổi nhưng không được chuyển giữa các rãnhcó kích thước khung hình khác nhau. Tương tự, có thể cho phép lựa chọn-không phải chuyển đổi-giữa các rãnh có codec hình ảnh khác nhau hoặc ngôn ngữ âm thanh khác nhau.

Việc phân biệt các rãnh dùng để chọn lựa và các rãnh dùng để chuyển đổi được xử lý bằng cách gán các rãnh dùng để chuyển đổi vào các nhóm chuyển đổi thêm vào trong nhóm thay thế. Một nhóm thay thế có thể chứa một hoặc nhiều nhóm chuyển đổi. Tất cả rãnh trong nhóm thay thế đều có thể được dùng cho việc lựa chọn đa phương tiện, trong khi các rãnh trong một nhóm chuyển đổi cũng được cung cấp cho chuyển đổi trong một phiên. Các nhóm chuyển đổi khác đại diện cho điểm hoạt động khác nhau, chẳng hạn như kích thước khung hình khác nhau, chất lượng cao/thấp, vv.

Đối với trường hợp các luồng bit không khả năng mở rộng, một số rãnh có thể nằm trong cùng một nhóm chuyển đổi. Điều nêu trên cũng áp dụng cho các luồng bitcó khả năng mở rộng không phân lớp, chẳng hạn như các luồng AVC truyền thống.

Có thể đặc tính hóa các rãnh bằng cách gắn nhãn cho chúng theo các thuộc tính. Mỗi rãnh có thể được dán nhãn với một danh sách các thuộc tính có thể được sử dụng để miêu tả các rãnh trong một nhóm chuyển đổi cụ thể hoặc các rãnh khác nhau thuộc các nhóm chuyển đổi khác nhau.

#### Định nghĩa

Loại hộp: ‘tsel’.

Thuộc đối tượng: Hộp Dữ liệu Người dùng(‘*udta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp chọn lựa rãnh nằm trong hộp dữ liệu người dùng của rãnh mà nó thay đổi.

#### Cú pháp

*aligned(8) class TrackSelectionBox extends FullBox(‘tsel’, version = 0, 0) {*

*template int(32) switch\_group = 0;*

*unsigned int(32) attribute\_list[]; // to end of the box*

*}*

#### Ngữ nghĩa

*switch\_group*có kiểu số nguyên chỉ một nhóm hoặc một tập các rãnh. Nếu trường nàybằng 0 (giá trị mặc định) hoặc nếuHộp Chọn lựa Rãnh không tồn tại thì sẽkhông có thông tin về việc rãnh có thể được sử dụng để chuyển đổi trong quá trình phát hoặc truyền. Nếu trường này khác 0 thìnó sẽ là giá trị chung của các rãnh có thể được sử dụng để chuyển đổi giữa lẫn nhau. Các rãnh cùng một nhóm chuyển đổi sẽ thuộc về cùng một nhóm thay thế. Một nhóm chuyển đổi có thể có chỉ có một thành viên.

*attribute\_list* là một danh sách các thuộc tính, kéo dài đến cuối hộp. Các thuộc tính trong danh sách này nên được sử dụng như miêu tả của rãnh hoặc tiêu chí phân biệt của các rãnh trong cùng một nhóm thay thế hoặc nhóm chuyển đổi. Mỗi thuộc tính riêng biệt được kết hợp với một con trỏ trỏ đến đến trường này hoặc hoặc thông tin phân biệt rãnh.

#### Các thuộc tính

Các thuộc tính được mô tả như sau:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tên thuộc tính** | **Mã** | **Mô tả** |
|  | Khả năng mở rộng theo thời gian | *‘tesc’* | Rãnh có thể được mở rộng theo thời gian |
|  | Khả năng mở rộng theo SNR tinh | ‘*fgsc’* | Rãnh có thể được mở rộng về chất lượng |
|  | Khả năng mở rộng theo SNR thô | ‘*cgsc’* | Rãnh có thể được mở rộng về chất lượng |
|  | Khả năng mở rộng theo không gian | ‘*spsc’* | Rãnh có thể được mở rộng theo không gian |
|  | Khả năng mở rộng theo vùng ưa thích | ‘*resc’* | Rãnh có thể được mở rộng theo vùng ưa thích |
|  | Khả năng mở rộng theo ảnh | ‘*vwsc’* | Rãnh có thể được mở rộng về số lượng của các ảnh |
| Các thuộc tính khác biệt: | | | |
|  | **Tên thuộc tính** | **Mã** | **Mô tả** |
|  | Codec | ‘cdec’ | Nằm trong mục mẫu (của Hộp Mô tả Mẫu của rãnh đa phương tiện) |
|  | Kích thước màn hình | ‘scsz’ | Các trường chiều rộng và chiều cao của các mục mẫu hình ảnh |
|  | Kích thước gói tin lớn nhất | ‘mpsz’ | Trường Maxpacketsize trong Mục Mẫu Chỉ dẫn RTP |
|  | Kiểu đa phương tiện | ‘mtyp’ | Kiểu xử lý trong Hộp Xử lý của rãnh đa phương tiện |
|  | Ngôn ngữ đa phương tiện | ‘mela’ | Trường ngôn ngữ trong Hộp Mào đầu Đa phương tiện |
|  | Tốc độ bit | ‘*bitr’* | Tính bằng tổng kích thước của các mẫu chia cho khoảng thời gian trong Hộp Mào đầu Rãnh |
|  | Tốc độ khung | ‘*frar’* | Tính bằng số lượng mẫu trong một rãnh chia cho khoảng thời gian trong Hộp Mào đầu Rãnh |
|  | Số hiển thị | ‘*nvws’* | Số lượng hiển thị trong rãnh con |

Các thuộc tính mô tả đặc trưng cho các rãnh thay đổi, trong khi các thuộc tính khác biệt phân biệt các rãnhthuộc về nhóm thay thế hoặc nhóm chuyển đổi. Con trỏ của một thuộc tính khác biệt chỉ ra vị trí của các thông tin phân biệt rãnh này với các rãnh khác trong cùng một thuộc tính.

## Hỗ trợ siêu dữ liệu

Hộp siêu dữ liệu là một cấu trúc cơ sở phổ biến được dùng chứa siêu dữ liệu.

### Hộp Siêu dữ liệu (Meta Box)

#### Định nghĩa

Loại hộp: ‘*meta*’.

Thuộc đối tượng: tệp tin, Hộp Movie(‘*moov*’),Hộp Rãnh(‘*trak*’), Hộp chứa Siêu dữ liệu Bổ sung (‘*meco*’), Hộp Phân mảnh Movie(‘*moof*’) orHộp Phân mảnh Rãnh (‘*traf*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: - không hoặc một (đối với tệp tin, ‘*moov*’ và ‘*trak*’).

- một hoặc nhiều (đối với ‘*meco*’).

Hộp Siêu dữ liệu chứa miêu tả hoặc chú giải cho siêu dữ liệu. Hộp ‘*meta*’ cần chứa một hộp ‘*hdlr*’ để cho biết cấu trúc hoặc định dạng của nội dung hộp ‘*meta*’. Siêu dữ liệu này nằm trong một hộp mà hộp đó nằm trong hộp Siêu dữ liệu này (ví dụ hộp XML), hoặc được xác định bởi hạng mục trong hộp hạng mục cơ bản.

Tất cả các hộp chứa kháccó định dạng do hộp xử lý quyết định.

Các hộp khác được đề cập đến ở đây có thể được xác định như một tùy chọn hoặc bắt buộc đối với một định dạng. Nếu chúng được sử dụng thì chúng phải theo mẫu được định rõ ở đây. Các hộp tùy chọn bao gồm hộp thông tin dữ liệu, hộp này lưu thông tin các tệp tin chứa vị trí của các giá trị siêu dữ liệu (ví dụ các hình ảnh) và hộp vị trí hạng mục, hộp này lưu thông tin cho biết vị trí của hạng mục trong các tệp tin (ví dụ trường hợp thông thường các hình ảnh được lưu trong cùng một tệp tin). Hầu hết Hộp Siêu dữ liệuđều có thể xuất hiện tại mức tệp tin, tại mức movie, hoặc mức rãnh, trừ khi chúng nằm trong hộp chứa siêu dữ liệu bổ sung (‘*meco*’).

Nếu tồn tại Hộp Bảo vệ Hạng mục thì một số hoặc tất cả siêu dữ liệu, có thể bao gồm cả nguồn cơ sở, có thể được bảo vệ và không thể đọc được nếu không có hệ thống bảo vệ.

Khi tồn tại các Hộp Siêu dữ liệu trong phân mảnh movie hoặc các hộp phân mảnh rãnh thì những điều được đề cập sau đây sẽ được áp dụng. Tệp tin phải được phân mảnh như mọi siêu dữ liệu cần thiết trong movie hoặc phân mảnh rãnh được hình thành từ sự kết hợp của các siêu dữ liệu trong hộp movie và phân mảnh đó, không được quan tâm hoặc sử dụng siêu dữ liệu trong bất kỳ phân mảnh nào khác. Siêu dữ liệu trong movie hoặc phân mảnh rãnh về mặt logic ‘đến sau’ nhưng phù hợp với toàn bộ rãnh. Khi một tệp tin được hủy phân mảnh, siêu dữ liệu trong movie hoặc các phân mảnh rãnh đóphải được ghép lại trong các Hộp Movie hoặc Hộp Rãnh tương ứng. Cần đảm bảo không có những xung đột trong quá trình thực hiện đó – ví dụ không bị trùng lặp *item\_ID*, và nếu có, nó không xác định được rõ hạng mục nào được ưu tiên. Quá trình này cho phép phân phối ‘đúng theo thời gian’ các tài nguyên cần thiết và quản lý băng thông, trong khi vẫn giữ các đặc tính cơ bản không bị ảnh hưởng của siêu dữ liệu không được định thời. Nếu siêu dữ liệu thực sự thay đổi theo thời gian thì sẽ cần một rãnh siêu dữ liệu định thời.

#### Cú pháp

*aligned(8) class MetaBox (handler\_type) extends FullBox(‘meta’, version = 0, 0) {*

*HandlerBox(handler\_type) theHandler;*

*PrimaryItemBox primary\_resource;// optional*

*DataInformationBox file\_locations;// optional*

*ItemLocationBoxitem\_locations;// optional*

*ItemProtectionBoxprotections;// optional*

IteminfoBox*item\_infos;// optional*

*IPMPControlBox IPMP\_control;// optional*

*ItemReferenceBox item\_refs;// optional*

*ItemDataBoxitem\_data;// optional*

*Box other\_boxes[];// optional*

*}*

#### Ngữ nghĩa

Cấu trúc hoặc định dạng của siêu dữ liệu được khai báo bởi bộ xử lý. Trong trường hợp dữ liệu cơ bản được xác định bởi hạng mục cơ bản và mục cơ bảncơ bản này có thông tin đầu vào hạng mục *item\_type* thì kiểu xử lý có thể giống như *item\_type.*

### Hộp XML (XML Box)

#### Định nghĩa

Loại hộp:‘*xml* ‘or‘*bxml*’.

Thuộc đối tượng:Hộp Siêu dữ liệu(‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Khi dữ liệu cơ bản có định dạng XML và muốn XML đó được lưu trữ trực tiếp trong Hộp Siêu dữ liệu thì một trong các định dạng này có thể được sử dụng. Hộp nhị phân XML có thể chỉ được sử dụng khi tồn tại dạng nhị phân dễ xác định của XML mà định dạng này được bộ xử lý nhận biết.

Trong một Hộp XML dữ liệu định dạng theo UTF-8 trừ khi dữ liệu bắt đầu với một*byte-order-mark* (BOM) để chỉ ra rằng dữ liệu theo định dạng UTF-16.

#### Cú pháp

*aligned(8) class XMLBox extends FullBox(‘xml ’, version = 0, 0) {*

*string xml;*

*}*

*aligned(8) class BinaryXMLBox extends FullBox(‘bxml’, version = 0, 0) {*

*unsigned int(8) data[];// to end of box*

*}*

### Hộp Định vị Hạng mục (Item Location Box)

#### Định nghĩa

Loại hộp: ‘*iloc*’.

Thuộc đối tượng: Hộp Siêu dữ liệu (‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộpđịnh vị hạng mục chứa một thư mục các tài nguyênnằm trong tệp tin này hoặc trong các tệp tin khác nhau thông qua việc định vị trí tệp tin chứa các tài nguyên, độ dịch chuyển vị trí của cáctài nguyên đó trong tệp tin và độ dài của chúng. Việc xác định vị trí theo dạng nhị phân cho phép xử lý dữ liệu đó theo cách thông thường, thậm chí có thể dùng được cho các hệ thống không hiểu hệ thống siêu dữ liệu cụ thể nào đó. Ví dụ, một hệ thống có thể tích hợp tất cả các tài nguyên siêu dữ liệu tham chiếu từ bên ngoài vào một tệp tin, điều chỉnh lại các độ dịch chuyểntệp tin và theo các tham chiếu tệp tintương ứng.

Hộp này bắt đầu với3 hoặc 4 giá trị thể hiện kích thước của trường độ dịch chuyển, trường độ dài, trường *base\_offset*theo byte và các trường *extent\_index*trong phiên bản 1 và 2 của hộp này. Các giá trị này phải nằm trong tập giá trị {0,4,8}.

Trường *construction\_method* chỉ ra ‘cách thức xây dựng’ của hạng mục:

* *file\_offset*:thông qua độ dịch chuyểntệp tin tuyệt đối trong tệp tintại *data\_reference\_index*; (*construction\_method* ==0).
* *idat\_offset*: thông qua độ dịch chuyểnhộp trong hộp *idat*trong cùng Hộp Siêu dữ liệu; cả *data\_reference\_index*và *extent\_index*đều không được sử dụng; (*construction\_method* ==1).
* *item\_offset*: thông qua độ dịch chuyển hạng mục trong các hạng mục được chỉ ra bởi một trường *extent\_index*mới, mà chỉ được sử dụng (hiện tại) bởi cách thức xây dựng này. (*construction\_method*==2).

Trường *extent\_index*chỉ được sử dụng cho *item\_offset*; điều đó thể hiện chỉ số dựa trên 1 của tham chiếu hạng mục với kiểu tham chiếu ‘*iloc*’ liên kết từ hạng mục đó. Nếu *index\_size*bằng 0 thì giá trị 1 được dùng, giá trị 0 được dự phòng.

Các mục có thể được phân mảnh để lưu trữ vào trong các mở rộng, ví dụ, để cho phép đan xem. Một mở rộng là một tập con liền kề của các byte của tài nguyên; tài nguyên đó được hình thành bởi việc ghép các mở rộng. Nếu sử dụng duy nhất một mở rộng (*extent\_count* =1) khi đó một trong hai hoặc cả hai độ dịch chuyển và độ dài có thể được sử dụng:

•Nếu độ dịch chuyển không xác định (trường này có độ dài 0) thì phần đầu của nguồn đó (độ dịch chuyển bằng 0) được sử dụng.

• Nếu độ dài không xác định hoặc bằng 0 thì toàn bộ độ dài của nguồn đó đượcsử dụng. Các tham chiếu vào cùng tệp tinnhư siêu dữ liệu này hoặc các hạng mục được phân chia thành nhiều hơn một mở rộng, nên có một độ dịch chuyển và độ dài rõ ràng, hoặc sử dụng một kiểu MIME để yêu cầu diễn giải khác nhau của tệp tin, để tránh lặp vô hạn.

Kích thước của hạng mục bằng tổng của các độ dài mở rộng.

LƯU Ý: các mở rộng có thể được đan xen với các đoạn được xác định bởi các bảng mẫu của các rãnh.

Độ dịch chuyểnliên quan đến dữ liệu ban đầu, dữ liệu ban đầu này được xác định như sau:

1. Khi Hộp Metaở trong phân mảnh movie,*construction\_method* chỉ độ dịch chuyểntệp tinvà tham chiếu dữ liệu chỉ cùng tệp tin thì dữ liệu gốc là byte đầu tiên của Hộp Phân mảnh Movie(như đối với các cờ *default-base-is-moof*trong mào đầu phân mảnh rãnh);
2. Trong tất cả các trường hợp khác nhau khi *construction\_method* chỉ độ dịch chuyểntệp tin thì nguồn gốc dữ liệunằm trong phần đầu của tệp tin thể hiện tham chiếu dữ liệu; phần đầu của data[] trong Hộp Dữ liệu Hạng mục;
3. Khi *construction\_method* chỉ các độ dịch chuyểntrong Hộp Dữ liệu Hạng mục thì nguồn gốc dữ liệu nằm trong phần đầu của data[]trong Hộp Dữ liệu Hạng mục;
4. Khi tham chiếu dữ liệu chỉ hạng mục khác, nguồn gốc dữ liệu là byte đầu tiên của dữ liệu được liên kết (của tất cả các mở rộng) của hạng mục này.

LƯU Ý: có nhiều cách tính toán độ dịch chuyển khác nhau trong tài liệu này dựa vào phần đầu của mào đầu hộp, trong khi đó độ dịch chuyển dữ liệu hạng mục được tính dựa trên nội dung của hộp.

Chỉ số *data-reference* có thể bằng 0, điều đó có nghĩa tham chiếu trỏ đến cùng tệp tin như siêu dữ liệu hoặc chỉ số trỏ đến bảng *data-reference*.

Một vài dữ liệu được tham chiếu có thể sử dụng các kỹ thuật độ dịch chuyển/độ dài để chỉ các tài nguyên trong cơ bảnnó (ví dụ một tệp tin MP4). Thông thường các độ dịch chuyểnnày liên quan tới phần đầu của tệp tin chứa. Trường ‘độ dịch chuyển cơ sở’ cung cấp độ dịch chuyểnbổ sung cho việc tính toán độ dịch chuyển trong dữ liệu chứa đó. Ví dụ, nếu một tệp tin MP4 bao gồm cả một tệp tin được định dạng theo tiêu chuẩn này, khi đó thông thường độ dịch chuyển dữ liệu trong MP4 liên quan tới phần đầu của tệp tin; độ dịch chuyển cơ sởsẽ thêm vào các độ dịch chuyển đó.

Nếu một hạng mục được cấu thành từ các hạng mục khác và các hạng mục nguồn đó được bảo vệ, thì thông tin độ dịch chuyển và độ dài sẽ áp dụng cho các hạng mục nguồn đó sau khi chúng được gỡ bảo vệ. Điều đó nghĩa là dữ liệu hạng mục đích được định dạng từ dữ liệu nguồn không được bảo vệ.

Để khả năng tương thích lớn nhất, phiên bản 0 của hộp này sẽ được sử dụng trước phiên bản 1 với *construction\_method* ==0, hoặc phiên bản 2 khi có thể. Tương tự như vậy, phiên bản 2 của hộp này chỉ nên được dùng khi cần hỗ trợ *item\_ID*có giá trị lớn (trên 65535).

#### Cú pháp

*aligned(8) class ItemLocationBox extends FullBox(‘iloc’, version, 0) {*

*unsigned int(4) offset\_size;*

*unsigned int(4) length\_size;*

*unsigned int(4) base\_offset\_size;*

*if ((version == 1) || (version == 2)) {*

*unsigned int(4) index\_size;*

*} else {*

*unsigned int(4) reserved;*

*}*

*if (version < 2) {*

*unsigned int(16) item\_count;*

*} else if (version == 2) {*

*unsigned int(32) item\_count;*

*}*

*for (i=0; i<item\_count; i++) {*

*if (version < 2) {*

*unsigned int(16) item\_ID;*

*} else if (version == 2) {*

*unsigned int(32) item\_ID;*

*}*

*if ((version == 1) || (version == 2)) {*

*unsigned int(12) reserved = 0;*

*unsigned int(4) construction\_method;*

*}*

*unsigned int(16) data\_reference\_index;*

*unsigned int(base\_offset\_size\*8) base\_offset;*

*unsigned int(16) extent\_count;*

*for (j=0; j<extent\_count; j++) {*

*if (((version == 1) || (version == 2)) && (index\_size > 0)) {*

*unsigned int(index\_size\*8) extent\_index;*

*}*

*unsigned int(offset\_size\*8) extent\_offset;*

*unsigned int(length\_size\*8) extent\_length;*

*}*

*}*

*}*

#### Ngữ nghĩa

*offset\_size*lấy từtập {0,4,8} và cho biết độ dài các byte của trường độ dịch chuyển.

*length\_size*lấy từ tập {0,4,8} và cho biết độ dài các byte của trường độ dài.

*base\_offset\_size*lấy giá trị từ tập {0,4,8} và cho biết độ dài các của trường *base\_offset*.

*index\_size* lấy từ tập {0,4,8} và cho biết độ dài các byte của trường *extent\_index*.

*item\_count*đếm số lượng các tài nguyên trong mảng sau.

*item\_ID*có kiểu số nguyên tùy ý, ‘tên’ cho nguồn này mà có thể được sử dụng tham chiếu tới nó (ví dụ trongmột URL).

*construction\_method*có giá trị nằm trong tập 0 (tệp tin), 1 (*idat*) hoặc 2 (hạng mục).

*data-reference-index* hoặc bằng 0 (‘tệp tin này’) hoặc 1 dựa vào chỉ số trong các tham chiếu dữ liệu trong hộp thông tin dữ liệu.

*base\_offset*cho biết giá trị cơ bản của các tính toán độ dịch chuyển trong dữ liệu tham chiếu. Nếu *base\_offset\_size* bằng 0 thì*base\_offset*bằng 0,điều đó có nghĩa là nó không được sử dụng.

*extent\_count* đếm số lượng của các mở rộng mà tài nguyên đó bị phân mảnh; nó phải có giá trị 1 hoặc lớn hơn.

*extent\_index*cung cấp một chỉ số như được xác định cho phương thức cấu tạo.

*extent\_offset* cho biết độ dịch chuyển tuyệt đối tính theo byte của nguồn gốc dữ liệu của các tệp tin chứa, của dữ liệu mở rộng. Nếu *offset\_size*bằng 0 thì *extent\_offset* cũng bằng 0.

*extent\_length*cho biết độ dài tuyệt đối tính theo byte của mở rộng hạng mục siêu dữ liệu này. Nếu *length\_size*bằng 0 thì *extent\_length* cũng bằng 0. Nếu *extent\_lenght*bằng 0, khi đó độ dàimở rộng này bằngđộ dài của toàn bộ vật chứa được tham chiếu.

### Hộp Hạng mục Cơ bản (Primary Item Box)

#### Định nghĩa

Loại hộp: ‘*pitm*’.

Thuộc đối tượng: Hộp Siêu dữ liệu (‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Đối với bộ xử lý đã cho, dữ liệu cơ bản có thể là một trong số các hạng mục được tham chiếu khi nó mong muốn được lưu trữ ở một nơi nào đó, hoặc được phân chia thành các mở rộng; hoặc các siêu dữ liệu bản có thể nằm trong Hộp Siêu dữ liệu (ví dụ trong hộp XML). Hoặc hộp này phải xuất hiện, hoặc có một hộp trong Hộp Siêu dữ liệu (ví dụ một hộp XML) chứa thông tin cơ bản trong định dạng được yêu cầu bởi bộ xử lý.

#### Cú pháp

*aligned(8) class PrimaryItemBox extends FullBox(‘pitm’, version, 0) {*

*if (version == 0) {*

*unsigned int(16) item\_ID;*

*} else {*

*unsigned int(32) item\_ID;*

*}*

*}*

#### Ngữ nghĩa

*item\_ID* là định danh của hạng mục cơ sở. Phiên bản 1 chỉ nên được dùng khi *item\_ID*có giá trị lớn (trên 65535).

### Hộp Bảo vệ Hạng mục (Item Protection Box)

#### Định nghĩa

Loại hộp: ‘*ipro*’.

Thuộc đối tượng: Hộp Siêu dữ liệu(‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Bảo vệ Hạng mục cung cấp một chuỗi thông tin bảo vệ hạng mục để sử dụng trongHộp Thông tin Hạng mục.

#### Cú pháp

*aligned(8) class ItemProtectionBox extends FullBox(‘ipro’, version = 0, 0) {*

*unsigned int(16) protection\_count;*

*for (i=1; i<=protection\_count; i++) {*

*ProtectionSchemeInfoBox protection\_information;*

*}*

}

### Hộp Thông tin Hạng mục (Item Information Box)

#### Định nghĩa

Loại hộp: ‘*iinf*’.

Thuộc đối tượng: Hộp Siêu dữ liệu (‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp thông tin hạng mục cung cấp thông tin bổ sung về các hạng mục được lựa chọn, bao gồm cả tên kí hiệu (‘tệp tin’). Nó có thể xuất hiện tùy ý, nhưng nếu xuất hiện, nó phải được giải thích, vì việc bảo vệ hạng mục hoặc mã hóa nội dung có thể thay đổi định dạng của dữ liệu trong hạng mục đó. Nếu cả việc mã hóa nội dung và việc bảo vệ đượcdùng cho một hạng mục, thiết bị đọc trước hết sẽ gỡ bảo vệ của hạng mục đó sau đó giải mã nội dung mã hóa của hạng mục. Nếu cần kiểm soát kỹ hơn thì có thể sử dụng mã hóa chuỗi IPMP.

Hộp này chứa một mảng các mục đầu vào và mỗi đầu vàomục ấy được định dạng như một hộp. Mảng này được sắp xếp theo thứ tự tăng dần của *item\_ID* trong các bản ghi đầu vào mục đó.

Có 4 phiên bản đầu vào thông tin hạng mục. Phiên bản 1 bao gồm các thông tin bổ sung cho phiên bản 0 và được chỉ ra thông qua kiểu mở rộng. Ví dụ, kiểu mở rộng ‘fdel’ cho các hạng mục được tham chiếu bởi Hộp Phân vùng Tệp tin (*‘fpar’*), kiểu mở rộng này xác định các phân vùng tệp tinnguồnvà được sử dụng cho truyền phân phối tệp tin. Phiên bản 2 và 3 cung cấp một cấu trúc thay thế trong đó các kiểu hạng mục siêu dữ liệu được chỉ rõ bởi mã đã được đăng ký hoặc mã đã được xác nhận 32 bit (thường là 4 kí tự);hai trong số các mã này được dùng để chỉ kiểu MIME hoặc siêu dữ liệu được phân kiểu bởi một URI. Phiên bản 2 hỗ trợ các *item\_ID* 16 bit, trong khi đó phiên bản 3 hỗ trợ *item\_ID* 32 bit.

Nếu không có phần mở rộng mong muốn, hộp này có thể kết thúc mà không có trường *extention\_type* và mở rộng; ngoài ra nếu không cần *content\_encoding*thìtrường này cũng có thể không tồn tại và hộp này kết thúc trước đó. Nếu tồn tại phần mở rộng mà không có*content\_encoding* thì một byte trống đơn nhất, thể hiện một chuỗi trống, phải được ấn định cho *content\_encoding*, trước khi nhận biết *extentsion\_type*.

Nếu cần thông tin hạng mục phân phối tệp tin và một Ite*minf*oEntry phiên bản 2 hoặc 3 được sử dụng thì thông tin phân phối tệp tinđó sẽ được lưu theo dạng: (a) như một hạng mục riêng biệt của kiểu ‘fdel’, (b) được liên kết bởi một tham chiếu hạng mục từ hạng mục đó đến thông tin phân phốitệp tin của kiểu ‘fdel’. Chỉ có một tham chiếu duy nhất đó nếu sử dụng thông tin phân phối tệp tin.

Có thể có các dạng URI thích hợp cho siêu dữ liệu MPEG-7 (ví dụ lược đồ URI với một phân mảnh mà phân mảnh này xác định một thành tố cụ thể) và có thể dùng cấu trúc đó cho MPEG-7.Tuy nhiên, việc hỗ trợ MPEG-7 trong họ cáctệp tintuân theo định dạng tiệp tin đa phương tiệncơ sở theo chuẩn ISO là có thực và sự hỗ trợ này được ưu tiên do nó cho phép:

1. Cập nhập thêm siêu dữ liệu (về mặt logic, mã hóa I/P, trong các thuật ngữ hình ảnh) trong khi bản nháp chỉ cókhung hình I;
2. Nhị phân hóa và nén;
3. Sử dụng đa lược đồ.

Do đó, việc sử dụng các cấu trúc nêu trên cho MPEG-7 không được tán thành (và không được ghi vào tài liệu).

Thông tin về các dạng URI của một số hệ thống siêu dữ liệu có thể được chỉ ra trong Phụ lục F.

Phiên bản 1 của IteminfoBox chỉ nên được sử dụng khi cần iteminfoEntries có giá trị lớn (trên 65535).

#### Cú pháp

*aligned(8) class IteminfoExtension(unsigned int(32) extension\_type) {*

*}*

*aligned(8) class FDIteminfoExtension() extends IteminfoExtension (’fdel’) {*

*string content\_location;*

*string content\_MD5;*

*unsigned int(64) content\_length;*

*unsigned int(64) transfer\_length;*

*unsigned int(8) entry\_count;*

*for (i=1; i <= entry\_count; i++)*

*unsigned int(32) group\_id;*

*}*

*aligned(8) class IteminfoEntry extends FullBox(‘infe’, version, 0) {*

*if ((version == 0) || (version == 1)) {*

*unsigned int(16) item\_ID;*

*unsigned int(16) item\_protection\_index*

*string item\_name;*

*string content\_type;*

*string content\_encoding; //optional*

*}*

*if (version == 1) {*

*unsigned int(32) extension\_type;//optional*

*IteminfoExtension(extension\_type);//optional*

*}*

*if (version >= 2) {*

*if (version == 2) {*

*unsigned int(16) item\_ID;*

*} else if (version == 3) {*

*unsigned int(32) item\_ID;*

*}*

*unsigned int(16) item\_protection\_index;*

*unsigned int(32) item\_type;*

*string item\_name;*

*if (item\_type==’mime’) {*

*string content\_type;*

*string content\_encoding; //optional*

*} else if (item\_type == ‘uri ‘) {*

*string item\_uri\_type;*

*}*

*}*

*}*

*aligned(8) class* IteminfoBox *extends FullBox(‘iinf’, version, 0) {*

*if (version == 0) {*

*unsigned int(16) entry\_count;*

*} else {*

*unsigned int(32) entry\_count;*

*}*

*IteminfoEntry[ entry\_count ] item\_infos;*

*}*

#### Ngữ nghĩa

*Item\_ID* chứa cả giá trị 0 đối với tài nguyên cơ bản (ví dụ, XML chứa trong một hộp ‘*xml*’) hoặc ID của hạng mục xác định thông tin sau.

*item\_protection\_index*chứa cả giá trị 0 cho một hạng mục không được bảo vệ, hoặc chỉ số 1 trong hộp bảo vệ hạng mục thể hiện việc bảo vệ đó được áp dụng cho hạng mục này (hộp đầu tiên trong hộp bảo vệ hạng mục có chỉ số là 1).

*item\_name* là một chuỗi ký tự UTF-8 kết thúc bởi kí tự trống chứa kí hiệu tên của hạng mục (tệp tin nguồn cho việc truyền phân phối tệp tin).

*item\_type*là một giá trị 32 bit, thường là 4 kí tự có thể hiển thị được, đó là một chỉ thịkiểu hạng mục hợp lệ ví dụ như ‘mime’.

*content\_type*là một chuỗi kí tự UTF-8 kết thúc bởi kí tự trống với kiểu MIME của hạng mục đó. Nếu hạng mục này được mã hóa nội dung (xem bên dưới), khi đó kiểu nội dung tham chiếu tới mục sau giải mã nội dung

*item\_uri\_type*là một chuỗi chỉ giá trị tuyệt đối của một URI được sử dụng như một chỉ thị kiểu.

*content\_encoding*là một chuỗi kí tự UTF-8 tùy chọnkết thúc bởi kí tự trống được sử dụng để chỉ tệp tin nhị phân được mã hóa và cần để giải mã trước khi được thông dịch. Các giá trị này được xác định như mã hóa nội dung cho HTTP/1.1. Một vài gía trị có thể là “gzip “,”compress “và “deflate “. Một chuỗi rỗng chỉ ra rằng không mã hóa nội dung. Lưu ý rằng hạng mục này được lưu sau khi mã hóa nội dung.

*extension\_type*là một mã 4 kí tự có thể in được mà dùng để nhận biết các trường mở rộng của phiên bản 1 so với phiên bản 0 của đầu vào mục thông tin hạng mục.

*content\_location* là chuỗi ký tự UTF-8 kết thúc bởi kí tự trống chứa URI của tệp tinđược chỉ ra trong HTTP/1.1 (RFC 2616).

*content\_MD5*là chuỗi ký tự UTF-8 kết thúc bởi kí tự trống chứa một luật mã hóa MD5 cho một tệp tin. Xem HTTP/1.1 (RFC 2616) và RFC 1864.

*content\_lenght* cho biết tổng độ dài (theo byte) của tệp tin(không mã hóa).

*transfer\_length* cho biết tổng độ dài (theo byte) của tệp tin (được mã hóa). Lưu ý rằng độ dài truyền bằng độ dài nội dung nếu không áp dụng mã hóa nội dung (xem bên trên).

*entry\_count* đếm số lượng của các đầu vào mục trong mảng sau.

*group\_ID* cho biết một nhóm tệp tinchứa hạng mục tệp tin đó (tệp tin nguồn). Xem 3GPP TS 26.346 để biết chi tiết về các nhóm tệp tin.

### Hộp chứa Siêu dữ liệu Bổ sung (Additional Metadata Container Box)

#### Định nghĩa

Loại hộp: ‘*meco*’.

Thuộc đối tượng: tệp tin,Hộp Movie (‘*moov*’), hoặcHộp Rãnh (‘*trak*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp chứa siêu dữ liệu bổ sung gồm một hoặc nhiều Hộp Siêu dữ liệu. Nó có thể được thực hiện tại mức cao của một tệp tin, trongHộp Movie (‘*moov*’), hoặc trongHộp Rãnh (‘*trak*’) và chỉ xuất hiện nếu nó đi kèm bởi một Hộp Siêu dữ liệu trong cùng vật chứa. Hộp Siêu dữ liệu không nằm trong hộp chứa siêu dữ liệu bổ sung là Hộp Siêu dữ liệucơ bản (được ưu tiên). Các Hộp Siêu dữ liệunằm trong hộp chứa siêu dữ liệu bổ sung bổ sung hoặc cung cấp thông tin siêu dữ liệu thay thế. Việc sử dụng nhiều Hộp Siêu dữ liệulà cần thiết ví dụ như khi một bộ xử lý đơn không có khă năng xử lý tất cả các siêu dữ liệu. Tất cả các Hộp Siêu dữ liệu ở một mức nhất định, bao gồm cả hộp được ưu tiên và hộp nằm trong hộp chứa siêu dữ liệu bổ sung, cần phải có các bộ xử lý khác nhau.

Một Hộp Siêu dữ liệunằm trong hộp chứa siêu dữ liệu bổ sung cần chứa một hộp hạng mục cơ bản hoặc hộp dữ liệu cơ bản theo yêu cầu của bộ xử lý (ví dụ, một hộp XML). Nó không bao gồm các hộp hoặc các thành phần cú pháp liên quan đến các mục khác nhau so với hạng mục cơ bản được chỉ ra bởi hộp hạng mục cơ bảnđang tồn tại hoặc hộp XML. Các URL trong Hộp Siêu dữ liệunằm trong hộp chứa siêu dữ liệu bổ sung liên quan đến bối cảnh của Hộp Siêu dữ liệuđược ưu tiên.

#### Cú pháp

*aligned(8) class AdditionalMetadataContainerBox extends Box('meco') {*

*}*

### Hộp Quan hệ Hộp siêu dữ liệu (Metabox Relation Box)

#### Định nghĩa

Loại hộp: ‘*mere*’.

Thuộc đối tượng: Hộp Chứa Siêu dữ liệu Bổ sung (‘*meco*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp Quan hệ Hộp siêu dữ liệu chỉ ra một mối quan hệ giữa hai Hộp Siêu dữ liệu tại cùng mức, ví dụ mức cao của tệp tin,Hộp Movie, hoặcHộp Rãnh. Mối liên hệ giữa hai Hộp Siêu dữ liệu không được xác định nếu không có Hộp Quan hệ Hộp Siêu dữ liệu cho các Hộp Siêu dữ liệu đó. Các Hộp Siêu dữ liệu được tham chiếu bằng bởi việc xác định các kiểuxử lý của chúng.

#### Cú pháp

*aligned(8) class MetaboxRelationBox extends FullBox('mere', version=0, 0) {*

*unsigned int(32)first\_metabox\_handler\_type;*

*unsigned int(32)second\_metabox\_handler\_type;*

*unsigned int(8)metabox\_relation;*

}

#### Ngữ nghĩa

*first\_metabox\_handler\_type*cho biết Hộp Siêu dữ liệu đầu tiên có liên quan.

*second\_metabox\_handler\_type*cho biết Hộp Siêu dữ liệu thứ hai có liên quan.

*metabox\_relation*cho biết mối liên hệ giữa hai Hộp Siêu dữ liệu. Các giá trị sau đây được xác định:

* Mối liên hệ giữa các hộp này không được xác định (đó là giá trị mặc định khi hộp này không tồn tại);
* Hai hộp này về mặt ngữ nghĩa là không liên quan đến nhau (ví dụ, một là bản trình diễn, một bản chú giải);
* Hai hộp này về mặt ngữ nghĩa là liên quan đến nhau nhưng bổ sung cho nhau (ví dụ, hai tập rời nhau của siêu dữ liệu được biểu thị trong hai hệ thống siêu dữ liệu riêng biệt);
* Hai hộp này về mặt ngữ nghĩa là liên quan đến nhau nhưng chồng lấn lên nhau (ví dụ, hai tập của siêu dữ liệu mà tập này là tập con của tập kia); không hộp nào được ưu tiên hơn hộp nào.
* Hai hộp này về mặt ngữ nghĩa là liên quan đến nhau nhưng hộp thứ hai là tập con hoặc phiên bản không đầy đủ của hộp thứ nhất; hộp đầu tiên được ưu tiên hơn;
* Hai hộp này về mặt ngữ nghĩa là liên quan và tương đương nhau (ví dụ, hai tập giống hệt nhau của siêu dữ liệu được biểu thị trong hai hệ thống siêu dữ liệu khác nhau).

### Các dạng URL của Hộp Siêu dữ liệu

Khi một Hộp Siêu dữ liệu được sử dụng, khi đó các URL có thể được sử dụng để tham chiếu các hạng mục trong Hộp Siêu dữ liệu bằng cách sử dụng một URL tuyệt đối hoặc URL tương đối. Các URL tuyệt đối chỉ có thể được sử dụng để tham chiếu tới các hạng mục trong Hộp Siêu dữ liệu cùng mức tệp tin.

Khi thông dịch dữ liệu trong bối cảnh của một Hộp Siêu dữ liệu (ví dụ như tệp tinchứa Hộp Siêu dữ liệu cùng mức tệp tin, bản trình diễncủa Hộp Siêu dữ liệu cùng mức tệp tin, hoặc rãnh của Hộp Siêu dữ liệu cùng mức tệp tin), các hạng mục trong Hộp Siêu dữ liệuđược xử lý như các tệp tinche chắn trong cùng một vị trí giống nhau mà từ đó bắt đầu các hộp chứa tệp tin. Sự che chắn này nghĩa là tham chiếu tới các tệp tin khác nhau trong cùng vị trí vì tệp tinchứa có thể được xử lý thành một hạng mục trong chính tệp tin chứa đó. Các hạng mục có thể được chỉ ra trongtệp tin chứa bằng gắn một phân mảnh vào URL của chính tệp tin chứa. Phân mảnh này bắt đầu với kí tự “#” và bao gồm cả:

1. *item\_ID =<n>,* nhận biết hạng mục thông qua ID của nó (ID có thể bằng 0 đối với tài nguyên cơ bản);
2. *item\_name =<item\_name>,* khi hộp thông tin hạng mục được sử dụng.

Nếu một phân mảnhđoạn trong hạng mục chứa cần được xử lý thì kí tự “#” khởi tạo của phân mảnh đó được thay thế bởi “\*”.

Hãy xem xét ví dụ sau đây: <[http://a.com/d/v.qrv#*item\_name*=tree.html\*branch1](http://a.com/d/v.qrv#item_name=tree.html*branch1)>.Giả thiết v.qrv là một tệp tin với một Hộp Siêu dữ liệu ở mức tệp tin. Đầu tiên bên nhận sẽ bỏ phân mảnh và lấy v.qrv từ a.com sử dụng HTTP. Sau đó nó kiểm tra Hộp Siêu dữ liệumức cao nhất và bổ sung các hạng mục này vào bộ đểm của nó trong thư mục “d “trên a.com. Sau đó lại tạo lại URL như sau <<http://a.com/d/tree.html#branch1>>. Lưu ý rằng phân đoạn được tăng cường tạo nên tên tệp tin đầy đủ và “\*” xuất hiện đầu tiên được biến đổi trở lại thành “#”. Bên nhận sau đó sẽ tìm một mục được đặt tên tree.html trong Hộp Siêu dữ liệu, hoặc lấy tree.html từ a.com và sau đó nó tìm neo “branch1“trong tree.html. Nếu trong html, một tệp tin được tham chiếu bằng cách sử dụng URL tương đối, ví dụ.”flower.gif“, khi đó bên nhận sẽ chuyển đổi thành một URL tuyệt đối sử dụng các quy tắc thông thường: <http://a.com/d/flower.gif> và một lần nữa kiểm tra xem có phải flower.gif là một hạng mục được đặt tên (và vì vậy che chắn một tệp tin riêng biệt của tên này) và sau đó nếu không sẽ phải là lấy flower.giftừ a.com.

### Siêu dữ liệu tĩnh (Static Metadata)

Mục này xác định siêu dữ liệu tĩnh (không được định thời)trong họ định dạng tệp tin ISO.

Việc hỗ trợ đọc siêu dữ liệu thường là một tùy chọn và do đó nó cũng là tùy chọnđối với các định dạng xác định ở đây hoặc ở một chỗ nào khác, trừ khi là chọn lựa bắt buộc của một tiêu chuẩn kỹ thuật dẫn xuất.

#### Văn bản đơn giản

Việc hỗ trợ các thẻ văn bản đơn giản tồn tại dưới dạng các hộp dữ liệu người dùng; hiện tại chỉ có mộtthẻ văn bản duy nhất là thẻ thông tin bản quyền. Siêu dữ liệu khác được phép sử dụng dạng đơn giản này nếu:

1. Nó sử dụng một kiểu hộp đã được đăng ký hoặc UUID (hiện nay đã được sử dụng);
2. Nó sử dụng một thẻ đã được đăng ký, cấu trúc MPEG-7 tương đương phải được ghi nhận như một phần của việc đăng ký.

#### Các dạng khác nhau

Khi các dạng khác nhau của siêu dữ liệu đượcyêu cầu, khi đó một hộp ‘*meta*’ như được định nghĩa ở trên có thể bao gồm mức độ phù hợp của tài liệu này. Nếu tài liệu này có thể dùng chủ yếu như là một tài liệu siêu dữ liệu khi đó Hộp Siêu dữ liệu ở mức tệp tin. Nếu siêu dữ liệu chú giải cho toàn bộ bản trình diễn thì Hộp Siêu dữ liệuở mức movie; toàn bộ luồngở mức rãnh.

#### Siêu dữ liệu MPEG-7

Siêu dữ liệu MPEG-7 được lưu trong cácHộp Siêu dữ liệu cho tiêu chuẩn này.

1. Kiểu xử lý là ‘mp7t’ đối với siêu dữ liệu văn bản theo định dạng Unicode;
2. Kiểu xử lý là ‘mp7b’ đối với siêu dữ liệu nhị phân được nén theo định dạng BIM. Trong trường hợp này, hộp nhị phân XML chứa thông tin cấu hình ngay theo ngay sau bởi XML được nhị phân hóa.
3. Khi định dạng là dạng văn bản, tồn tại một hộp khác trong hộp chứa siêu dữ liệu ‘*meta*’, được gọi là ‘*xml*’, hộp này chứa tài liệu MPEG-7 nhị phân, hoặc có một hộp mục cơ bản nhận biết mục chứa MPEG-7 XML.
4. Khi định dạng là nhị phân, một trong các hộp khác nhau trong hộp chứa siêu dữ liệu ‘*meta*’, được gọi là ‘*bxml*’, trong đó có tài liệu nhị phân MPEG-7, hoặc hộp hạngmục cơ bảnxác định hạng mục chứa MPEG-7 được nhị phân hóa XML.
5. Nếu một hộp MPEG-7 được sử dụng ở mức tệp tin, khi đó nhãn ‘mp71’ phải là một thành viên của danh sách các nhãn có khả năng tương thích trong hộpkiểutệp tin.

### Hộp Dữ liệu Hạng mục (Item Data Box)

#### Định nghĩa

Loại hộp: ‘*idat*’.

Thuộc đối tượng:Hộp Siêu dữ liệu (‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này chứa dữ liệu của các hạng mục siêu dữ liệu sử dụng phương thức có cấu trúc để thể hiện các mở rộng dữ liệu của hạng mục được lưu trong hộp này.

#### Cú pháp

*aligned(8) class ItemDataBox extends Box(‘idat’) {*

*bit(8) data[];*

*}*

#### Ngữ nghĩa

*data*là dữ liệu của siêu dữ liệu.

### Hộp Tham chiếuHạng mục (Item Reference Box)

#### Định nghĩa

Loại hộp: ‘*iref*’.

Thuộc đối tượng: Hộp Siêu dữ liệu (‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Tham chiếu Hạng mục cho phép liên kết một hạng mục với các hạng mục khác thông qua các tham chiếu được phân kiểu. Tất cả các tham chiếu của một hạng mục của một kiểu cụ thể được một hộp tham chiếu kiểu hạng mục đơn nhất, tham chiếu có trường ‘from item ID’ thể hiện hạng mục được liên kết. Các hạng mục được liên kết đó được thể hiện thông qua một mảng các giá trị ‘to item ID’. Tất cả các hộp tham chiếu kiểu hạng mục đơn nhất này được tập hợp trong hộp tham chiếu hạng mục. Các kiểu tham chiếu được chỉ ra trong hộp tham chiếu rãnh trong 8.3.3 (hoặc các kiểu tham chiếu được đăng ký khác) có thể được sử dụng ở đây nếu phù hợp. Phiên bản 1 của ItemReferenceBox với SingleItemReferenceBoxLarge chỉ nên sử dụngkhi trường *from*\_*item\_ID* hoặc *to\_item\_ID* có giá trị lớn (trên 65535).

LƯU Ý:Kiểu thiết kế này khiến việc tìm kiêm tất cả các tham chiếu của một kiểu cụ thể, hoặc từ một hạng mục cụ thể rất dễ dàng.

#### Cú pháp

*aligned(8) class SingleItemTypeReferenceBox(referenceType) extends Box(referenceType) {*

*unsigned int(16) from\_item\_ID;*

*unsigned int(16) reference\_count;*

*for (j=0; j<reference\_count; j++) {*

*unsigned int(16) to\_item\_ID;*

*}*

*}*

*aligned(8) class SingleItemTypeReferenceBoxLarge(referenceType) extends Box(referenceType) {*

*unsigned int(32) from\_item\_ID;*

*unsigned int(16) reference\_count;*

*for (j=0; j<reference\_count; j++) {*

*unsigned int(32) to\_item\_ID;*

*}*

*}*

*aligned(8) class ItemReferenceBox extends FullBox(‘iref’, version, 0) {*

*if (version==0) {*

*SingleItemTypeReferenceBox references[];*

*} else if (version==1) {*

*SingleItemTypeReferenceBoxLarge references[];*

*}*

*}*

#### Ngữ nghĩa

*reference\_type*chứa kiểu tham chiếu.

*from\_item\_ID*chứa ID của các hạng mục tham chiếu tới các hạng mục khác.

*reference\_count*chỉ số lượng các tham chiếu.

*to\_item\_ID*chứa ID của hạng mục được tham chiếu tới.

### Siêu dữ liệu hình ảnh bổ trợ

Một rãnh hình ảnh bổ trợ được sử dụng để chỉ thông tin về độ sâu hoặc thông tin thị sai có thể chứa một hạng mục siêu dữ liệu kiểu ‘auvd’ (miêu tảhình ảnhbổ trợ); dữ liệu của hạng mục này chính xác là một si\_rbsp() như được chỉ ra trong ISO/IEC 23002-3. (Lưu ý rằng si\_rbsp() này được đóng khung bên ngoài và độ dài được cung cấp bởi thông tin vị trí hạng mục trong định dạng tệp tin này). Có thể có nhiều hơn một hạng mục siêu dữ liệu (ví dụ một cho thông tin thị sai và một cho thông tin độ sâu, trong trường hợp cùng một luồng).

## Hỗ trợ các luồng được bảo vệ

Mục này thể hiện việc chuyển đổi định dạng tệp tin được sử dụng cho nội dung được bảo vệ. Việc chuyển đổi này có thể được sử dụng trong một số trường hợp:

* Chúng phải được sử dụng khi nội dung được chuyển đổi (ví dụ bằng cách mã hóa) theo cách mà nó không thể được giải mã bởi bộ giải mã thông thường;
* Chúng có thể được sử dụng khi nội dung chỉ được giải mã khi hệ thống bảo vệ hiểu và thực hiện được.

Các chức năng biến đổi bằng cách đóng gói các khai báo đa phương tiện gốc. Việc đóng gói thay đổi mã 4 kí tự của các đầu vào mục mẫu, vì vậy các thiết bị đọc không biết việc bảo vệ sẽ hiểu luồng đa phương tiện như một định dạng luồng mới.

Bởi vì định dạng của một đầu vào mục mẫubiến đổi theo kiểuđa phương tiện nên một mã 4 ký tự đóng gói khác nhau sẽ được sử dụng cho mỗi kiểuđa phương tiện (âm thanh, hình ảnh, văn bản…). Đó là:

|  |  |
| --- | --- |
| **Kiểu luồng (Kiểu rãnh)** | **Mã Sample-Entry** |
| Hình ảnh | encv |
| Âm thanh | enca |
| Văn bản | enct |
| Hệ thống | encs |

Việc chuyển đổi của một mô tả mẫu được miêu tả bởi các thủ tục sau đây:

1. Mã bốn kí tự của miêu tả mẫu được thay thế bởimã bốn kí tự thể hiện việc đóng gói được bảo vệ: các mã này chỉ thay đổi theo kiểuđa phương tiện. Ví dụ, *‘mp4v’*được thay bởi*‘encv’* và *‘mp4a’*được thay bởi*‘enca’*.
2. Một ProtectionSchemeInfoBox (định nghĩa bên dưới) được bổ sung vào miêu tả mẫu, tất cả các hộp khác không bị thay đổi.
3. Kiểuđầu vào mục mẫu(mã bốn kí tự) được lưu trong ProtectionSchemeInfoBox nằm trong một hộp mới gọi là OriginalFormatBox (được định nghĩa bên dưới);

Có ba phương pháp để cho báo hiệu sự tồn tại của việc bảo vệ, có thể được sử dụng riêng hoặc kết hợp.

1. Khi hệ thống MPEG-4 được sử dụng thì IPMP phải được sử dụng để báo rằng các luồng này được bảo vệ.
2. Các bộ miêu tả IPMP cũng có thể được sử dụng bên ngoài hệ thống MPEG-4 sử dụng các hộp chứa các bộ miêu tả IPMP.
3. Việc bảo vệ được sử dụng cũng có thể được miêu tả sử dụng các hộp thông tin và các hộp kiểu lược đồ.

Khi IPMP được sử dụng bên ngoài hệ thống MPEG-4, khi đó một IPMPControlBox ‘global’cũng có thể xuất hiện trong ‘*moov*’.

LƯU Ý: khi hệ thống MPEG-4 được sử dụng, một thiết bị đầu cuối hệ thống MPEG-4 có thể xử lý một cách hiệu quả, ví dụ, *‘encv’* với một định dạng ban đầu của *‘mp4v’*giống như *‘mp4v’*, bằng cách sử dụng các bộ miêu tả IPMP.

### Hộp Thông tin Lược đồ Bảo vệ (Protection Scheme Information Box)

#### Định nghĩa

Loại hộp:‘*sinf*’.

Thuộc đối tượng: Hộp Đầu vào mục MẫuBảo vệ, hoặc Hộp Bảo vệ Hạng mục (‘*ipro*).

Hình thức sử dụng: bắt buộc.

Số lượng: một hoặc nhiều.

Hộp Thông tin Lược đồ Bảo vệ chứa tất cả thông tin cần thiết để hiểu việc chuyển đổi mã hóa được sử dụng và các tham số của nó, cũng như để tìm kiếm các thông tin khác như kiểu và vị trí của hệ thống quản lý chính. Hộp này cũng lưu định dạng gốc (chưa mã hóa) của đa phương tiện. Hộp Thông tin Lược đồ Bảo vệlà một hộp chứa. Bắt buộc trong một đầu vào mục mẫuphải sử dụng một mã cho biết một luồng được bảo vệ.

Khi sử dụng trong một đầu vào mục mẫuđược bảo vệ, hộp này phải chứa hộp định dạng gốc để ghi lại định dạng ban đầu. Ít nhất một trong các phương pháp báo hiệu sau phải được sử dụng để nhận biết việc bảo vệ được áp dụng:

1. Các hệ thống MPEG-4 với IPMP: không có hộp nào khác khi bộ miêu tảIPMP trong các hệ thống MPEG-4 được sử dụng;
2. Lược đồ báo hiệu: một SchemeTypeBox và SchemeInformationBox, khi chúng được sử dụng (hoặc cả hai đều phải tồn tại, hoặc không).

Ít nhất một hộp thông tin lược đồ bảo vệphải tồn tại trong đầu vào mục mẫubảo vệ. Khitồn tại nhiều hơn một hộp thì chúng tương đương, thay thế, miêu tả sự bảo vệ giống nhau. Thiết bị đọc có thể chọn một để xử lý.

#### Cú pháp

*aligned(8) class ProtectionSchemeInfoBox(fmt) extends Box('sinf') {*

*OriginalFormatBox(fmt)original\_format;*

*SchemeTypeBox scheme\_type\_box;// optional*

*SchemeInformationBoxinfo;// optional*

}

### Hộp Định dạng Gốc (Original Format Box)

#### Định nghĩa

Loại hộp: ‘*frma*’.

Thuộc đối tượng :Hộp Thông tin Lược đồ Bảo vệ (‘*sinf*’),Hộp Thông tin Lược đồ Giới hạn (*‘rinf’*), Hộp Thông tin Rãnh Hoàn chỉnh (‘cinf’).

Hình thức sử dụng: bắt buộc khi sử dụng trong đầu vào mục mẫu bảo vệ, trong đầu vào mục mẫu giới hạn, hoặc trong đầu vào mục mẫu của một rãnh không hoàn chỉnh.

Số lượng: một.

Hộp Định dạng Gốc ‘*frma*’chứa mã gồm bốn kí tự của miêu tả mẫu gốc không chuyển đổi.

#### Cú pháp

*aligned(8) class OriginalFormatBox(codingname) extends Box ('frma') {*

*unsigned int(32)data\_format = codingname;*

*// format of decrypted, encoded data (in case of protection)*

*// or un-transformed sample entry (in case of restriction*

*// and complete track information)*

*}*

#### Ngữ nghĩa

*data\_format*là mã bốn ký tự của đầu vào mục mẫukhông chuyển đổi gốc (ví dụ,”mp4v“nếu luồng đó chứa tư liệu hình ảnh MPEG-4 được bảo vệ hoặc được giới hạn).

### Hộp thông tin IPMP (IPMP Info Box)

Mục trống.

### Hộp Điều khiển IPMP (IPMP Control Box)

Mục trống.

### Hộp Kiểu Lược đồ (Scheme Type Box)

#### Định nghĩa

Loại hộp: ‘*schm*’.

Thuộc đối tượng: Hộp Thông tin Lược đồ Bảo vệ (‘*sinf*’), Hộp Thông tin Lược đồ Giới hạn (*‘rinf’*), hoặc Hộp Xử lý SRTP (‘srpp’).

Hình thức sử dụng: không bắt buộc.

Số lượng:

* không hoặc một (đối với ‘*sinf*’, phụ thuộc vào cấu trúc bảo vệ).
* một(đối với *‘rinf’* và ‘srpp’).

Hộp Kiểu Lược đồ (‘*schm*’) xác định lược đồ bảo vệ hoặc lược đồ giới hạn.

#### Cú pháp

*aligned(8) class SchemeTypeBox extends FullBox('schm', 0, flags) {*

*unsigned int(32)scheme\_type;// 4CC nhận biết lược đồ*

*unsigned int(32)scheme\_version; // phiên bản lược đồ*

*if (flags & 0x000001) {*

*unsigned int(8)scheme\_URI[];*

*}*

*}*

#### Ngữ nghĩa

*scheme\_type* là mã xác định lược đồ bảo vệ hoặc giới hạn.

*scheme\_version*là phiên bản của lược đồ (được sử dụng để tạo ra nội dung).

*scheme\_URI*cho phép tùy chọn hướng người dùng tới một trang web nếu chúng không có lược đồ được thiết lập trên hệ thống của chúng. Đó là một URI tuyệt đối được tạo ra như một chuỗi kí tự UTF-8 kết thúc bởi kí tự trống.

### Hộp Thông tin Lược đồ (Scheme Information Box)

#### Định nghĩa

Loại hộp: ‘*schi*’.

Thuộc đối tượng: Hộp Thông tin Lược đồ Bảo vệ (‘*sinf*’), Hộp Thông tin Lược đồ Giới hạn(*‘rinf’*), hoặc Hộp Xử lý SRTP (‘srpp’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Thông tin Lược đồ là hộp chứa được thông dịch bởi lược đồ đang được sử dụng. Mọi thông tin mà hệ thống mã hóa hoặc hệ thống giới hạn cần được lưu ở đây. Nội dung của hộp này là một chuỗi các hộp có kiểu và định dạng được xác định bởi lược đồ được khai báo trong Hộp KiểuLược đồ.

#### Cú pháp

*aligned(8) class SchemeInformationBox extends Box('schi') {*

*Box scheme\_specific\_data[];*

*}*

## Hỗ trợ định dạng vận chuyển tệp tin

### Giới thiệu

Các tệp tin dành cho truyền trên ALC/LCT hoặc FLUTE được lưu trữ như các hạng mục trong Hộp Siêu dữ liệu (‘*meta*’) mức cao. Hộp vị trí hạng mục (‘*iloc*’) xác định vị trí lưu trữ thực tế của mỗi hạng mục trong tệp tin chứa cũng như kích thước tệp tin của mỗi hạng mục. Tên tệp tin, kiểu nội dung (kiểu MIME),... của mỗi hạng mục được cung cấp bởi phiên bản 1 của hộp thông tin hạng mục (‘*iinf*’).

Các dự trữ FEC đượctính trước được lưu trữ như các hạng mục bổ sung trong Hộp Siêu dữ liệu. Nếu tệp tin nguồn được chia thành một vài khối nguồn, các dự trữ FEC cho mỗi khối nguồn được lưu trữ như các hạng mục riêng biệt. Mối liên hệ giữa các dự trữ FEC và các hạng mục nguồn ban đầu được ghi lại trong hộp đầu vào phân vùng (*‘paen’*) nằm trong hộp thông tin hạng mục FD (*‘fiin’*).

Các dự trữ tệp tin được thiết lập trước được lưu như là các hạng mục bổ sung trongtệp tin chứa. Nếu một tệp tinnguồn được phân thành một vài khối nguồn, mỗi khối nguồn sẽ được lưu như một hạng mục riêng gọi dự trữ tệp tin. Mối liên hệ giữa dự trữtệp tin và các hạng mục nguồn ban đầu được ghi lại trong hộp đầu vào phân vùng (*‘paen’*) nằm trongHộp Thông tin Hạng mục FD (*‘fiin’*).

Xem mục 9.2 để biết thêm chi tiết vê việc sử dụng định dạng vận chuyển tệp tin.

### Hộp Thông tin Hạng mục FD (FD Item Information Box)

#### Định nghĩa

Loại hộp:*‘fiin’*.

Thuộc đối tượng: Hộp Siêu dữ liệu (‘*meta*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Thông tin Hạng mục FD là tùy chọn, mặc dù nó là bắt buộc đối với các tệp tin sử dụng FD rãnh chỉ dẫn. Hộp này cung cấp thông tin về phân vùng của các tệp tin nguồn và cách thức các rãnh chỉ dẫnFD kết hợp lại thành các phiên FD. Mỗi đầu vào mục phân vùngcung cấp chi tiết về một phân vùng tệp tincụ thể, về việc mã hóa FEC, tệp tin liên kết và các dự trữ FEC. Có thể cung cấp nhiều đầu vào mục cho một tệp tin nguồn (nhận biết bởi ID hạng mục của nó) nếu các lược đồ mã hóa FEC thay thế hoặc các phân vùng được sử dụng trong tệp tin. Tất cả các đầu vào mục phân vùngđược đánh số và đầu vào mục đầu tiên có giá trị 1.

#### Cú pháp

*aligned(8) class PartitionEntry extends Box(‘paen’) {*

*FilePartitionBoxblocks\_and\_symbols;*

*FECReservoirBox FEC\_symbol\_locations; //optional*

*FileReservoirBoxFile\_symbol\_locations; //optional*

*}*

*aligned(8) class FDIteminformationBox*

*extends FullBox(‘fiin’, version = 0, 0) {*

*unsigned int(16)entry\_count;*

*PartitionEntrypartition\_entries[ entry\_count ];*

*FDSessionGroupBox session\_info;//optional*

*GroupIdToNameBoxgroup\_id\_to\_name; //optional*

*}*

#### Ngữ nghĩa

*entry\_count* chỉ số lượng đầu vào mục trong mảng sau đây.

Ngữ nghĩa của các hộp được miêu tảtại các hộp được ghi nhận.

### Hộp Phân vùng Tệp tin (File Partition Box)

#### Định nghĩa

Loại hộp:*‘fpar’*.

Thuộc đối tượng: Hộp Đầu vào mục Phân vùng (*‘paen’*).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp Phân vùng Tệp tin xác địnhtệp tin nguồn và cung cấp một phân vùng của tệp tin đó trong các khối nguồn và các biểu tượng. Thông tin về tệp tin nguồn, ví dụ,tên tệp tin, vị trí của nội dung và ID của các nhóm được lưu trong Hộp Thông tin Hạng mục (‘*iinf*’), trong đó đầu vào mục thông tin hạng mục tương ứng với ID hạng mục trong tệp tin gốc của phiên bản 1 và bao gồm cả Mở rộng Thông tin Hạng mục Phân phối Tệp tin (‘fdel’). Phiên bản 1 của Hộp Phân vùng Tệp tin chỉ được sử dụng khi cần *item\_ID* hoặc *entry\_count* có giá trị lớn (trên 65535).

#### Cú pháp

*aligned(8) class FilePartitionBox extends FullBox(‘fpar’, version, 0) {*

*if (version == 0) {*

*unsigned int(16) item\_ID;*

*} else {*

*unsigned int(32) item\_ID;*

*}*

*unsigned int(16) packet\_payload\_size;*

*unsigned int(8)reserved = 0;*

*unsigned int(8) FEC\_encoding\_ID;*

*unsigned int(16) FEC\_instance\_ID;*

*unsigned int(16) max\_source\_block\_length;*

*unsigned int(16) encoding\_symbol\_length;*

*unsigned int(16) max\_number\_of\_encoding\_symbols;*

*string scheme\_specific\_info;*

*if (version == 0) {*

*unsigned int(16) entry\_count;*

*} else {*

*unsigned int(32) entry\_count;*

*}*

*for (i=1; i <= entry\_count; i++) {*

*unsigned int(16) block\_count;*

*unsigned int(32) block\_size;*

*}*

*}*

#### Ngữ nghĩa

*item\_ID*tham chiếu hạng mục trong hộp vị trí mục (‘*iloc*’) mà phân vùng tệp tin được áp dụng.

*packet\_payload\_size* cho biết kích thước tải trọng gói ALC/LCT hoặc FLUTE mục tiêu của thuật toán phân vùng. Lưu ý rằng các tải trọng gói UDP là lớn hơn vì chúng cũng chứa các mào đầu ALC/LCThoặc FLUTE.

*FEC\_encoding\_ID*xác định lược đồ mã hóa FEC và là đối tượng của việc đăng ký IANA (xem RFC 5052). Lưu ý rằng i) giá trị là không tương đương với “Compact No-Code FEC“cũng được hiểu là “Null-FEC “(RFC 3695); ii) giá trị một tương đương với “MBMS FEC “(3GPP TS 26.346); iii) với giá trị trong dải từ 0-127, bao gồm, sơ đồ FEC được thiết lập đầy đủ, trái ngược đối với giá trị trong dải từ 128 -255,bao gồm sơ đồ FEC được thiết lập bên dưới.

*FEC\_instance\_ID*cung cấp một nhận dạng cụ thể của bộ mã hóa FEC đang được sử dụng cho một sơ đồ Under-Specified FEC. Giá trị này sẽ sét tới không cho sơ đồ Fully-Specified FEC và sẽ được bỏ qua khi phân tích một tệp tin với FEC\_encoding\_ID nằm trong khoảng từ 0-127. FEC\_instance\_ID nằm trong phạm vi của FEC\_encoding\_ID. Xem RFC 5052 để biết thêm nhiều chi tiết.

*max\_source\_block\_length*cho biết số lượng tối đa của các biểu tượng nguồn cho mỗi khối nguồn.

*encoding\_symbol\_length*cho biết kích thước (theo byte) của một kí hiệu mã hóa. Tất cả các kí hiệu mã hóa của một mục có độ dài giống nhau, ngại trừ kí hiệu cuối có thể ngắn hơn.

*max\_number\_of\_encoding\_symbols* cho biết số lượng tối đa của các kí hiệu mã hóa mà có thể được tạo ra cho một khối nguồn đối với các sơ đồ FEC trong đó số lượng tối đa của các kí tự mã hóa là liên quan với nhau như mã hóa FEC ID 129 được định nghĩa trong RFC 5052. Đối với các sơ đồ FEC trong đó số lượng tối đa của các kí hiệu mã hóa là không liên quan với nhau, về mặt ngữ nghĩa trường này không được chỉ rõ.

*scheme\_specific\_info*là một chuỗi mã hóa 64 bit kết thúc bởi ký tự trống của đối tượng truyền thông tin sơ đồ cụ thể (FEC-OTI-Scheme-Specific-Info). Định nghĩa của thông tin này tùy thuộccủa vào ID mã hóa FEC.

*entry\_count* cho biết số lượng entries trong danh sách (block\_count, block\_size) cặp cung cấp một phân vùng của tệp tin nguồn. Bắt đầu từ điểm đầu của tệp tin, mỗi đầu vào mục chỉ ra cách đoạn kế tiếp của tệp tin được chia thành các khối nguồn và các kí hiệu nguồn.

*block\_count*chỉ ra kích thước block\_size của số khối nguồn liên tiếp.

*block\_size*chỉ ra kích thước của một khối (theo byte). Một block\_size không phải là bội số của kích thướckí hiệu encoding\_symbol\_length cho biết với CompactNo-CodeFEC rằng các kí hiệu nguồn cuối cùng gồm đệm mà không được lưu trữ trong mục. Với MBMS FEC (3GPP TS 26.346) việc đệm có thể được mở rộng trên nhiều kí tự nhưng kích thước đệm không bao giờ nhiều hơn encoding\_symbol\_length.

### Hộp Dự trữ FEC (FEC Reservoir Box)

#### Định nghĩa

Loại hộp: ‘*fecr*’.

Thuộc đối tượng: Đầu vào mục Phân vùng (*‘paen’*).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Dự trữ FEC liên kết tệp tin nguồn được xác định trong Hộp Phân vùng Tệp tin (*‘fpar’*) với dự trữ FEC được lưu như các hạng mục bổ sung. Hộp này chứa một danh sách bắt đầu với dự trữ FEC đầu tiên được liên kết với khối nguồn đầu tiên của tệp tin nguồn và tiếp tục như vậy đối với các khối nguồn khác trong tệp tin nguồn. Phiên bản 1 của Hộp Dự trữ FEC chỉ nên được dùng khi cần *item\_ID* và *entry\_count* có giá trị lớn (trên 65535).

#### Cú pháp

*aligned(8) class FECReservoirBox extends FullBox('fecr', version, 0) {*

*if (version == 0) {*

*unsigned int(16) entry\_count;*

*} else {*

*unsigned int(32) entry\_count;*

*}*

*for (i=1; i <= entry\_count; i++) {*

*if (version == 0) {*

*unsigned int(16) item\_ID;*

*} else {*

*unsigned int(32) item\_ID;*

*}*

*unsigned int(32) symbol\_count;*

*}*

*}*

#### Ngữ nghĩa

*entry\_count* cho biết số lượng của các đầu vào mục trong danh sách sau. Một số đếm đầu vào mục ở đây cần phù hợp với tổng số các khối trong phân vùng tệp tin tương ứng của hộp.

*item\_ID*chỉ ra vị trí của nguồn cung cấp FEC được liên kết với một khối nguồn.

*symbol\_count* chỉ ra chỉ số của các biểu tượng sửa chữa được chứa trong nguồn cung cấp FEC.

### Hộp Nhóm Phiên FD (FD Session Group Box)

#### Định nghĩa

Loại hộp: ‘*segr*’.

Thuộc đối tượng: Hộp thông tin FD (*‘fiin’*).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp nhóm phiên FD là một tùy chọn, mặc dù nó là bắt buộc cho các tệp tin chứa nhiều hơn một FD rãnh chỉ dẫn. Nó chứa một danh sách của các phiên cũng như tất cả các nhóm tệp tin và các rãnh chỉ dẫnmà cùng mỗi phiên. Một phiên FD gửi đồng thời trên tất cả các FD rãnh chỉ dẫn(các kênh) mà được liệt kê theo phiên FD trong hộp nhóm cho một phiên FD cụ thể.

Tại mọi thời điểm chỉ một nhóm phiên được xử lý. Trước hết liệt kê rãnh chỉ dẫntrong một nhóm phiên được chỉ định cho kênh cơ bản. Nếu máy chủ không có ưu tiên giữa các nhóm phiên, sự lựa chọn mặc định sẽ là nhóm phiên đầu tiên. Các ID nhóm của tất cả các nhóm tệp tin chứa các tệp tin được tham chiếu bởi các rãnh chỉ dẫnsẽ được bao gồm trong danh sách của các nhóm tệp tin. Các ID nhóm tệp tin có thể lần lượt được chuyển đổi thành tên nhóm tệp tin (sử dụng ID nhóm cho tên hộp) có thể được bao gồm bởi máy chủ trong các FDT.

#### Cú pháp

*aligned(8) class FDSessionGroupBox extends Box('segr') {*

*unsigned int(16)num\_session\_groups;*

*for(i=0; i <num\_session\_groups; i++) {*

*unsigned int(8) entry\_count;*

*for (j=0; j < entry\_count; j++) {*

*unsigned int(32)group\_ID;*

*}*

*unsigned int(16) num\_channels\_in\_session\_group;*

*for(k=0; k < num\_channels\_in\_session\_group; k++) {*

*unsigned int(32) hint\_track\_id;*

*}*

*}*

*}*

#### Ngữ nghĩa

*num\_session\_groups*chỉ rõ số nhóm phiên.

*entry\_count* cho biết số lượng của các entries trong danh sách kế tiếp bao gồm tất cả các nhóm tệp tin mà nhóm phiên tuân theo. Nhóm phiên này chứa tất cả các tệp tin được bao gồm trong danh sách các nhóm tệp tin như được chỉ rõ bởi thông tin mục đầu vào mục của mỗi tệp tin nguồn. Lưu ý rằng FDT cho nhóm phiên chỉ chứa các nhóm được liệt kê trong cấu túc này.

*group\_ID* cho biết một nhóm tệp tin mà nhóm phiên tuân theo. num\_channels\_in\_session\_groupsquy định cụ thể số kênh trong nhóm phiên. Giá trị của num\_channels\_in\_session\_groups sẽ có kiểu số nguyên dương.

*hint\_track\_ID*quy định rãnh ID của FD rãnh chỉ dẫnthuộc về một nhóm phiên cụ thể. Lưu ý rằng một rãnh chỉ dẫntương ứng cho một kênh LCT.

### Hộp Ánh xạ Tên và Nhóm định danh (Group ID to Name Box)

#### Định nghĩa

Loại hộp: ‘*gitn*’.

Thuộc đối tượng: Hộp thông tin FD (*‘fiin’*).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp này thể hiện mối quan hệ giữa tên nhóm của tệp tin với định danh nhóm của tệp tin, được sử dụng trong phiên bản 1 của các mục thông tin trong Hộp Thông tin Mục (‘*iinf*’).

#### Cú pháp

*aligned(8) class GroupIdToNameBox extends FullBox('gitn', version = 0, 0) {*

*unsigned int(16)entry\_count;*

*for (i=1; i <= entry\_count; i++) {*

*unsigned int(32)group\_ID;*

*stringgroup\_name;*

*}*

*}*

#### Ngữ nghĩa

*entry\_count* cho biết số entries trong danh sách tiếp theo.

*group\_ID* chỉ ra một nhóm tệp tin.

*group\_name* là một chuỗi kí tự UTF-8 được kêt thúc bởi kí tự trống bao chứa tên một nhóm tệp tin.

### Hộp Dự trữ Tệp tin (File Reservoir Box)

#### Định nghĩa

Loại hộp: ‘*fire*’.

Thuộc đối tượng: Đầu vào mục Phân vùng (*‘paen’*).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp Dự trữ Tệp tin liên kết tệp tin nguồn được xác định trong Hộp Phân vùng Tệp tin (*‘fpar’*) với dự trữ FEC được lưu như các hạng mục bổ sung. Hộp này chứa một danh sách bắt đầu với dự trữ tệp tin đầu tiên được liên kết với khối nguồn đầu tiên của tệp tin nguồn và tiếp tục như vậy đối với các khối nguồn khác trong tệp tin nguồn. Phiên bản 1 của Hộp Dự trữ Tệp tin chỉ nên được dùng khi cần *item\_ID* và *entry\_count* có giá trị lớn (trên 65535).

#### Cú pháp

*aligned(8) class FileReservoirBox extends FullBox('fire', version, 0) {*

*if (version == 0) {*

*unsigned int(16) entry\_count;*

*} else {*

*unsigned int(32) entry\_count;*

*}*

*for (i=1; i <= entry\_count; i++) {*

*if (version == 0) {*

*unsigned int(16) item\_ID;*

*} else {*

*unsigned int(32) item\_ID;*

*}*

*unsigned int(32) symbol\_count;*

*}*

*}*

#### Ngữ nghĩa

*entry\_count* đem lại số lượng các bản ghi trong danh sách sau đây. Một số lượng bản ghi ở đây sẽ phù hợp với tổng số block trong hộp phân vùng tệp tin tương ứng.

*item\_ID* chỉ vị trí của Bộ chứa tệp tin được kết hợp với một khối nguồn.

*symbol\_count* chỉ số lượng các biểu tượng nguồn được chứa trong Bộ chứa tệp tin**.**

## Rãnh con

### Giới thiệu

Các rãnh con được sử dụng để chỉ định các phần của rãnh cho luân phiên và chuyển mạch các nhóm theo cùng một cách giống như các rãnh có thể được gán để luân phiên và chuyển mạch nhóm, để chỉ ra không biết các rãnh đó có luân phiên tới từng rãnh khác và không biết chúng có Ngữ nghĩa cho chuyển mạch trong suốt một phiên không. Các rãnh con có thể phù hợp cho đa phương tiện theo lớp, ví dụ SVC và MVC, tại nơi đa phương tiện luân phiên thường là không tương đương với các cấu trúc rãnh. Bằng cách định nghĩa các nhóm luân phiên và chuyển mạch tại mức rãnh con, có thể sử dụng các luật đang tồn tại cho việc lựa chọn đa phương tiện và chuyển mạch đối với các codec theo dạng lớp. Cộng chung tất cả các cú pháp là chung nhất cho tất cả các kiểuđa phương tiện và tương thích ngược với các định nghĩa tại mức rãnh. Mức rãnh con luân phiên và chuyển mạch các nhóm sử dụng cùng số như các nhóm tại mức rãnh. Các số là toàn cục đối với tất cả các rãnh như là các nhóm có thể được định nghĩa thông qua các biên của rãnh và rãnh con.

Để định nghĩa các rãnh con, nhiều định nghĩa đặc tả đa phương tiện là bắt buộc. Các định nghĩa cho SVC và MVC được đặc tả trong định dạng tệp tin AVC (ISO / IEC 14.496-15). Một cách khác để định nghĩa các nhóm mẫu và ánh xạ chúng tới các rãnh con bằng cách sử dụng Hộp Nhóm Mẫu Rãnh con được định nghĩa ở đây. Cú pháp cũng có thể được mở rộng để bao gồm các định nghĩa đặc tả đa phương tiện khác.

Đối với từng rãnh con sẽ được định nghĩa một Hộp Rãnh con nằm trong Hộp Dữ liệu Người dùngcủa các rãnh tương ứng. Hộp rãnh con chứa các đối tượng được định nghĩa và cung cấp thông tin về một rãnh con trong cùng rãnh.Hộp rãnh concủa rãnh cũng được đặt tại đó.

### Khả năng tương thích ngược

Mặc định gán các nhóm luân phiên và chuyển mạch về 0 cho (toàn bộ) các rãnh, điều này có Ngữ nghĩa rằng không có thông tin trên các nhóm luân phiên và/hoặc chuyển mạch cho (toàn bộ) các rãnh đó. Tuy nhiên, các bộ đọc tệp tin nhận ra được các định nghĩa của rãnh con sẽ có thể tìm kiếm thông tin rãnh con tại các nhóm luân phiên và chuyển mạch thậm chí nếu cho biết rãnh được cài đặt về 0. Bằng cách này có thể chỉ ra rằng một tệp tin có thể được sử dụng bởi các bộ đọc truyền thống bằng cách bao gồm các kiểu thích hợp trong Hộp Kiểu Tệp tin. Một bộ tạo tệp tin mà yêu cầu một bộ đọc phải nhận ra thông tin rãnh con sẽ không bao gồm các nhánh truyền thống.

Cùng phương pháp gán thông tin cho rãnh con cũng có thể được áp dụng cho mọi thành phần của rãnh ngoại trừ một rãnh con thuộc cùng nhóm luân phiên hoặc chuyển mạch. Sau đó các định nghĩa tổng thể có thể được tạo nên tại mức rãnh thông thường và các đặc tả diễn giải có thể được tạo tại mức rãnh con. Đối với các rãnh con không có các đặc tả diễn giải, thì áp dụng theo mặc định. Như trước kia, nếu một bộ tạo tệp tin yêu cầu một đầu đọc phải hiểu thông tin rãnh con nếu nó không bao gồm các nhánh truyền thống (nếu không sẽ cho biết thông tin rãnh con có thể bị bỏ qua).

### Hộp Rãnh con (SubTrack Box)

#### Định nghĩa

Loại hộp: ‘*strk*’.

Thuộc đối tượng:Hộp Dữ liệu Người dùng (‘*udta*’) củaHộp Rãnh tương ứng (‘*trak*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp này chứa các đối tượng dùng để định nghĩa và cung cấp thông tin về một rãnh con trong rãnh hiện tại.

#### Cú pháp

*aligned(8) class SubTrack extends Box(‘strk’) {*

*}*

### Hộp Thông tin Rãnh con (SubTrackInformation Box)

#### Định nghĩa

Loại hộp: ‘*stri*’.

Thuộc đối tượng:Hộp Rãnhcon(‘*strk*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

#### Cú pháp

*aligned(8) class SubTrackInformationextends FullBox(‘stri’, version = 0, 0){*

*template int(16) switch\_group = 0;*

*template int(16) alternate\_group = 0;*

*template unsigned int(32) sub\_track\_ID = 0;*

*unsigned int(32) attribute\_list[]; // to the end of the box*

*}*

#### Ngữ nghĩa

*switch\_group*có kiểu số nguyên mà xác định một nhóm hoặc một tập hợp các rãnh và/hoặc các rãnh con. Nếu trường này là 0 (giá trị mặc định), sau đó không có thông tin vào việc rãnh con có thể được sử dụng để chuyển mạch trong suốt quá trình đọc hoặc streaming. Nếu số nguyên này không bằng 0 nó sẽ giống như các rãnh và/hoặc các rãnh con có thể được sử dụng để chuyển mạch giữa chúng. Các rãnh thuộc cùng nhóm chuyển mạch sẽ thuộc cùng nhóm luân phiên. Một nhóm chuyển mạch có thể chỉ có một thành viên.

*alternate\_group*có kiểu số nguyên xác định một nhóm hoặc tập hợp các rãnh và/hoặc các rãnh con. Nếu trường này là 0 (giá trị mặc định), thì không có thông tin trong các liên hệ có thể tới các rãnh khác hoặc các rãnh con khác. Nếu trường này khác 0, nó có cùng giá trị cho các rãnh hoặc rãnh con chứa dữ liệu luân phiên cho một rãnh khác và khác biệt đối với các rãnh hoặc các rãnh con thuộc về các nhóm khác. Chỉ một rãnh hoặc rãnh con trong một nhóm luân phiên sẽ được đọc hoặc phát đi tại một thời điểm bất kỳ.

*sub\_track\_ID* là một số integer. Giá trị duy nhất khác 0 xác định tính địa phương riêng của rãnh con trong *rãnh*. Giá trị 0 (mặc định) có nghĩa rằng ID của rãnh con không được gán.

attribute\_list là một danh sách các thuộc tính, tại cuối cùng của hộp. Các thuộc tính trong danh sách này sẽ được sử dụng như các miêu tả của các rãnh con hoặc sự phân biệt các tiêu chuẩn cho các rãnh và các rãnh con trong cùng nhóm luân chuyển hoặc chuyển mạch.

Các thuộc tính được miêu tả sau đây:

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Mã** | **Mô tả** |
| Khả năng mở rộng theo thời gian | *‘tesc’* | Rãnh con có thể được mở rộng theo thời gian |
| Khả năng mở rộng theo SNR tinh | ‘*fgsc’* | Rãnh con có thể được mở rộng về chất lượng |
| Khả năng mở rộng theo SNR thô | ‘*cgsc’* | Rãnh con có thể được mở rộng về chất lượng |
| Khả năng mở rộng theo không gian | ‘*spsc’* | Rãnh con có thể được mở rộng theo không gian |
| Khả năng mở rộng theo vùng ưa thích | ‘*resc’* | Rãnh con có thể được mở rộng theo vùng ưa thích |
| Khả năng mở rộng theo ảnh | ‘*vwsc’* | Rãnh con có thể được mở rộng về số lượng của các ảnh |

Các thuộc tính sau đây cần được phân biệt rõ:

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Mã** | **Mô tả** |
| Tốc độ bit | ‘*bitr’* | Tổng kích cỡ các mẫu trong rãnh được phân chia bởi thời gian trongHộp Mào đầu Rãnh |
| Tốc độ khung | ‘*frar’* | Số lượng các mẫu trong rãnh được phân chia bởi thời gian trongHộp Mào đầu Rãnh |
| Số lượng ảnh | ‘*nvws’* | Số lượng ảnh trong rãnh con |

### Hộp Định nghĩa Rãnh con (Sub Track Định nghĩa Box)

#### Định nghĩa

Loại hộp: ‘*strd*’.

Thuộc đối tượng:Hộp Rãnh con (‘*strk*’).

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp này chứa các đối tượng dùng để định nghĩa của rãnh con.

#### Cú pháp

*aligned(8) class SubTrackĐịnh nghĩa extends Box(‘strd’) {*

*}*

### Hộp Nhóm Mẫu Rãnh con (Sub TrackSample Group Box)

#### Định nghĩa

Loại hộp:*‘stsg’*

Thuộc đối tượng: Hộp Định nghĩa Rãnh con (‘*strd*’).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp này định nghĩa một rãnh con như một hoặc nhiều nhóm mẫu bằng cách đề cập tới các miêu tả nhóm mẫu tương đương miêu tả các mẫu của từng nhóm.

#### 8.14.6.2 Cú pháp

*aligned(8) class SubTrackSampleGroupBoxextends FullBox(‘stsg’, 0, 0){*

*unsigned int(32) grouping\_type ;*

*unsigned int(16) item\_count;*

*for(i = 0; i<item\_count; i++)*

*unsigned int(32) group\_description\_index;*

*}*

#### Ngữ nghĩa

*grouping\_type* có kiểu số nguyên xác định các nhóm mẫu. Giá trị sẽ giống như trong các hộp *SampletoGroup* và *SampleGroupDescription* tương ứng.

*Item\_count đếm số lượng các nhóm mẫu được liệt kê trong hộp này.*

*Group\_description\_index*có kiểu số nguyên cung cấp các chỉ số của bản ghi nhóm mẫu mà miêu tả các mẫu trong nhóm.

## Yêu cầu đối với bộ tiền giải mã

### Tổng quan

Để kiểm soát các tình huốngmà tệp tin yêu cầu có các hành động nhất định tại đầu đọc hoặc bộ dựng hình, mục này quy định một cơ chế mà cho phép các đầu đọc thực hiện đơn giản hóa tiêm một tệp tin để tìm ra các yêu cầu cho việc dựng hình một luồng bit và ngừng các đầu đọc truyền thống khỏi phải giải mã và dựng hình các tệp tin mà yêu cầu các xử lý hơn nữa. Cơ chế này áp dụng cho bất kỳ kiểuhình ảnh codec nào. Đặc biệt áp dụng cho AVC và cho đặc tả báo hiệu trường hợp này được định nghĩa trong định dạng tệp tin AVC (ISO/IEC 14496-15) mà cho phép một tệp tin liệt kê các ID bản tin SEI xuất hiện và phân biệt giữa các hành động được yêu cầu và không được yêu cầu cho xử lý dựng hình.

Cơ chế tương tự như việc chuyển đổi bảo vệ nội dung mà các bản ghi mẫu được ẩn đằng sau mẫu các bản ghi mẫu chung, 'encv', 'ENCA', vv, cho biết các đa phương tiện được mã hóa hoặc được đóng gói. Cơ chế tương tự cho việc hạn chế hình ảnh sử dụng một chuyển đổi với bản ghi mẫu chung ‘resv’. Phương pháp có thể được áp dụng khi nội dung chỉ được giải mã bởi các đầu đọc mà hiển thị cơ bản xác.

### Chuyển đổi

Phương pháp chuyển đổi được áp dụng như sau:

1. Đoạn mã bốn ký tự của đầu vào mục mẫu được thay thế bởi một mã đầu vào mục mẫu mới ‘resv’ mang ý nghĩahình ảnh bị hạn chế.
2. Một Hộp Thông tin Lược đồ Giới hạn được thêm vào miêu tả mẫu, thay thế cho tất cả các hộp khác không được sử đổi.
3. Một kiểu bản ghi mẫu khởi đầu được lưu trữ trong một Hộp Định dạng Gốc được chứa trong Hộp Thông tin Lược đồ Giới hạn.

Một Hộp Thông tin Lược đồ Giới hạn được định dạng cơ bản xác như một Hộp Thông tin Lược đồ Bảo vệ, ngoại trừ việc sử dụng định dang *‘rinf’* thay cho ‘*sinf*’ (xem dưới đây).

Kiểu bản ghi mẫu khởi đầu được chứa trong Hộp Định dạng Gốc được đặt trong Hộp Thông tin Lược đồ Giới hạn(trong một cách thức xác định tới Hộp Thông tin Lược đồ Bảo vệ cho dữ liệu đa phương tiện được mã hóa).

Tính chất bắt buộc của việc hạn chế được định nghĩa trong Hộp Kiểu Lược đồ, và dữ liệu cần thiết cho lược đồ đó được lưu trữ trong Hộp Thông tin Lược đồ, một lần nữa, tương tự để bảo vệ thông tin.

Lưu ý rằngviệc hạn chế và bảo vệ có thể được áp dụng tại cùng một thời điểm. Thứ tự của các biến đổi theo sau mã bốn ký tự của mẫu bản ghi. Ví dụ, nếu kiểu bản ghi mẫu là ‘resv’, hoàn tác việc chuyển đổi trên có thể dẫn đến kết quả trong một kiểu bản ghi mẫu *‘encv’*, chỉ ra rằng dữ liệu đa phương tiện đã được bảo vệ.

Lưu ý: nếu tác giả tệp tin chỉ muốn cung cấp thông tin tư vấn mà không thực hiện dừng các đầu đọc cũ việc đọc tệp tin, Hộp Thông tin Lược đồ Giới hạn có thể được đặt bên trong bản ghi mẫu mà không áp dụng chuyển đổi mã bốn ký tự. Trong trường hợp này không cần thiết bao gồm một Hộp Định dạng Gốc.

### Hộp Thông tin Lược đồ Giới hạn (Restricted Scheme Information box)

#### Định nghĩa

Các Loại hộp:*‘rinf’*.

Thuộc đối tượng:Mục Mẫu Giới hạn hoặc Mục Mẫu.

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp Thông tin Lược đồ Giới hạn chứa tất cả thông tin được yêu cầu đồng thời để hiểu lược đồ hạn chế được áp dụng và các thông số của nó. Hộp cũng dẫn chứng kiểu bản ghi khởi đầu (chưa được chuyển đổi) của đa phương tiện. Hộp Thông tin Lược đồ Giới hạn là một hộp chứa. Đây là bắt buộc trong một bản ghi mẫu có sử dụng một mã chỉ một luồng hạn chế, ví dụ ‘resv’.

Khi được sử dụng trong một đầu vào mục mẫu bị giới hạn, hộp này phải chứa Hộp Định dạng Gốc để ghi lại kiểu bản ghi mẫu khởi đầu và một Hộp Kiểu Lược đồ. Một Hộp Thông tin Lược đồ có thể được yêu cầu phụ thuộc vào lược đồ hạn chế.

#### Cú pháp

*aligned(8) class RestrictedSchemeInfoBox(fmt) extends Box('rinf') {*

*OriginalFormatBox(fmt) original\_format;*

*SchemeTypeBox scheme\_type\_box;*

*SchemeInformationBox info; // optional*

*}*

### Lược đồ bố trí hình ảnh lập thể

#### Tổng quan

Khi các khung hình ảnh được mã hóa lập thể được giải mã, các khung được giải mã này cũng chứa một trình diễn của các khung đóng gói hai phần theo không gian mà tạo thành một cặp lập thể (khung đóng gói) hoặc chỉ một hình ảnh của của một cặp lập thể (các ảnh trái và phải trong các rãnh khác nhau). Các hạn chế do hình ảnhmã hóa lập thể được chứa trong Hộp Hình ảnh Lập thể. Giá trị SchemeType ‘*stvi*’ (hình ảnh lập thể) được sử dụng.

#### Hộp Hình ảnh Lập thể (Stereo Video Box)

##### Định nghĩa

Loại hộp: `*stvi*’.

Thuộc đối tượng:Hộp Thông tin Lược đồ(‘*schi*’).

Hình thức sử dụng: bắt buộc(khi Kiểu Lược đồ là ‘*stvi*’).

Số lượng: một.

Hộp Hình ảnh Lập thể được sử dụng để chỉ ra rằng các khung được giải mã chứa một trình diễn của các khung đóng gói hai phần theo không gian mà tạo thành một cặp lập thể hoặc chứa một hoặc hai cảnh của một cặp lập thể. Hộp Hình ảnh Lập thể sẽ được giới thiệu khi giá trị SchemeType là ‘*stvi*’.

##### Cú pháp

*aligned(8) class StereoVideoBox extends extends FullBox(‘stvi’, version = 0, 0){*

*template unsigned int(30) reserved = 0;*

*unsigned int(2) single\_view\_allowed;*

*unsigned int(32) stereo\_scheme;*

*unsigned int(32) length;*

*unsigned int(8)[length] stereo\_indication\_type;*

*Box[] any\_box; // optional*

*}*

##### Ngữ nghĩa

*single\_view\_allowed*có kiểu số nguyên. Giá trị 0 chỉ ra rằng nội dung chỉ được hiển thị trên các màn hình hỗ trợ không gian lập thể. Khi (*single\_view\_allowed*& 1) bằng 1, nó cho phép hiển thị ảnh bên phải trên màn hình hiển thị monoscopic single-view. Khi (*single\_view\_allowed*& 2) bằng 2, nó cho phép hiển thị ảnh trái trên một màn hiển thị monoscopic single-view.

*stereo\_scheme* là một số nguyên chỉ lược đồ bố trí âm thanh lập thể được sử dụng và kiểu dấu hiệu theo lược đồ được sử dụng. Các giá trị sau đây cho lược đồ âm thanh lập thể được định nghĩa:

1: lược đồ đóng gó*khung hình I* như định nghĩa bởi bản tin Supplemental Enhancement Information xắp xếp đóng gó*khung hình I* của tiêu chuẩn ISO/IEC 14496-10 [ISO/IEC 14496-10];

2: kiểu lược đồ xắp xếp như định nghĩa trong Phụ lục L của tiêu chuẩn ISO/IEC 13818-2 [ISO/IEC 13818- 2:2000/Amd.4] ;

3: lược đồ âm thanh lập thể được định nghĩa trong tiêu chuẩn ISO/IEC 23000-11 cho cả khung hình/dịch vụ tương thích và các dịch vụ kết hợp giữa 2D/3D.

Các giá trị của *stereo\_scheme* được dành riêng.

*length* chỉ số lượng các byte dành cho trường *stereo\_indication\_type*.

*stereo\_indicaiton\_type* chỉ ra kiểu xắp xếp âm thanh lập thể theo lược đồ chỉ định âm thanh lập thể được sử dụng. Cú pháp và các ngữ nghĩa của *stereo\_indication\_type* phụ thuộc vào giá trị của lược đồ âm thanh lập thể. Cú pháp và các ngữ nghĩa của*stereo\_indication\_type*với các giá trị của *stereo\_scheme* được thể hiện như sau:

*stereo\_scheme* bằng 1: giá trị của độ dài sẽ là 4 và *stereo\_indicaiton\_type* sẽ có kiểu số nguyên không dấu (32 bit) chứa giá trị *frame\_packing\_arrangement\_type* từ Bảng D-8 của ISO/IEC 14496-10 [ISO/IEC 14496-10] (‘Định nghĩa của *frame\_packing\_arrangement\_type*’).

*stereo\_scheme* bằng 2: giá trị của độ dài sẽ là 4 và *stereo\_indication\_type* sẽ có kiểu số nguyên không dấu (32 bit) chứa kiểu giá trị từ Bảng L-1 của ISO/IEC 13818-2 [ISO/IEC 13818-2:2000/Amd.4] (‘Định nghĩa các kiểu xắp xếp’).

*stereo\_scheme bằng 3*: giá trị của độ dài sẽ là 2 và *stereo\_indication\_type* sẽ chứa 2 thành phần cú pháp của số nguyên không dấu(8 bit). Thành phần cú pháp đầu tiên sẽ chứa kiểu thành phần stereoscopic từ Bảng 4 của ISO/IEC 23000-11:2009. Bit ít quan trọng nhất của thành phần cú pháp thứ 2 sẽ chứa giá trị của *is\_left\_first* như định nghĩa trong 8.4.3 của ISO/IEC 23000-11:2009, trong khi các bit khác được dành riêng và sẽ được cài đặt về 0.

Các áp dụng sau đây khi Hộp Hình ảnh Lập thể được sử dụng:

* TrongHộp Mào đầu Rãnh
  + *width* và *height*miêu tả kích thước hiển thị trình diễn của một ảnh đơn sau khi mở đóng gói.
* Trong Hộp Mô tả Mẫu
  + *frame\_count* sẽ là 1, bởi vì các bộ giải mã cụ thể cho ra một khung đơn. Nói một cách khác, các khung thành phần bao gồm trong một hình ảnh khung được đóng gói không được dẫn chứng bởi *frame\_count*.
  + *width* và *height* dẫn chứng số lượng điểm ảnh được tính của một hình ảnh khung được đóng gói (và không phải là số lượng điểm ảnh của một ảnh đơn trong một hình ảnh khung được đóng gói).
  + Hộp Tỉ lệ Khung theo Điểm ảnhlưu tỉ lệkhung theo điểm ảnh của từng khung hình khi hình được hiển thị trên một màn hình đơn hình monoscopic. Lấy ví dụ, trong nhiều cách thức đóng gói theo không gian, Hộp Tỉ lệ Khung theo Điểm ảnh chỉ giá trị tỉ lệ khung theo điểm ảnh 2:1 hoặc 1:2, tức độ phân giải theo không gian của một khung hình hình ảnh với khung hình được đóng gói bị giảm đi một nửa theo một trục không gian so với hình ảnh đơn hình với cùng định dạng.

## Các phân đoạn

### Giới thiệu

Các trình diễnđa phương tiện có thể được phân chia thành các phân đoạn để vận chuyển, lấy ví dụ, có thể (ví dụ trong HTTP streaming) tạo thành các tệp tin mà chứa một phân đoạn hoặc các phân đoạn nối tiếp nhau – mà không cần thiết có dạng theo chuẩn ISO dựa trên định dạng tệp tinđa phương tiện phù hợp (ví dụ. chúng không chứa mộtHộp Movie).

Mục con này định nghĩa đặc tả các hộp mà có thể được sử dụng trong các thành phần như miêu tả trên.

### Hộp Kiểu Phân đoạn (Segment Type Box)

Loại hộp: `*styp*’.

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Nếu các phân đoạn được lưu trữ trong các tệp tin riêng biệt (ví dụ trong một máy chủ HTTP tiêu chuẩn), khuyến nghị rằng những “tệp tin phân đoạn “này chứa một Hộp Kiểu Phân đoạn, phải là phân đoạn đầu tiên nếu có mặt, để có thể nhận ra những tệp tin đó, và thông báo các đặc tả có sự phù hợp.

Một kiểu phân đoạn có cùng định dạng như một hộp ‘*ftyp*’[4.3], ngoại trừ trường hợp kiểu hộp là ‘*styp*’. Các nhánh trong nó có thể bao gồm cùng với các nhánh mà được bao gồm trong hộp ‘*ftyp*’ mà đứng trước hộp ‘*moov*’, và cũng có thể bao gồm các nhánh bổ sung để chỉ ra khả năng tương thích của phân đoạn này với nhiều đặc tả khác nhau.

Các Hộp Kiểu Phân đoạn hợp lệ sẽ là hộp đầu tiên trong một phân đoạn. Các Hộp Kiểu Phân đoạn có thể bị xóa đi nếu các phân đoạn nối tiếp nhau (ví dụ tạo thành một tệp tin đầy đủ), nhưng điều này là không bắt buộc. Các Hộp Kiểu Phân đoạn mà không nằm đầu tiên trong các tệp tin có thể bị kiểu bỏ.

### Hộp Chỉ số Phân đoạn (Segment Index Box)

#### Định nghĩa

Loại hộp: `*sidx*’.

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp Chỉ số Phân đoạn (‘*sidx*’) cung cấp một cho biết thu gọn của một luồng đa phương tiện trong phân đoạn đa phương tiện mà nó áp dụng. Nó được thiết kế vì vậy nó có thể được sử dụng không chỉ với các định dạng đa phương tiện dựa trên đặc tả này (ví dụ các phân đoạn chứa các bảng mẫu hoặc các phân đoạn hình ảnh), mà còn các định dạng đa phương tiện khác (ví dụ, MPEG-2 Transport Streams [ISO/IEC 13818-1]). Đối với lý do này, miêu tả hình thức của hộp ở đây cố ý chung chung, và do đó tại cuối của Đoạn miêu tả này các định nghĩa cụ thể cho các phân đoạn sử dụng các phân mảnh hình ảnh được đưa ra.

Từng Hộp Chỉ số Phân đoạn chỉ ra làm thế nào một phân đoạn được phân chia thành một hoặc nhiều phân đoạn con(bản thân các phân đoạn này có thể được chia nhỏ hơn nữa bằng cách sử dụng các Hộp Chỉ số Phân đoạn).

Một phân đoạn con được định nghĩa như một khoảng thời gian của bộ chứa phân đoạn con và tương ứng tới một dải các byte củabộ chứa phân đoạn con. Các khoảng thời gian của tất cả các phân đoạn con là tổng hợp của khoảng thời gian của bộ chứa phân đoạn con.

Từng bản ghi trong Hộp Chỉ số Phân đoạn chứa một kiểu tham chiếu mà chỉ ra các điểm tham chiếu trực tiếp tới các byte đa phương tiện của một lá phân đoạn con tham chiếu, hoặc tới một Hộp Chỉ số Phân đoạn mà miêu tả làm thế nào phân đoạn con được tham chiếu được chia nhỏ thêm nữa; kết quả là, phân đoạn có thể được tạo chỉ mục trong một ‘phân cấp’ hoặc ‘chuỗi hình sao’ hoặc hình thức khác theo thời gian và thông tin bù byte cho các Hộp Chỉ số Phân đoạn khác áp dụng cho các phần của cùng phân đoạn con.

Từng Hộp Chỉ số Phân đoạn cung cấp thông tin về một luồng đa phương tiện đơn của Phân đoạn, được chỉ dẫn tới như luồng tham chiếu. Nếu được cung cấp, Hộp Chỉ số Phân đoạn đầu tiên trong một phân đoạn, cho một luồng đa phương tiện đưa ra, sẽ ghi lại toàn bộ luồng đa phương tiện đó trong phân đoạn, và sẽ ưu tiên bất kỳ Hộp Chỉ số Phân đoạn khác trong phân đoạn cho cung luồng đa phương tiện.

Nếu một chỉ mục phân đoạn có mặt trong ít nhất một luồng đa phương tiện nhưng không phải tất cả các luồng đa phương tiện trong phân đoạn này, sau đó thông thường một luồng đa phương tiện trong đó không phải mọi đơn vị truy nhập được mã hóa độc lập, như hình ảnh, được lựa chọn để tạo chỉ mục. Đối với bất kỳ luồng đa phương tiện nào mà không có chỉ mục phân đoạn được giới thiệu, được tham chiếu như luồng không chỉ mục, luồng đa phương tiện kết được kết hợp với Hộp Chỉ số Phân đoạn đầu tiên phục vụ như một luồng tham chiếu trong một cảm nhận rằng nó cũng miêu tả các phân đoạn con cho bất kỳ luồng đa phương tiện không chỉ mục nào.

LƯU Ý 1:Các hạn chế có thể được miêu tả trong các đặc tả được suy ra hơn nữa.

Các Hộp Chỉ số Phân đoạn có thể ở trong nội tuyến cùng tệp tin như đa phương tiện được đánh chỉ mục, hoặc trong một vài trường hợp, trong một tệp tin riêng biệt chứa chỉ thông tin đánh chỉ mục.

Một Hộp Chỉ số Phân đoạn chứa một chuỗi các tham chiếu tới các phân đoạn con của phân đoạn được ghi lại bởi hộp. Các phân đoạn con được tham chiếu là liên tiếp trong thời gian trình diễn. Tương tự, các byte được tham chiếu tới một Hộp Chỉ số Phân đoạn là luôn luôn liên tiếp trong cùng tệp tinđa phương tiện, và phân đoạn chỉ mục riêng biệt, hoặc trong tệp tin đơn nhất nếu các chỉ mục được đặt trong tệp tinđa phương tiện. Kích thước tham chiếu tính số lượng các byte trong tài liệu tham chiếu.

LƯU Ý 2:Một phân đoạn đa phương tiện có thể được đánh chỉ mục bởi nhiều hơn một “top-level “Hộp Chỉ số Phân đoạn mà độc lập với từng phân đoạn, từng chỉ mục của từng luồng đa phương tiện trong phân đoạn đa phương tiện. Trong các phân đoạn chứa các byte được tham chiếu nhiều luồng đa phương tiện có thể chứa đa phương tiện từ nhiều luồng, thậm chí Hộp Chỉ số Phân đoạn cung cấp thông tin đồng bộ cho chỉ một luồng đa phương tiện.

Trong tệp tin chứa Hộp Chỉ số Phân đoạn, điểm neo cho một Hộp Chỉ số Phân đoạn là byte đầu tiên sau hộp đó. Nếu có 2 tệp tin, điểm neo trong tệp tinđa phương tiện là bắt đầu của phân đoạn ở mức trên cùng (ví dụ bắt đầu của tệp tin phân đoạn nếu từng phân đoạn được lưu trữ trong một tệp tin riêng biệt). Tài liệu trong tệp tin chứa đa phương tiện (đây có thể cũng là tệp tin mà chứa các Hộp Chỉ số Phân đoạn) bắt đầu tại phần bù cho biết từ điểm neo. Nếu có 2 tệp tin, tài liệu trong tệp tin chỉ mục bắt đầu tại điểm neo, ví dụ tại ngay sau Hộp Chỉ số Phân đoạn.

Trong hai yêu cầu bắt buộc (a) là, về thời gian, các phân đoạn con là liền kề nhau, đó là, từng bản ghi trong vòng lặp là liên tục từ ngay phía trước một phân đoạn con và (b) trong một tệp tin nhất định (tệp tin được tích hợp, tệp tinđa phương tiện, hoặc chỉ số cạnh tệp tin) các byte tham chiếu là liền kề, có một số khả năng, bao gồm:

* Một tham chiếu tới một Hộp Chỉ số Phân đoạn có thể bao gồm, trong số các byte của nó, theo ngay sau các Hộp Chỉ số Phân đoạn mà ghi lại các phân đoạn con.
* Trong một tệp tin tích hợp, sử dụng trường *first\_offset*, có thể phân tách các Hộp Chỉ số Phân đoạn từ đa phương tiện mà chúng tham khảo tới;
* Trong một tệp tin tích hợp, có thể đặt các Hộp Chỉ số Phân đoạn cho các phân đoạn con gần với đa phương tiện chúng đánh chỉ mục;
* Khi một tệp tin riêng biệt chứa các chỉ số phân đoạn được sử dụng, có thể cho các bản ghi vòng lặp để thành ‘dạng tổng hợp’, một vài tới Hộp Chỉ số Phân đoạn trong chỉ mục phân đoạn con, một vài tới các phân đoạn con đa phương tiện trong tệp tinđa phương tiện.

LƯU Ý 3:Các hồ sơ có thể được sử dụng để hạn chế các vị trí của các chỉ mục phân đoạn, hoặc toàn bộ các phức tạp của việc đánh chỉ mục.

Hộp Chỉ số Phân đoạn ghi lại sự hiện diện của các điểm truy nhập luồng (SAPs), được định nghĩa trong Phụ lục I, trong các phân đoạn mạng được tham chiếu. Phụ lục miêu tả các đặc điểm của SAPs, như là ISAU, ISAP và TSAP, cũng như các kiểu SAP, tất cả được sử dụng trong các giải nghĩa dưới đây. Một phân đoạn con bắt đầu với một SAP khi phân đoạn con chứa một SAP, và đối với SAP đầu tiên, ISAU là chỉ mục của đơn vị truy nhập đầu tiên theo sau ISAP, và ISAP được chứa trong phân đoạn con.

Đối với các phân đoạn được dựa trên tài liệu đặc tả này (ví dụ dựa trên các bảng mẫu moive hoặc các phân mảnh movie):

* Một đơn vị truy nhập là một ví dụ;
* Một phân đoạn con là một tập hợp tự chứa đựng của một hoặc nhiều hơn các mảnh ghép movie liên tục; một tập hợp tự chứa đựng chứa một hoặc nhiều các Hộp Phân mảnh Movievới các Hộp Dữ liệu Đa phương tiện tương ứng, và một Hộp Dữ liệu Đa phương tiện chứa dữ liệu được tham chiếu bởi một Hộp Phân mảnh Moviephải theo sau Hộp Phân mảnh Movieđó và đứng trước Hộp Phân mảnh Moviekế tiếp chứa thông tin về cùng rãnh đó;
* Các Hộp Chỉ số Phân đoạn sẽ được đặt trước tài liệu phân đoạn con chúng ghi lại, đó là, trước bất kỳ Hộp Phân mảnh Movie (‘*moof*’) của tài liệu đã được ghi lại của phân đoạn con;
* Các luồng là các rãnh trong định dạng tệp tin, và các ID luồng là các ID rãnh;
* Một phân đoạn con chứa một điểm truy nhập luồng nếu một phân phân mảnh rãnhtrong phân đoạn con cho rãnh với *track\_id* bằng với *reference\_ID* chứa một điểm truy nhập luồng;
* Khởi tạo dữ liệu cho các SAP bao gồmHộp Movie;
* Số lần trình diễn trong trục thời gian movie là tổng số lần thành phần sau khi ứng dụng cho danh sách chỉnh sửa bất kỳ cho rãnh;
* ISAP là một vị trí cơ bản xác điểm khởi động của một hộp mức cao nhất, như là một hộp phân mảnh movie ‘*moof*’;
* Một SAP kiểu 1 hoặc kiểu 2 được cho biết như là một mẫu đồng bộ, hoặc bởi sample\_is\_non\_sync\_sample bằng 0 trong phân mảnh movie;
* Một SAP kiểu 3 được đánh dấu như một thành viên của một nhóm mẫu của kiểu ‘rap’;
* Một SAP kiểu 4 được đánh dấu như một thành viên của một nhóm mẫu của kiểu ‘roll’ nơi giá trị của trường *roll\_distance* lớn hơn 0.

LƯU Ý 4: Đối với các SAP kiểu 5 và 6, không hỗ trợ đặc tả báo hiệu trong định dạng tệp tinđa phương tiệntheo chuẩn ISO.

#### Cú pháp

*aligned(8) class SegmentIndexBox extends FullBox(‘sidx’, version, 0) {*

*unsigned int(32) reference\_ID;*

*unsigned int(32) timescale;*

*if (version==0) {*

*unsigned int(32) earliest\_presentation\_time;*

*unsigned int(32) first\_offset;*

*}*

*else {*

*unsigned int(64) earliest\_presentation\_time;*

*unsigned int(64) first\_offset;*

*}*

*unsigned int(16) reserved = 0;*

*unsigned int(16) reference\_count;*

*for(i=1; i <= reference\_count; i++)*

*{*

*bit (1) reference\_type;*

*unsigned int(31) referenced\_size;*

*unsigned int(32) subsegment\_duration;*

*bit(1) starts\_with\_SAP;*

*unsigned int(3) SAP\_type;*

*unsigned int(28) SAP\_delta\_time;*

*}*

*}*

#### Ngữ nghĩa

*reference\_ID* cung cấp ID của luồng cho luồng tham chiếu; nếu Hộp chỉ số phân đoạnnày được tham chiếu từ một Hộp Chỉ số Phân đoạn “cha “, giá trị *reference\_ID* sẽ là cùng với giá trị của *reference\_ID* của Hộp Chỉ số Phân đoạn “cha “;

*timescale* cung cấp khoảng thời gian, đánh dấu từng giây, đối với thời gian và các trường thời gian trong hộp này; khuyến nghị rằng điều này phù hợp với khoảng thời gian của luồng tham chiếu hoặc rãnh; đối với các tệp tin được dựa trên đặc tả này, đó là trường khoảng thời gian của Hộp Mào đầu Đa phương tiện của rãnh;

*earliest\_presentation\_time* là thời gian trình diễn sớm nhất của bất kỳ đơn vị truy nhập nào trong luồng tham chiếu trong phân đoạn con đầu tiên, trong khoảng thời gian được cho biết trong trường khoảng thời gian;

*first\_offset* là khoảng cách bằng byte, trong tệp tin chứa đa phương tiện, từ điểm neo, tới byte đầu tiên của tài liệu đã được đánh chỉ mục;

*reference\_count* cung cấp số lượng các mục được tham chiếu;

*reference\_type:* khi thiết lập giá trị 1 cho biết tham chiếu tới một Hộp Chỉ số Phân đoạn (‘*sidx*’); mà không tham chiếu tới nội dung đa phương tiện (ví dụ, trong trường hợp các tệp tin được dựa trên đặc tả này, tới một Hộp Phân mảnh Movie); nếu một phân đoạn chỉ mục riêng biệt được sử dụng, sau đó các bản ghi với tham chiếu kiểu 1 trong phân đoạn chỉ mục, và các bản ghi với tham chiếu kiểu 0 là trong tệp tinđa phương tiện;

*referenced\_size:* khoảng cách theo byte từ byte đầu tiên của đối tượng được tham chiếu tới byte đầu tiên của đối tượng tham chiếu kế tiếp, hoặc trong trường hợp bản ghi cuối cùng, điểm cuối của tài liệu được tham chiếu;

*subsegment\_duration:* ki tham chiếu tới Hộp Chỉ số Phân đoạn, trường này mang tổng của các trường *subsegment\_duration* trong hộp đó; khi tham chiếu tới một phân đoạn con, trường này mang sự khác biệt giữa thời gian trình diễn sớm nhất của bất kỳ đơn vị truy nhập nào của luồng tham chiếu trong phân đoạn con kế tiếp (hoặc phân đoạn con đầu tiên của phân đoạn kế tiếp, nếu đây là phân đoạn con cuối cùng của phân đoạn, hoặc là kết cuối của thời gian trình diễn của luồng tham chiếu nếu đây là phân đoạn cuối cùng của luồng) và là thời gian trình diễn sớm nhất của bất kỳ đơn vị truy nhập nào của luồng tham chiếu trong phân đoạn con được tham chiếu; khoảng thời gian trong cùng các đơn vị như *earliest\_presentation\_time*;

*SAP\_type* chỉ ra một kiểu SAP như định nghĩa trong Phụ lục I, hoặc giá trị 0. Các giá trị khác được dành riêng. Đối với các giải nghĩa chi tiết của trường này trong sự kết hợp với các trường khác, xem bảng dưới đây.

*SAP\_delta\_time*: chỉ ra TSAP của SAP đầu tiên, để giải mã, trong phân đoạn con được tham chiếu cho luồng tham chiếu. Nếu các phân đoạn con được tham chiếu không chứa một SAP, SAP\_delta\_time được dành riêng với giá trị 0; mặt khác SAP\_delta\_time là khác biệt giữa thời gian trình diễn sớm nhất của phân đoạn con, và TSAP (ghi chú rằng sự khác biệt này có thể là 0, trong trường hợp phân đoạn con này bắt đầu với một SAP).

Bảng 4 – Ngữ nghĩa của SAP và việc kết hợp kiểu tham chiếu

|  |  |  |  |
| --- | --- | --- | --- |
| ***starts\_with\_SAP*** | ***SAP\_type*** | ***reference\_type*** | **Ngữ nghĩa** |
| 0 | 0 | 0 hoặc 1 | Không thông tin của các SAP được cung cấp |
| 0 | 1 tới 6, bao gồm | 0 (đa phương tiện) | Phân đoạn con chứa (nhưng có thể không bắt đầu với) một SAP của *SAP\_type* được cho và SAP đầu tiên của *SAP\_type* được cho tương ứng tới SAP\_delta\_time. |
| 0 | 1 tới 6, bao gồm | 1 (chỉ mục) | Tất cả các phân đoạn con được tham chiếu chứa một SAP của phần lớn *SAP\_type* được cho và không có các SAP nào là kiểu không được biết. |
| 1 | 0 | 0 (đa phương tiện) | Phân đoạn con bắt đầu với một SAP của một kiểu không được biết. |
| 1 | 0 | 1 (chỉ mục) | Tất cả các phân đoạn con được tham chiếu bắt đầu với một SAP mà có thể của một kiểu không được biết. |
| 1 | 1 tới 6, bao gồm | 0 (đa phương tiện) | Phân đoạn con được tham chiếu bắt đầu với một SAP của *SAP\_type* được cho. |
| 1 | 1 tới 6, bao gồm | 1 (chỉ mục) | Tất cả các phân đoạn con được tham chiếu bắt đầu với một SAP của tất cả *SAP\_type* được cho và không SAP nào là kiểu không được biết. |

### Hộp Chỉ số Phân đoạn con (Subsegment Index Box)

#### Định nghĩa

Loại hộp: `*ssix*’.

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

Hộp Chỉ số Phân đoạn con (‘*ssix*’) cung cấp một ánh xạ từ các mức (như định nghĩa bởi Hộp Chỉ định Mức) tới các dải byte của phân đoạn con được đánh chỉ mục. Nói một cách khác, hộp này cung cấp một chỉ mục ngắn gọn để làm thế nào dữ liệu trong một phân đoạn con được yêu cầu theo các mức vào trong từng phần các phân đoạn con.

Mỗi byte trong phân đoạn con sẽ được chỉ định một mức và vì vậy dải số đếm sẽ phải lớn hơn hoặc bằng 2. Nếu dải này không liên quan tới bất kỳ thông tin nào trong danh sách mức được chỉ định thì có thể sử dụng bất kỳ mức nào không có trong danh sách đó.

Sẽ có các Hộp Chỉ số Phân đoạn con 0 hoặc 1 trong mỗi Hộp Chỉ số Phân đoạn, các chỉ số này chỉ trỏ đến các phân đoạn con lớp lá, ví dụ, chỉ có chỉ số các phân đoạn con mà không có các chỉ số phân đoạn. Hộp Chỉ số Phân đoạn con, nếu có sẽ là hộp đứng sau Hộp Chỉ số Phân đoạn. Hộp Chỉ số Phân đoạn con lưu các phân đoạn con mà Hộp Chỉ số Phân đoạn đứng ngay trước đó trỏ tới.

Nhìn chung, dữ liệu đa phương tiện được xây dựng từ các dải byte chưa hoàn thiện, ví dụ nó không phù hợp với định dạng đa phương tiện của toàn bộ phân đoạn con.

Đối với các phân đoạn con lớp lá được dựa trên đặc tả này (ví dụ được dựa trên các bảng mẫu hình ảnh và các phân mảnh hình ảnh):

* Từng cấp sẽ được gán tới cơ bản xác một phần phân đoạn con, ví dụ các dải byte cho từng cấp sẽ là liên tiếp;
* Các mức của một phần các phân đoạn con sẽ được gán bởi số tăng dần trong một phân đoạn con, ví dụ các mẫu của một phần phân đoạn con có thể phụ thuộc vào bất kỳ mẫu nào của các phần phân đoạn con đứng trước trong cùng phân đoạn con, nhưng không phải là các phân đoạn con khác xung quanh. Lấy ví dụ, từng phần phân đoạn con chứa các mẫu có một mức định danh thời gian và các phần phân đoạn con xuất hiện trong mức thời gian tăng dần trật tự trong phân đoạn con;
* Khi một phần phân đoạn con được truy nhập trong cách thức này, đối với bất kỳ giá trị *assignment\_type* khác giá trị 3, Hộp Dữ liệu Đa phương tiện cuối cùng có thể không được hoàn thành, đó là, ít dữ liệu được truy nhập hơn là độ dài chỉ định của các cho biếtHộp Dữ liệu Đa phương tiện đang có mặt. Độ dài của Hộp Dữ liệu Đa phương tiện có thể cần điều chỉnh, hoặc lớp đệm được sử dụng. Giá trị *padding\_flag* trong các cho biếtHộp Chỉ định Mức có thể được thay bằng các giá trị 0 bất kể điều này làm mất dữ liệu. Nếu không, dữ liệu mẫu cho các mẫu được gán cho các mức mà không được truy nhập là không có mặt, và cần cẩn trọng không cố gắng để xử lý các mẫu như vậy.
* Các khoảng dữ liệu ứng với các phần phân đoạn con bao gồm cả các Hộp Phân mảnh Movie và Hộp Dữ liệu Đa phương tiện. Phần phân đoạn con đầu tiên, ví dụ như phần có mức thấp nhất, sẽ tương ứng với Hộp Phân mảnh Movie và Hộp Dữ liệu Đa phương tiện (hoặc các phần của Hộp Dữ liệu Đa phương tiện), trong khi đó, các phân đoạn con thành phần (các mức cao hơn) có thể chỉ tương ứng với các Hộp Dữ liệu Đa phương tiện (hoặc các phần của Hộp Dữ liệu Đa phương tiện).

LƯU Ý:giá trị *assignment\_type* bằng 0 (được định nghĩa trong Hộp Chỉ số Phân đoạn con ‘*leva*’) có thể được sử dụng, lấy ví dụ, kết hợp với nhóm mẫu mức thời gian (‘tele’) khi các khung của một dòng bit hình ảnh được yêu cầu tạm thời trong các phân đoạn con; giá trị *assignment\_type* bằng 2 có thể được sử dụng, lấy ví dụ, khi từng ảnh của một dòng bit hình ảnh đa ảnh được chứa trong một phân mảnh movie đơn.

Giá trị *assignment\_type* bằng 3 có thể được sử dụng, lấy ví dụ, khi các phân mảnh movie bao gồm âm thanh và hình ảnh (bao gồm các Hộp Dữ liệu Đa phương tiện tương ứng) là đan xen nhau. Mức đầu tiên có thể được xác định để chứa các phân mảnh âm thanh movie (bao gồm các Hộp Dữ liệu Đa phương tiện tương ứng), trong khi mức thứ 2 có thể được xác định để chứa cả các phân mảnh âm thanh và hình ảnh (bao gồm tất cả Hộp Dữ liệu Đa phương tiện)

#### Cú pháp

*aligned(8) class SubsegmentIndexBox extends FullBox(‘ssix’, 0, 0) {*

*unsigned int(32) subsegment\_count;*

*for*(*i=1; i <= subsegment\_count; i++){*

*unsigned int(32) range\_count;*

*for* (*j=1; j <= range\_count; j++) {*

*unsigned int(8) level;*

*unsigned int(24) range\_size;*

*}*

*}*

*}*

#### Ngữ nghĩa

*subsegment\_count*có kiểu số nguyên dương xác định số lượng của các phân đoạn con mà thông tin phần phân đoạn con được xác định trong hộp này. Giá trị *subsegment\_count* sẽ bằng với giá trị*reference\_count* (ví dụ số lượng các tham chiếu phân mảnh movie) trong Hộp Chỉ số Phân đoạn liền ngay phía trước.

*range\_count* xác định số lượng các mức của phần phân đoạn con trong đó dữ liệu đa phương tiện được nhóm lại. Giá trị này sẽ lớn hơn hoặc bằng 2.

*range\_size* chỉ kích thước của phần phân đoạn con.

*level* xác định mức để phần phân đoạn con này được gán.

### Hộp Thời gian Tham chiếu Nguồn (Producer Reference Time Box)

#### Định nghĩa

Loại hộp: `*prft*’

Thuộc đối tượng: tệp tin.

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc nhiều.

The Hộp Tham chiếu Thời gian Nguồn cung cấp thời gian liên hệ tới các phân mảnh movie, hoặc các tệp tin chứa các phân mảnh movie (như các phân đoạn) được xuất ra. Khi những tệp tin này đều được xuất ra và tiêu thụ trong thời gian thực, thì điều này có thể cung cấp cho khách hàng thông tin để kích hoạt tại tốc độ tương đương, do đó có khả năng ngăn ngừa tràn bộ đệm.

Hộp này có liên hệ tới hộp phân mảnh movie kế tiếp theo sau nó trong yêu cầu của dòng bit. Hộp này phải tuân theo bất kỳ kiểu phân đoạn hoặc hộp chỉ mục phân đoạn (nếu có) trong phân đoạn, và xuất hiện trước hộp phân mảnh movie theo ngay sau (mà nó đề cập). Nếu một phân đoạn tệp tin chứa bất kỳ các hộp tham chiếu thời gian sản xuất, thì phân đoạn đầu tiên của chúng sẽ xuất hiện trước khi hộp phân mảnh movie đầu tiên trong phân đoạn đó.

Hộp chứa một giái trị thời gian được đo trên một đồng hồ mà tăng dần tại cùng tốc độ như một đồng hồ UTC đồng bộ theo giao thức NTP [RFC 5905]. Hộp được kết hợp với một thời gian của đa phương tiện cho một phân mảnh của các rãnh trong các phân đoạn movie. Thời gian đa phương tiện đó sẽ ở trong dải các khoảng thời gian trong rãnh đó được kết hợp trong phân mảnh movie. Các thời gian tham chiếu sản sinh sẽ được kết hợp với ít nhất một rãnh.

#### Cú pháp

*aligned(8) class ProducerReferenceTimeBox extends FullBox(‘prft’, version, 0) {*

*unsigned int(32) reference\_track\_ID;*

*unsigned int(64) ntp\_timestamp;*

*if (version==0) {*

*unsigned int(32) media\_time;*

*} else {*

*unsigned int(64) media\_time;*

*}*

*}*

#### Ngữ nghĩa

*reference\_track\_ID* cung cấp *track\_ID* cho rãnh tham chiếu.

*ntp\_timestamp* chỉ một thời gian UTC trong định dạng NTP tương đương với giá trị *decoding\_time.*

*media\_time* tương ứng với cùng thời gian với *ntp\_timestamp*, nhưng trong các đơn vị thời gian được sử dụng cho rãnh tham chiếu, và được đo lường trên đồng hồ đa phương tiện này khi đa phương tiện được sản sinh.

LƯU Ý: Trong phần lớn các trường hợp nhãn thời gian này sẽ không bằng với nhãn thời gian của mẫu đầu tiên của phân mảnh tiếp giáp của rãnh tham chiếu, nhưng nó được khuyến nghị nó là trong dải của phân mảnh chứa hộp thời gian tham chiếu sản sinh này.

## Hỗ trợ các rãnh không hoàn chỉnh

### Tổng quan

Phần này mô tả các định dạng đầu vào mục mẫu của các rãnh không hoàn chỉnh. Các rãnh không hoàn chỉnh có thể chứa các mẫu trống hoặc chưa nhận được đúng theo như định dạng mẫu.

Các rãnh không hoàn chỉnh có thể là kết quả của việc chỉ nhận được một phần các phân mảnh con theo như danh sách chỉ định mức và *padding\_flag* trong Hộp Chỉ định Mức sẽ thể hiện dữ liệu trong Hộp Dữ liệu Đa phương tiệnkhông nhận được có thể bị thay thế bởi các số 0. Vì vậy, dữ liệu mẫu được ấn định thuộc các mức không truy nhập được sẽ không xuất hiện và lưu ý không nên cố gắng xử lý các mẫu như vậy. Tuy nhiên, trong một phần các phân đoạn con nhận được, một số rãnh có thể vẫn đầy đủ nội dung trong khi một số khác thì chưa đầy đủ (không hoàn chỉnh) và chỉ chứa dữ liệu tham chiếu đến các rãnh hoàn chỉnh.

Phần này mô tả việc hỗ trợ các định dạng đầu vào mẫu của các rãnh không hoàn chỉnh. Với sự hỗ trợ đó, các thiết bị đọc có thể phát hiện các rãnh không hoàn chỉnh từ các đầu vào mục mẫu và tránh được việc phải xử lý các rãnh như vậy, hoặc các mẫu rỗng hoặc các mẫu không nhận được.

Việc hỗ trợ các rãnh không hoàn chỉnh tương tự như việc chuyển đổi bảo vệ nội dung, trong đó các đầu vào mục mẫu được ẩn phía sau các đầu vào mục mẫu tổng quát, ví dụ như *‘encv’* và *‘enca’*. Bởi vì định dạng của đầu vào mục mẫu thay đổi theo media-type nên mã 4 ký tự đóng gói khác nhau được sử dụng cho các rãnh không hoàn chỉnh của mỗi kiểu đa phương tiện (âm thanh, hình ảnh, văn bản,...). Đó là:

|  |  |
| --- | --- |
| **Kiểu luồng (rãnh)** | **Mã Sample-Entry** |
| Hình ảnh | icpv |
| Âm thanh | icpa |
| Văn bản | icpt |
| Hệ thống | icps |
| Chỉ dẫn | icph |
| Siêu dữ liệu được sắp xêp theo thời gian | icpm |

Dữ liệu mẫu của rãnh không hoàn chỉnh có thể được tích hợp trong các mẫu của rãnh khác thông qua tham chiếu, vì vậy không nên kiểu bỏ một rãnh không hoàn chỉnh nếu có các điểm tham chiếu rãnh trỏ tới.

LƯU Ý: Việc chọn lựa mức theo máy khách ghi ban đầu có thể thay đổi theo thời gian và tại các thời điểm rãnh hoàn chỉnh. Mức này không được đề cập đến ở đây và không yêu cầu đầu vào mục mẫu phải chuyển đổi từ “không hoàn chỉnh “sang “hoàn chỉnh “khi thực tế tất cả các mức đều nhận được trong một khoảng thời gian. Lưu ý rằng định dạng gốc (original format) có thể được mã hóa nếu việc tiếp nhận và giải mã một phần phù hợp với định dạng được mã hóa đó.

### Chuyển đổi

Đầu vào mục mẫu của một rãnh không hoàn chỉnh nguyên nhân ví dụ do nhận được một phần thì nên được chỉnh sửa như sau:

1. Mã 4 ký tự của đầu vào mục mẫu, ví dụ như ‘avc 1’, được thay bởi một mã mới là ‘icpv’, thể hiện đó là một rãnh không hoàn chỉnh.
2. Hộp Thông tin Rãnh không hoàn chỉnh được thêm vào mô tả mẫu, các hộp khác không giữ nguyên.
3. Kiểu đầu vào mục mẫu gốc, ví dụ ‘avc1’, được lưu trong Hộp Định dạng Gốc, hộp này lại nằm trong Hộp Thông tin Rãnh Hoàn chỉnh.

Sau khi chuyển đổi, đầu vào mục mẫu AVC có thể có dạng sau:

*class IncompleteAVCSampleEntry() extends VisualSampleEntry (‘icpv’){*

*CompleteTrackInfoBox();*

*AVCConfigurationBox config;*

*MPEG4BitRateBox (); // optional*

*MPEG4ExtensionDescriptorsBox (); // optional*

*}*

### Hộp Thông tin Rãnh Hoàn chỉnh (Complete Track Information Box)

#### Định nghĩa

Loại hộp: ‘cinf’.

Thuộc đối tượng: Đầu vào mục Mẫu của Rãnh không hoàn chỉnh.

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp Thông tin Rãnh Hoàn chỉnh nằm trong Hộp Định dạng Gốc, chứa định dạng đầu vào mục mẫu của một rãnh hoàn chỉnh được chuyển đổi thành rãnh không hoàn chỉnh. Hộp Thông tin Rãnh Hoàn chỉnh có thể chứa các hộp tùy chọn, ví dụ như thông tin yêu cầu để xử lý các mẫu trong các rãnh không hoàn chỉnh. Hộp Thông tin Rãnh hoàn chỉnh này là hộp chứa, đồng thời là hộp bắt buộc phải có trong các đầu vào mục mẫu có sử dụng mã chỉ thị rãnh không hoàn chỉnh.

#### Syntax

*aligned(8) class CompleteTrackInfoBox(fmt) extends Box('cinf') {*

*OriginalFormatBox(fmt) original\_format;*

*}*

# Định dạng rãnh chỉ dẫn

## Định dạngrãnh chỉ dẫn RTP và SRTP

### Giới thiệu

RTP là giao thức truyền tải thời gian thực được định nghĩa bởi IETF (RFC 3550 và 3551) và hiện tại được định nghĩa để có thể mang một tập xác định các dạng đa phương tiện (chủ yếu là âm thanh và hình ảnh) và mã hóa. Việc đóng gói các luồng MPEG-4 cơ bản vào RTP đang được thảo luận ở cả hai tổ chức. Tuy nhiên, rõ ràng rằng cách thức đa phương tiện được đóng gói không khác với các kỹ thuật hiện thời đang được sử dụng cho các codec khác trong RTP và được hỗ trợ bởi cơ chế này.

Trong RTP chuẩn, mỗi luồng đa phương tiện được gửi như một luồng RTP riêng; việc dồn kênh được thực hiện bằng cách sử dụng dồn kênh theo IP mức cổng, chứ không phải dồn dữ liệu từ nhiều luồng vào một phiên RTP. Tuy nhiên, nếu sử dụng MPEG, có thể cần phải thực hiện dồn một vài rãnh đa phương tiệnvào một rãnh RTP (ví dụ như khi sử dụng MPEG-2 transport trong RTP, hoặc FlexMux). Mỗi một rãnh chỉ dẫnvì vậy được gắn vào một tập các rãnh đa phương tiệnthông qua rãnh tham chiếu. Các rãnh chỉ dẫnđó trích xuất dữ liệu từ các rãnh đa phương tiệncủa chúng thông qua việc lập bảng. Rãnh chỉ dẫntham khảo tới các rãnh đa phương tiệncó kiểu tham chiếu là ‘*hint’*.

Thiết kế này quyết định kích cỡ gói vào thời điểm rãnh chỉ dẫn máy chủđược tạo ra; vì thế trong phần khai báo cho rãnh chỉ dẫn, cần chỉ ra kích cỡ gói tin đã được lựa chọn. Phần này trong miêu tả mẫu.Lưu ý rằng nó hợp lệ với nhiều RTP rãnh chỉ dẫncho mỗi rãnh đa phương tiện, với những lựa chọn kích cỡ gói khác nhau.Tương tự như vậy, thang thời gian cho RTP clock cũng được cung cấp. Timescale cho rãnh chỉ dẫn máy chủthường được lựa chọn để phù hợp với thang thời gian của các rãnh đa phương tiện, hoặc một giá trị phù hợp sẽ được lựa chọn cho máy chủ. Trong một số trường hợp, thang thời gian RTP sẽ khác biệt (ví dụ 90 kHz cho một số dữ liệu MPEG), và cho phép sự biến động đó. Thông tin miêu tả phiên (SAP/SDP) được lưu trữ trong những Hộp Dữ liệu Người dùngtrong rãnh đó.

Các rãnh chỉ dẫn RTPkhông sử dụng bảng tổ hợp thời gian dịch chuyển (‘*ctts’*). Thay vào đó, quá trình hinting cho các rãnh chỉ dẫn máy chủthiết lập trình tự phát và các nhãn thời gian đúng, có thể bằng cách sử dụng độ dịch chuyểnthời gian truyền dẫn để đặt các thời gian truyền dẫn.

Các nội dung được chỉ dẫn có thể yêu cầu sử dụng SRTP cho streaming bằng cách sử dụng định dạng rãnh chỉ dẫndành cho SRTP, được định nghĩa ở đây. Các rãnh chỉ dẫn SRTPđược định dạng giống như các rãnh chỉ dẫn RTP, ngoại trừ:

1. Tên của đầu vào mục mẫu được thay đổi từ “*rtp*” sang “*srtp*” để báo cho máy chủbiết cần sử dụng giao thức SRTP;
2. Một hộp phụ trợ được thêm vào đầu vào mục mẫu có thể được sử dụng để hướng dẫn máy chủ trong trường hợp phải áp dụng các biện pháp mã hóa trực tiếp (on-the-fly) và bảo vệ tính toàn vẹn.

### Định dạng miêu tả mẫu

Rãnh chỉ dẫn máy chủ RTPlà những rãnh chỉ dẫn(media handler *'hint'*), có entry-format trong miêu tả mẫu là *'rtp'*:

*class RtpHintSampleEntry() extends SampleEntry (‘rtp ‘) {*

*uint(16)hinttrackversion = 1;*

*uint(16)highestcompatibleversion = 1;*

*uint(32) maxpacketsize;*

*boxadditionaldata[];*

*}*

Phiên bản rãnh chỉ dẫn(*hinttrackversion*) hiện tại là 1; trường phiên bản tương thích cao nhất (*highestcompatibleversion*) cho biết phiên bản cũ nhất mà rãnh này có thể tương thích ngược.

Trường *maxpacketsize* biểu thị kích cỡ của gói tin lớn nhất mà rãnh sẽ tạo ra.

Thông tin bổ sung là một tập các hộp, được lấy từ các hộp sau đây.

*class timescaleentry() extends Box(‘tims’) {*

*uint(32) timescale;*

*}class timeoffset() extends Box(‘tsro’) {*

*int(32)offset;*

*}class sequenceoffset extends Box(‘snro’) {*

*int(32)offset;*

*}*

Trường đầu vào mục thang thời gian là bắt buộc. Hai trường còn lại là tùy chọn. Các giá trị độ dịch chuyển sẽ thay thế hoạt động mặc định của máy chủ, đó là chọn một độ dịch chuyển ngẫu nhiên. Nếu giá trị độ dịch chuyển là 0, máy chủ sẽ không đặt độ dịch chuyển vào nhãn thời gian hoặc số thứ tự tương ứng.

Một đầo vào mục mẫu chỉ dẫn SRTP được sử dụng khi có yêu cầu xử lý SRTP.

*class SrtpHintSampleEntry() extends SampleEntry (‘srtp‘) {*

*uint(16)hinttrackversion = 1;*

*uint(16)highestcompatibleversion = 1;*

*uint(32) maxpacketsize;*

*boxadditionaldata[];*

*}*

Các trường và hộp được định nghĩa như cho *RtpHintSampleEntry* (‘*rtp*’) của định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO. Tuy nhiên, một Hộp Xử lý SRTP sẽ được thêm vào *SrtpHintSampleEntry* như là một trong những hộp bổ sung dữ liệu.

#### Hộp xử lý SRTP ‘srpp’ (Box SRTP Process ‘srpp‘)

Loại hộp:‘*srpp’*.

Thuộc đối tượng: SrtpHintSampleEntry

Hình thức sử dụng: bắt buộc.

Số lượng: một.

Hộp Xử lý SRTP có thể chỉ cho máy chủ biết thuật toán SRTP nào sẽ được áp dụng.

*aligned(8) class SRTPProcessBox extends FullBox(‘srpp’, version, 0) {*

*unsigned int(32)encryption\_algorithm\_rtp;*

*unsigned int(32)encryption\_algorithm\_rtcp;*

*unsigned int(32)integrity\_algorithm\_rtp;*

*unsigned int(32)integrity\_algorithm\_rtcp;*

*SchemeTypeBoxscheme\_type\_box;*

*SchemeInformationBoxinfo;*

*}*

Hộp Kiểu Lưu đồvàHộp Thông tin Lưu đồ có cú pháp được định nghĩa ở mục trước dành cho những rãnh đa phương tiện được bảo vệ. Chúng có nhiệm vụ cung cấp các thông số yêu cầu cho việc áp dụng SRTP. Hộp Kiểu Lược đồ được sử dụng để biểu thị việc quản lý khóa và các cơ bản sách an ninh cần thiết cho luồng đó bên cạnh các con trỏ thuật toán đã được định nghĩa được cung cấp bởi SRTPProcessBox. Chức năng quản lý khóa cũng được sử dụng để tạo lập tất cả các thông số SRTP cần thiết như đã được liệt kê trong mục 8.2 của bản đặc tả SRTP. Việc định nghĩa một cách cơ bản xác sơ đồ bảo vệ nằm ngoài phạm vi của định đạng tệp tin này.

Các thuật toán cho mã hóa và bảo vệ tính toàn vẹn được định nghĩa bởi SRTP. Những định danh định dạng được định nghĩa ở đây. Một đầu vào mục gồm bốn dấu trắng (*$20$20$20$20*) có thể được sử dụng để biểu thị rằng việc lựa chọn thuật toán cho mã hóa và bảo vệ tính toàn vẹn được quyết định bởi một quá trình bên ngoài định dạng tệp tin.

|  |  |
| --- | --- |
| **Định dạng** | **Thuật toán** |
| *$20$20$20$20* | Việc lựa chọn thuật toán cho mã hóa và bảo vệ tính toàn vẹn được quyết định bởi một quá trình bên ngoài định dạng tệp tin |
| *ACM1* | Mã hóa sử dụng AES trong Counter Mode với mã dài 128 bit, như đã được định nghĩa trong mục 4.1.1 của bản đặc tả SRTP. |
| *AF81* | Mã hóa sử dụng AES trong mode F8 với độ dài mã 128 bit, như đã được định nghĩa trong mục 4.1.2 của bản đặc tả SRTP. |
| *ENUL* | Mã hóa sử dụng thuật toán TRỐNG như đã được định nghĩa trong mục 4.1.3 của bản đặc tả SRTP. |
| *SHM2* | Bảo vệ tính toàn vẹn sử dụng HMAC-SHA-1 với mã khóa có độ dài 160 bit, như đã được định nghĩa trong mục 4.2.1 trong bản đặc tả SRTP. |
| *ANUL* | Bảo vệ tính toàn vẹn không áp dụng cho RTP (nhưng vẫn áp dụng với RTCP). LƯU Ý: chỉ có hiệu lực đối với integrity\_algorithm\_rtp. |

### Định dạngmẫu

Mỗi mẫu trong một rãnh chỉ dẫn máy chủsẽ tạo ra một hoặc nhiều gói RTP, nhãn thời gian RTP của chúng giống như thời gian mẫu chỉ dẫn. Bởi thế, tất cả các gói tin tạo ra bởi cùng một mẫu sẽ có chung nhãn thời gian. Tuy thế, cho phép yêu cầu máy chủ “nắn “thời gian truyền dẫn thực tế, chẳng hạn để làm mịn tốc độ dữ liệu.

Mỗi mẫu chứa hai vùng: các lệnh để tổ hợp các gói và bất cứ các dữ liệu phụ trợ cần thiết khi gửi các gói tin đó. Lưu ý rằng kích cỡ của mẫu được xác định từ bảng kích cỡmẫu.

*aligned(8) class RTPample {*

*unsigned int(16)packetcount;*

*unsigned int(16)reserved;*

*RTPpacket packets[packetcount];*

*byteextradata[];*

*}*

#### Định dạng Packet Entry

Mỗi gói trong bảng đầu vào mục gói tin có cấu trúc sau đây:

*aligned(8) class RTPpacket {*

*int(32)relative\_time;*

*// the next fields form initialization for the RTP*

*// header (16 bits), and the bit positions correspond*

*bit(2) RTP\_version;*

*bit(1) P\_bit;*

*bit(1) X\_bit;*

*bit(4) CSRC\_count;*

*bit(1) M\_bit;*

*bit(7) payload\_type;*

*unsigned int(16)RTPequenceseed;*

*unsigned int(13)reserved = 0;*

*unsigned int(1)extra\_flag;*

*unsigned int(1) bframe\_flag;*

*unsigned int(1) repeat\_flag;*

*unsigned int(16)entrycount;*

*if (extra\_flag) {*

*uint(32) extra\_information\_length;*

*box extra\_data\_tlv[];*

*}*

*dataentry constructors[entrycount];*

*}*Ngữ nghĩa của các trường trong rãnh chỉ dẫn máy chủ RTPđược định nghĩa dưới đây. Các rãnh chỉ dẫn tiếp nhận RTPsử dụng cùng chung cấu trúc gói trên. Ngữ nghĩa của các trường khi cấu trúc gói được sử dụng trong rãnh chỉ dẫn tiếp nhận RTPđược nêu trong 9.4.1.4.

Trong các rãnh chỉ dẫn máy chủ, trường *relative\_time* “nắn “thời gian truyền dẫn thực tế so với thời gian mẫu. Điều này cho phép mịn hóa lưu lượng.

2 byte tiếp theo thay thế toàn bộ mào đầu RTP; chúng trợ giúp máy chủ trong quá trình tạo thành mào đầu RTP (máy chủ điền vào các trường còn lại). Trong 2 byte đó, các trường*RTP\_version* và *CSRC\_count*được lưu lại tại các rãnh chỉ dẫnmáy chủ (truyền dẫn) và máy chủ điền vào các trường này.

Hạt chuỗi là cơ sở cho số chuỗi RTP. Nếu một rãnh chỉ dẫnchỉ thị gửi đi nhiều bản sao của cùng một gói RTP, thì seed value sẽ có cùng giá trị cho tất cả các gói đó. Thông thường máy chủ sẽ gán một giá trị độ dịch chuyển ngẫu nhiên cho giá trị này (nhưng xem ở trên, phần ‘*sequenceoffset*’).

*extra\_flag* bằng 1 biểu thị có thông tin bổ sung trước các constructor, dưới dạng tập type-length-value.

*extra\_information\_length* biểu thị độ dài tính bằng byte của tất cả các thông tin bổ sung phía trước các constructor, bao gồm cả bốn byte của trường *extra\_information\_length*. Các hộp tiếp theo đó trước các hàm tạo, được gọi là các hộp TLV, có kích cỡ 32 bit. Kích cỡ hộp của bất kỳ hộp TLV nào biểu thị số byte sử dụng thực tế, chứ không phải độ dài yêu cầu chèn thêm cho đủ 32 bit. Giá trị của*extra\_information\_length* bao gồm cả phần padding yêu cầu cho 32 bit.

*rtpoffsetTLV ('rtpo')* cung cấp một giá trị độ dịch chuyển 32-bit nguyên có dấu cho nhãn thời gian RTP thực tế để đặt trong gói. Điều này cho phép các gói tin được đặt trong rãnh chỉ dẫntheo trật tự giải mã, nhưng lạicó các nhãn thời gian trình diễn của chúng ở trong gói được truyền đi theo một trật tự khác. Điều này là cần thiết cho một số MPEG payload.

*bframe\_flag* biểu thị một 'b-frame' có thể thải kiểu được. *repeat\_flag* biểu thị một ‘gói tin lặp lại’, tức là một gói tin được gửi đi và lặp lại một gói tin đã được gửi trước nó. Các máy chủ cần tối ưu việc điều khiển các gói tin này.

#### Định dạng hàm tạo

Có rất nhiều dạng constructor. Mỗi constructor có 16 byte, để làm cho việc lặp lại dễ dàng hơn. Byte đầu tiên là một union discriminator:

*aligned(8) class RTPconstructor(type) {*

*unsigned int(8) constructor\_type = type;*

*}*

*aligned(8) class RTPnoopconstructorextends RTPconstructor(0)*

*{*

*uint(8)pad[15];*

*}*

*aligned(8) class RTPimmediateconstructor extends RTPconstructor(1)*

*{*

*unsigned int(8) count;*

*unsigned int(8) data[count];*

*unsigned int(8) pad[14 - count];*

*}*

*aligned(8) class RTPampleconstructor extends RTPconstructor(2)*

*{*

*signed int(8)trackrefindex;*

*unsigned int(16)length;*

*unsigned int(32)samplenumber;*

*unsigned int(32)sampleoffset;*

*unsigned int(16)bytesperblock = 1;*

*unsigned int(16)samplesperblock = 1;*

*}*

*aligned(8) class RTPampledescriptionconstructor extends RTPconstructor(3)*

*{*

*signed int(8)trackrefindex;*

*unsigned int(16)length;*

*unsigned int(32)sampledescriptionindex;*

*unsigned int(32)sampledescriptionoffset;*

*unsigned int(32)reserved;*

*}*

Mode trung gian cho phép việc chèn các mào đầu đặc thù cho dữ liệu (ví dụ như mào đầu RTP H.261). Với những rãnh chỉ dẫnmà đa phương tiện được gửi ‘in the clear’, thì mục mẫu sẽ quy định số byte sẽ được sao chép từ rãnh đa phương tiện, bằng cách đưa ra số mẫu, data độ dịch chuyển, và độ dài để sao chép. rãnh tham chiếu có thể đánh chỉ số vào bảng rãnh tham chiếu (bắt buộc giá trị dương), đặt tên cho bản thân rãnh chỉ dẫnđó (-1), hoặc cho rãnh đa phương tiệnduy nhất kèm theo (0). (Giá trị 0 vì thế tương đương với giá trị 1.)

*bytesperblock* và *samplesperblock* liên quan việc nén âm thanh, sử dụng một cơ chế xuất hiện trước MP4, trong đó khung âm thanh không hiển thị rõ ràng trong tệp tin. Những trường này có giá trị cố định bằng 1 cho các tệp tin MP4.

Mode *sampledescription* cho phép gửi các miêu tả mẫu (trong đó chứa các miêu tả luồng cơ bản), thông qua tham chiếu, như một phần của gói RTP. Index là chỉ số của *SampleEntry* trong mộtHộp Mô tả Mẫu, và độ dịch chuyển tương ứng với khởi đầu của *SampleEntry* đó.

Đối với những trường hợp phức tạp (ví dụ như mã hóa hay việc sửa lỗi phía trước), dữ liệu được chuyển đổi sẽ được đặt vào trong các mẫu chỉ dẫn, trong trường *extradata*, và rồi mẫu mode tham chiếu tới bản thân rãnh chỉ dẫnđó sẽ được sử dụng.

LƯU Ý:không có yêu cầu bắt buộc các gói tin kế tiếp nhau phải truyền các byte kế tiếp nhau từ luồng của đa phương tiện. Ví dụ, để thỏa mãn đóng gói theo chuẩn RTP của H.261, có lúc yêu cầu byte cuối của một gói cũng sẽ là byte đầu của gói tiếp theo (khi một đường biên khối mã hóa nằm trong một byte).

### Thông tin SDP

Các máy chủ streaming sử dụng RTSP và SDP thông thường sử dụng SDP làm định dạng mô tả và có các mối quan hệ cần thiết giữa thông tin SDP và các luồng RTP, chẳng hạn như việc ánh xạ các dữ liệu ID với các tên MIME. Vì vậy cần chuẩn bị để bộ chỉ dẫn đưa các phân mảnh của thông tin SDP vào tệp tin nhằm hỗ trợ máy chủ trong việc hình thành miêu tả SDP đầy đủ. Lưu ý rằng có những đầu vào mục SDP bắt buộc mà máy chủ cũng sẽ phải tạo ra. Thông tin ở đây chỉ là một phần.

Thông tin SDP được định dạng như là một tập các hộp bên trong các Hộp Dữ liệu Người dùng, ở cả mức movie vàmức rãnh.Chữ trong hộp SDP ở mức movie cần được đặt trước bất cứ dòng đa phương tiện cụ thể nào (trước chữ 'm=' đầu tiên trong tệp tin SDP).

#### Thông tin SDP movie

Ở mức movie, trong hộp dữ liệu người dùng (*'udta'*), một hộp chứa thông tin chỉ dẫn có thể có dạng:

*aligned(8) class moviehintinformation extends box(‘hnti’) {*

*}aligned(8) class rtpmoviehintinformation extends box(‘rtp ‘) {*

*uint(32) descriptionformat = ‘sdp ‘;*

*charsdptext[];*

*}*

Hộp Thông tin Chỉ dẫn có thể chứa thông tin cho nhiều giao thức; chỉ giao thức RTP được định nghĩa ở đây. Hộp RTP có thể chứa thông tin cho nhiều miêu tả định dạng khác nhau; chỉ SDP được định nghĩa ở đây. *sdptext* được định dạng chuẩn bằng một loạt các dòng, mỗi dòng kết thúc bằng <crlf> theo yêu cầu của SDP.

#### Thông tin rãnh SDP

Ởmức rãnh, cấu trúc cũng tương tự; tuy vậy, chúng ta đều đã biết rằng rãnh này là một Rãnh chỉ dẫn RTP, từ miêu tả mẫu. Bởi thế hộp con này chỉ định ra định dạng của miêu tả.

*aligned(8) class trackhintinformation extends box(‘hnti’) {*

*}*

*aligned(8) class rtptracksdphintinformation extends box(‘sdp ‘) {*

*charsdptext[];*

*}*

*sdptext* được định dạng cơ bản xác bằng một loạt các dòng, mỗi dòng được kết thúc bằng <crlf>, theo yêu cầu của SDP.

### Thông tin thống kê

Cùng với các thống kê trong mào đầu đa phương tiện chỉ dẫn, hinter cũng có thể đặt dữ liệu bổ sung vào một hộp thống kê, nằm trong hộp rãnh user-data. Đây là một hộp chứa với rất nhiều các hộp con mà nó có thể chứa được.

*aligned(8) class hintstatisticsbox extends box(‘hinf’) {*

*}*

*aligned(8) class hintBytesSent extends box(‘trpy’) {*

*uint(64) bytessent; } // total bytes sent, including 12-byte RTP headers*

*aligned(8) class hintPacketsSent extends box(‘nump’) {*

*uint(64) packetssent; } // total packets sent*

*aligned(8) class hintBytesSent extends box(‘tpyl’) {*

*uint(64) bytessent; } // total bytes sent, not including RTP headers*

*aligned(8) class hintBytesSent extends box(‘totl’) {*

*uint(32) bytessent; } // total bytes sent, including 12-byte RTP headers*

*aligned(8) class hintPacketsSent extends box(‘npck’) {*

*uint(32) packetssent; } // total packets sent*

*aligned(8) class hintBytesSent extends box(‘tpay’) {*

*uint(32) bytessent; } // total bytes sent, not including RTP headers*

*aligned(8) class hintmaxrate extends box(‘maxr’) {// maximum data rate*

*uint(32) period; // in milliseconds*

*uint(32) bytes; } // max bytes sent in any period ‘period’ long*

*//including RTP headers*

*aligned(8) class hintmediaBytesSent extends box(‘dmed’) {*

*uint(64) bytessent; } // total bytes sent from trackmedia*

*aligned(8) class hintimmediateBytesSent extends box(‘dimm’) {*

*uint(64) bytessent; } // total bytes sent immediate mode*

*aligned(8) class hintrepeatedBytesSent extends box(‘drep’) {*

*uint(64) bytessent; } // total bytes in repeated packets*

*aligned(8) class hintminrelativetime extends box(‘tmin’) {*

*int(32) time; } // smallest relative transmission time, milliseconds*

*aligned(8) class hintmaxrelativetime extends box(‘tmax’) {*

*int(32) time; } // largest relative transmission time, milliseconds*

*aligned(8) class hintlargestpacket extends box(‘pmax’) {*

*uint(32) bytes; } // largest packet sent, including RTP header*

*aligned(8) class hintlongestpacket extends box(‘dmax’) {*

*uint(32) time; } // longest packet duration, milliseconds*

*aligned(8) class hintpayloadID extends box(‘payt’) {*

*uint(32) payloadID; // payload ID used in RTP packets*

*uint(8) count;*

*charrtpmap\_string[count];*

*}*

LƯU Ý:Không phải tất cả các hộp con này đều xuất hiện và rằng có thể có nhiều hộp 'maxr', bao phủ các giai đoạn khác nhau.

## Định dạngrãnh chỉ dẫnALC/LCT và FLUTE

### Giới thiệu

Định dạngtệp tin này hỗ trợ multicast/broadcast của các tệp tin có mã bảo vệ chữa lỗi FEC. Các tệp tin sẽ phát đi được lưu giữ trong một tệp tin (được định nghĩa bởi định dạng tệp tin này) và Hộp Siêu dữ liệu được bổ sung thêm các thông tin về phương thức phân chia các tệp tin ra thành các biểu tượng nguồn. Đối với mỗi khối nguồn FEC encoding, các biểu tượng chẵn lẻ bổ sung có thể được tính toán trước và lưu trữ như các hạng mục dự trữ FEC. Việc phân chia này phụ thuộc vào cơ chế FEC, kích cỡ gói tin mục tiêu và mức FEC overhead mong muốn. Các biểu tượng nguồn được hình thành trước có thể được lưu trữ như là các tệp tin reservoir item nhằm giảm thiểu tối đa việc trùng thông tin trong các container tệp tin nhất là với MBMS-FEC. Việc điều khiển thực tế sẽ được điều hành bởi các rãnh chỉ dẫncó chứa các lệnh máy chủ thuận lợi cho việc đóng gói nguồn và các biểu tượng FEC vào các gói tin.

Các rãnh chỉ dẫnFD đã được thiết kế dành cho các giao thức ALC/LCT (AsynchronousLayeredCoding/LayeredCoding Transport) và FLUTE (FileDeliveryoverUnidirectionalTransport). LCT cung cấp mức hỗ trợ ở lớp vận tải cho các giao thức phân phối nội dung và các giao thức phân phối luồng một cách tin cậy. ALC là một giao thức khởi tạo của khối tạo LCT, và nó được sử dụng như giao thức cơ sở cho vô số phân phối multicast mở rộng được và tin cậy của các đối tượng nhị phân bất kỳ. FLUTE được xây dựng trên nền ALC/LCT và định nghĩa một giao thức phân phối tệp tin theo một chiều.

FLUTE định nghĩa một bảng vận chuyển tệp tin(File Delivery Table - FDT), trong đó chứa siêu dữ liệu gắn với các tệp tin được chuyển đi trong phiên ALC/LCT, và cung cấp các cơ chế cho việc phân phối in-band và cập nhật FDT. Đối ngược lại, ALC/LTC phụ thuộc vào các phương thức khác cho việc phân phối out-of-band tệp tinsiêu dữ liệu, ví dụ: một bản mềm hướng dẫn sử dụng dịch vụ thường được phân phối tới các khách hàng rất lâu trước phiên ALC/LTC kết hợp với các phần cập nhật sẽ được gửi trong phiên ALC/LTC.

Việc phân vùngtệp tin và các dự trữ FEC có thể được sử dụng độc lập với các FD rãnh chỉ dẫnvà ngược lại. Việc phân vùng tệp tin hỗ trợ việc thiết kế các rãnh chỉ dẫnvà cho phép các rãnh chỉ dẫnthay thế, ví dụ với các mức FEC overhead khác, để tái sử dụng cùng các biểu tượng FEC đó. Chúng cũng cung cấp các cách thức truy nhập tới các biểu tượng nguồn và các biểu tượng FEC bổ sung một cách độc lập cho việc sửa lỗi sau khi phân phối, có thể được thực hiện thông quan ALC/LCT hoặc FLUTE hoặc out-of-band thông qua một giao thức khác. Nhằm làm giảm độ phức tạp khi máy chủ tuân theo chỉ dẫn của rãnh chỉ dẫn, rãnh chỉ dẫnchỉ trực tiếp đến dải dữ liệu của các hạng mục hoặc các dữ liệu đã được sao chép vào các mẫu chỉ dẫn.Khuyến nghị các máy chủ gửi một bộ biểu tượng FEC khác cho mỗi lần gửi lại của một tệp tin.

Cú pháp cho việc sử dụng Hộp Siêu dữ liệu làm container tệp tin cho các tệp tin nguồn được định nghĩa trong 8.11, partition, tệp tin và dự trữ FEC được định nghĩa trong 8.13, còn cú pháp cho các FD rãnh chỉ dẫnđược định nghĩa trong 9.2.

### Nguyên tắc thiết kế

Việc hỗ trợ chuyển tệp tin được thiết kế nhằm tối ưu quá trình phát dẫn của máy chủ bằng cách cho phép các máy chủ ALC/LCT hoặc FLUTE tuân theo các chỉ dẫn đơn giản. Để phát một phiên chỉ cần tuân theo một loạt các chỉ dẫn đã được định ra trước trên mỗi kênh. Định dạngtệp tin này cho phép việc lưu trữ các source block được tính toán trước và việc phân chia biểu tượng, nghĩa là các tệp tin có thể được chia nhỏ thành các biểu tượng vừa với kích cỡ gói tin dự định, và tính toán trước một số lượng cụ thể biểu tượng FEC có thể được sử dụng cho sửa chữa sau phiên. Định dạngtệp tin này cũng cho phép lưu trữ các chỉ dẫn truyền dẫn phiên thay thế ALC/LCT hoặc FLUTE có thể dẫn tới kết quả cuối cùng tương đương. Những thay thế như vậy có thể được để dành cho những điều kiện kênh truyền khác bởi vì mức bảo vệ FEC cao hơn hoặc thậm chí bởi việc sử dụng các cơ chế sửa lỗi khác. Các lựa chọn thay thế phiên khác nhau có thể cùng chỉ tới một tập hợp biểu tượng chung. Các rãnh chỉ dẫncó tính linh động và có thể được sử dụng để tổng hợp các phân mảnh FDT vàđan xen (interleaving) các phân mảnh như thế với việc truyền dẫn các đối tượng thực sự. Một vài rãnh chỉ dẫncó thể được kết hợp vào một hoặc nhiều phiên liên quan tới việc truyền dẫn đồng thời trên nhiều kênh.

Một điều quan trọng là cần có định nghĩa khác biệt giữa các phiên cho truyền dẫn và việc lập lịch của các phiên như thế. Các tệp tin của ALC/LCT và FLUTE máy chủ chỉ giải quyết vấn đề tối ưu của quá trình truyền dẫn của máy chủ. Nhằm đảm bảo mức sử dụng và độ linh động tối đa của các phiên đã được định nghĩa trước như thế, tất cả các chi tiết liên quan đến các địa chỉ lập lịch... được giữ bên ngoài định nghĩa của định dạng tệp tin này. Các ứng dụng lập lịch bên ngoài quyết định những chi tiết như thế, điều này vốn không quan trọng đối với việc tối ưu hóa các phiên truyền dẫn.Cụ thể, những thông tin sau nằm ngoài phạm vị của bản định dạng tệp tin này: lập lịch, các địa chỉ và cổng mục tiêu, địa chỉ và cổng nguồn, và các định danh phiên truyền (TSI).

Số lượng mẫu gắn với mẫu của một rãnh chỉ dẫntruyền dẫn tệp tin cungcấp một chuỗi có đánh số thứ tự.Thời gian mẫu của rãnh chỉ dẫncho biết thời gian gửi của các gói ALC/LCT hoặc FLUTE ở một bitrate mặc định. Tùy thuộc vào tốc độ truyền bitrate thực tế, ALC/LCT hoặc FLUTE máy chủ có thể áp dụng thang thời gian tuyến tính. Thời gian mẫu có thể làm đơn giản hóa quá trình lập lịch, nhưng việc gửi các gói tin ALC/LCT hoặc FLUTE kịp thời phụ thuộc vào máy chủ.

Hình vẽ sơ đồ nguyên lý của một tệp tin bao gồm ba rãnh chỉ dẫncó thể thay thế lẫn nhau cùng với FEC overhead khác nhau cho một tệp tin nguồn được biểu thị trong Hình 6. Trong ví dụ này, mỗi khối nguồn bao gồm chỉ một khối con.

Hình 4 - Các FEC overhead khác nhau của một tệp tin nguồn được cung cấp bởi các rãnh chỉ dẫn có thể thay thế nhau.

Tệp tin nguồn trong hình trên được chia thành 2 khối nguồn chứa các biểu tượng có kích cỡ cố định. Các biểu tượng FEC dư thừa được tính toán cho cả hai khối nguồn và được lưu trữ như các hạng mục dự trữ FEC. Bởi vì các rãnh chỉ dẫnchỉ đến cùng các item trong tệp tin đó nên không có sự lặp lại thông tin. Các biểu tượng nguồnvà các dự trữ FEC ban đầu cũng có thể được sử dụng bởi các máy chủ chữa lỗi không sử dụng rãnh chỉ dẫn.

### Định dạng miêu tả mẫu

#### Định nghĩa

FD rãnh chỉ dẫnlà các rãnh có *handler\_type* là *'hint'* và entry\_format là *'fdp'* trong Hộp Mô tả Mẫu. Đầu vào mục mẫu chỉ dẫn FD nằm trong Hộp Mô tả Mẫu (*'stsd'*).

#### Cú pháp

*class FDHintSampleEntry() extends SampleEntry ('fdp ') {*

*unsigned int(16)hinttrackversion = 1;*

*unsigned int(16)highestcompatibleversion = 1;*

*unsigned int(16)partition\_entry\_ID;*

*unsigned int(16)FEC\_overhead;*

*Boxadditionaldata[]; //optional*

*}*

#### Ngữ nghĩa

*partition\_entry\_ID* biểu thị đầu vào mục phân vùng trong Hộp Thông tin Hạng mục FD. Giá trị bằng không biểu thị rằng không có đầu vào mục phân vùng gắn với mục mẫu này, ví dụ cho FDT. Nếu như FD rãnh chỉ dẫntương ứng chỉ chứa dữ liệu overhead, giá trị này chỉ cần thể hiện đầu vào mục phân vùng mà dữ liệu overhead của nó đang được nhắc tới.

*FEC\_overhead* là một giá trị cố định 8.8 biểu thị phần trăm bảo vệ overhead được sử dụng bởi (các) mẫu chỉ dẫn. Mục đích của việc đưa ra giá trị này là để cung cấp các đặc tính nhằm giúp máy chủ lựa chọn một nhóm phiên (và các FD rãnh chỉ dẫntương ứng). Nếu như FD rãnh chỉ dẫntương ứng chỉ chứa các dữ liệu overhead giá trị này nên chỉ tới mức bảo vệ overhead đạt được bằng cách sử dụng tất cả các FD rãnh chỉ dẫntrong nhóm phiên có chứa FD rãnh chỉ dẫnđang xem xét.

Các trường *hinttrackversion* và *highestcompatibleversion* cũng được diễn giải như đầu vào mẫu chỉ dẫn RTP được miêu tả trong 9.1.2. Một Hộp Đầu vào mục Thang thời gian có thể được cung cấp như là dữ liệu phụ trợ. Nếu nó không được cấp, sẽ không có chỉ thị cho việc định thời gian của các gói.

Các tệp tin đầu vào mục cần thiết cho một FDT hoặc một hướng dẫn dịch vụ điên tử có thể được tạo ra bằng cách quan sát tất cả các mục mẫu của một rãnh chỉ dẫnvà các Hộp Thông tin Hạng mục tương ứng của các hạng mục được tham chiếu đến bởi những đầu vào mục phân vùng ID ở trên. Các mục mẫu sẽ không được chứa trong rãnh chỉ dẫnnếu chúng không được bất cứ mẫu nào tham chiếu đến.

### Định dạng mẫu

#### Đối tượng chứa mẫu

Mỗi mẫu FD trong rãnh chỉ dẫnsẽ tạo ra một hoặc nhiều gói FD.

Mỗi mẫu chứa hai trường: các lệnh để tổ hợp các gói và bất cứ các dữ liệu phụ trợ cần thiết khi gửi các gói tin đó (ví dụ như các biểu tượng mã hóa được sao chép vào trong mẫu thay vì để ở các hạng mục của source tệp tin hay FEC). Lưu ý rằng kích cỡ của mẫu được biết thông qua bảng kích cỡ mẫu.

*aligned(8) class FDsample extends Box(‘fdsa’) {*

*FDPacketBox packetbox[]*

*ExtraDataBox extradata;//optional*

*}*

Số mẫu của các mẫu FD định ra thứ tự mà chúng sẽ được xử lý tại máy chủ. Tương tự như thế, các FD packet trong mỗi mẫu FD cũng cần được xuất hiện theo thứ tự mà chúng sẽ được xử lý. Nếu trong đầu vào mục mẫu chỉ dẫn FD có Hộp Đầu vào mục Thang thời gian, thì các mẫu time được định nghĩa và cung cấp thời gian gửi tương đối của các gói với tốc độ bitrate mặc định. Tùy thuộc vào tốc độ bitrate thu phát thực sự, máy chủ có thể áp dụng time scaling tuyến tính. Sample time có thể làm đơn giản hóa quá trình lập lịch, nhưng việc gửi các gói tin kịp thời phụ thuộc vào máy chủ.

#### Định dạngđầu vào mục gói tin

Mỗi gói tin trong mẫu FD có cấu trúc như sau (Tham khảo: RFC 3926,3450,3451):

*aligned(8) class FDpacketBox extends Box(‘fdpa’) {*

*LCTheaderTemplateLCT\_header\_info;*

*unsigned int(16) entrycount1;*

*LCTheaderExtension header\_extension\_constructors[ entrycount1 ];*

*unsigned int(16) entrycount2;*

*dataentrypacket\_constructors[ entrycount2 ];*

*}*

Thông tin của mào đầu LCT chứa các khuôn mẫu mào đầu LCT cho các gói FD hiện tại. Các hàm tạo mở rộng mào đầu là các cấu trúc được sử dụng để tạo dựng nên các mở rộng mào đầu LCT. Các hàm tạo gói được sử dụng để cấu trúc nên FEC payload ID và các biểu tượng nguồn trong một gói FD.

#### Định dạngkhuôn mẫu mào đầu LCT

Khuôn mào đầu LCT được định nghĩa như sau:

*aligned(8) class LCTheaderTemplate {*

*unsigned int(1) sender\_current\_time\_present;*

*unsigned int(1) expected\_residual\_time\_present;*

*unsigned int(1) session\_close\_bit;*

*unsigned int(1) object\_close\_bit;*

*unsigned int(4) reserved;*

*unsigned int(16) transport\_object\_identifier;*

*}*

Nó có thể được sử dụng bởi máy chủ để hình thành nên mào đầu LCT cho một gói tin. Lưu ý rằng một vài phần của mào đầu tùy thuộc vào cơ bản sách của máy chủ và không được bao gồm trong khuôn mẫu. Độ dài của một số trường cũng phụ thuộc vào các bitmào đầu LTC được chỉ định bởi máy chủ. Máy chủ cũng có thể cần thay đổi giá trị của định danh đối tượng truyền tải (TOI).

#### Định dạng hàm tạo mở rộng mào đầu LCT

Định dạng của hàm tạo mở rộng mào đầu LCT được định nghĩa như sau:

*aligned(8) class LCTheaderextension {*

*unsigned int(8) header\_extension\_type;*

*if (header\_extension\_type> 127) {*

*unsigned int(8) content[3];*

*}*

*else {*

*unsigned int(8) length;*

*if (length > 0) {*

*unsigned int(8) content[(length\*4) - 2];*

*}*

*}*

Giá trị dương của trường *length*xác định độ dài của nội dung hàm tạo là bội số của các word 32 bit. Giá trị 0 ngụ ý mào đầu được tạo ra bởi máy chủ.

Ứng dụng và các nguyên tắc cho mở rộng mào đầu LCT được định nghĩa trong RFC 3451 (LCT RFC). *header\_extension\_type* chứa giá trị kiểu mở rộng mào đầu LTC (HET).

Các giá trị HET từ 0 đến 127 được sử dụng cho các mở rộng có độ dài biến đổi (bội số word 32 bit). Các giá trị HET từ 128 đến 255 được sử dụng cho những mở rộng có độ dài cố định (một word 32 bit). Nếu *header\_extension\_type* nhỏ hơn128,thì trường độ dài tương ứng với Độ dài Mở rộng Mào đầu LCT(HEL)được định nghĩa trongRFC 3451. Trường content luôn tương ứng với Nội dung Mở rộng Mào đầu (HEC).

LƯU Ý: Một máy chủ có thể xác định các gói bao gồm cả FDT bằng cách quan sát xem có sự hiện diện của EXT\_FDT hay không (*header\_extension\_type*== 192).

#### Định dạng hàm tạo gói tin

Có rất nhiều dạng hàm tạo (constructor). Mỗi hàm tạo có 16 byte để làm cho việc lặp lại dễ dàng hơn.Byte đầu tiên là một union discriminator. Hàm tạo gói này được sử dụng để đưa FEC payload ID cũng như các biểu tượng nguồn và các biểu tượng chẵn lẻ vào trong gói FD.

*aligned(8) class FDconstructor(type) {*

*unsigned int(8)constructor\_type = type;*

*}*

*aligned(8) class FDnoopconstructor extends FDconstructor(0){*

*unsigned int(8)pad[15];*

*}*

*aligned(8) class FDimmediateconstructor extends FDconstructor(1){*

*unsigned int(8)count;*

*unsigned int(8)data[count];*

*unsigned int(8)pad[14 - count];*

*}*

*aligned(8) class FDsampleconstructor extends FDconstructor(2){*

*signed int(8)trackrefindex;*

*unsigned int(16)length;*

*unsigned int(32)samplenumber;*

*unsigned int(32)sampleoffset;*

*unsigned int(16)bytesperblock = 1;*

*unsigned int(16)samplesperblock = 1;*

*}*

*aligned(8) class FDitemconstructor extends FDconstructor(3){*

*unsigned int(16)item\_ID;*

*unsigned int(16)extent\_index;*

*unsigned int(64)data\_offset; //offset in byte within extent*

*unsigned int(24)data\_length;//non-zero length in byte within extent or*

*//if (data\_length==0) rest of extent*

*}*

*aligned(8) class FDitemconstructorLarge extends FDconstructor(5) {*

*unsigned int(32) item\_ID;*

*unsigned int(32) extent\_index;*

*unsigned int(64)data\_offset;//offset in byte within extent*

*unsigned int(24)data\_length;//non-zero length in byte within extent or*

*//if (data\_length==0) rest of extent*

*}*

*aligned(8) class FDxmlboxconstructor extends FDconstructor(4){*

*unsigned int(64)data\_offset; //offset in byte within XMLBox or BinaryXMLBox*

*unsigned int(32)data\_length;*

*unsigned int(24)reserved;*

*}*

#### Hộp Dữ liệu Phụ trợ(Extra Data Box)

Mỗi mẫu của một FD rãnh chỉ dẫncó thể bao gồm các thông tin phụ trợ được lưu trữ trong mỗi hộp dữ liệu phụ trợ:

*aligned(8) class ExtraDataBox extends Box(‘extr’) {*

*FECInformationBoxfeci;*

*bit(8) extradata[];*

*}*

#### Hộp Thông tin FEC (FEC Information Box)

##### Định nghĩa

Loại hộp:‘*feci*’.

Thuộc đối tượng:Hộp Dữ liệu Phụ trợ (*‘extr’*).

Hình thức sử dụng: không bắt buộc.

Số lượng: không hoặc một.

Hộp thông tin FEC lưu giữ FEC encoding ID, FEC instance ID và FEC payload ID cần thiết khi gửi gói FD.

##### Cú pháp

*aligned(8) class FECInformationBox extends Box('feci') {*

*unsigned int(8) FEC\_encoding\_ID;*

*unsigned int(16)FEC\_instance\_ID;*

*unsigned int(16)source\_block\_number;*

*unsigned int(16)encoding\_symbol\_ID;*

*}*

##### Ngữ nghĩa

*FEC\_encoding\_ID* xác định cơ chế FEC encoding và is subject to IANA registration (see RFC 5052), trong đó (i) giá trị bằng không tương ứng với “Compact No-Code FEC scheme” còn được biết đến là “Null-FEC” (RFC 3695); (ii) giá trị một tương ứng với “MBMS FEC “(3GPP TS 26.346); (iii) đối với các giá trị trong dải từ 0 tới 127, kể cả 2 giá trị biên, cơ chế FEC được xác định hoàn toàn, trong khi đối với các giá trị trong dải từ 128 tới 255, kể cả 2 giá trị biên, cơ chế FEC chưa được xác định hoàn toàn.

*FEC\_instance\_ID* cung cấp một định danh cụ thể của bộ mã hóa FEC được sử dụng cho lược đồ Under-Specified FEC. Giá trị này cần được đặt bằng 0 đối với các lược đồ Fully-Specified FEC và sẽ bị bỏ qua khi phân tích một tệp tin với FEC\_encoding\_IDtrong dải 0 đến 127. FEC\_instance\_ID được kiểm tra bởi FEC\_encoding\_ID. Xem RFC 5052 để biết chi tiết hơn.

*source\_block\_number* xác định xem (các) biểu tượng mã hóa của gói FD đó đã được tạo ra từ khối nguồn nào của object đó.

*encoding\_symbol\_ID* xác định (các) biểu tượng mã hóa cụ thể nào được tạo ra từ khối nguồn được mang trong gói FD.

## Định dạngrãnh chỉ dẫn truyền tảiMPEG-2

### Giới thiệu

MPEG-2 TS (Transport Stream) là một luồngphức hợp có thể chứa một hoặc nhiều chương trình, bao gồm âm thanh, hình ảnh và các đa phương tiện khác. Định dạngtệp tin này hỗ trợ việc lưu trữMPEG-2 TS trong một rãnh chỉ dẫn. Một rãnh chỉ dẫnMPEG-2 có thể được sử dụng cho cả lưu trữ các gói TS nhận được (như một rãnh chỉ dẫn tiếp nhận), và như một rãnh chỉ dẫn máy chủđược sử dụng để khởi tạoMPEG-2 TS.

Định nghĩa của rãnh chỉ dẫnMPEG-2 TS hỗ trợ “các chỉ dẫn được tính toán trước “. Các chỉ dẫn được tính toán trước không sử dụng các dữ liệu được chứa trong đó được tham chiếu tới các rãnh khác, mà thay vào đó sử dụng các gói MPEG-2 TS được lưu giữ trong đó. Điều này cho phép việc tái sử dụng các gối MPEG-2 TS được lưu giữ trong một tệp tin riêng biệt. Thêm vào đó, các chỉ dẫn được tính toán trước cho phép các hoạt động thu một cách đơn giản.

Cùng với các chỉ dẫn được tính toán trước, cũng có thể đưa các dữ liệu đa phương tiện tham chiếu tới các rãnh đa phương tiệnvào trong các mẫu chỉ dẫn. Việc biến đổi một luồng truyền tải nhận được thành rãnh đa phương tiệnsẽ cho phép các thiết bị đọc hiện tại tương thích với những phiên bản trước đó của định dạng tệp tinđa phương tiệncơ sở theo chuẩn ISO để xử lý các tệp tin đã đươc thu lạicũng như các định dạng đa phương tiện cũng được hỗ trợ. Việc lưu giữ các mào đầu truyền tải ban đầu cung cấp các thông tin giá trị trong việc khoanh vùng lỗi và tái tạo luồng truyền tải ban đầu.

### Nguyên tắc thiết kế

Nguyên tắc thiết kế của định dạng rãnh chỉ dẫnMPEG-2 TS như sau.

Một chuỗi các mẫu trong một MPECG-2 rãnh chỉ dẫnlà một tập các gói MPEG-2 TS được tính toán và xây dựng trước. Các gói được tính toán trước là những gói TS được lưu giữ không thay đổi trong trường hợp nhận hoặc sẽ được gửi đi y nguyên. Điều này là vô cùng quan trọng khi dữ liệu không thể bị chia tách và các luồng cơ bản không thể khởi tạo được - ví dụ khi luồng truyền tải được mã hóa và không cho phép được lưu trữ ở dạng không mã hóa. Bởi vậy, khả năng lưu giữ MPEG-2 TS như trong các rãnh chỉ dẫnlà cần thiết. Các gói được tạo dựng (constructed packet) sử dụng cùng một cách tiếp cận như Rãnh chỉ dẫn RTP, chẳng hạn như mẫu chứa các lệnh để cho máy chủ streaming hình thành các gói tin. Các dữ liệu đa phương tiện thực sự được chứa trong các rãnh khác. Một tham khảo rãnh dạng 'hint' được sử dụng.

#### Tái sử dụng luồng truyền tải

Việc tái sử dụng cácthành tố TS được mong muốn và bởi thế có một cơ chế bổ sung để bao quát nhiều dạng khác nhau của TS recoding hiện có. Những hình thức thu này có thể bao gồm không chỉ các gói TS nhưng có các dữ liệu kèm phía trước hoặc phía sau mỗi gói TS. Một trường hợp cụ thể cho dữ liệu kèm phía trước là một nhãn thời gian 4 byte phía trước mỗi gói TS để kiểu bỏ jitter của mỗi hệ thống truyền dẫn. Một trường hợp cụ thể minh họa dữ liệu gắn sau là FEC khi một gói TS được truyền trong một kênh dễ gây lỗi.

#### Định thời

MPEG-2 TS định ra một đồng hồ cho mỗi chương trình, chạy ở 27 MHz, sampling value được vận chuyển như là PCR trong TS để phục hồi xung nhịp đồng hồ. Timescale của MPEG-2 TS rãnh chỉ dẫnđược khuyến nghị ở giá trị 90000, hoặc một số nguyên là bội của giá trị đó.

Thời gian decoding của một mẫu trong một MPEG-2 TS rãnh chỉ dẫnlà thời gian nhận/truyền phát của bit đầu tiên của gói đó hoặc nhóm gói được suy ra từ các PCR nhãn thời gian của TS đó, bởi nếu PCR time được sử dụng, có thểgiả sử là tuyến tính piece-wise và bảng '*stts*' nhỏ gọn đáng kể. Hộp tùy chọn 'tsti' trong miêu tả mẫu có thể được sử dụng để thông báo liệu thời gian tiếp nhận có hay không có việc sử dụng phục hồi đồng hồ khi rãnh chỉ dẫnđó là một rãnh chỉ dẫn tiếp nhận. PCR timing được giả sử trong trường hợp của rãnh chỉ dẫn máy chủ.

LƯU Ý: Khi có nhiều gói trong một mẫu, chúng không thể được cấp các độ dịch chuyển thời gian truyền dẫn độc lập.

#### Nhóm gói tin

Định dạng mẫu cho các rãnh chỉ dẫn luồng truyền tải MPEG-2 cho phép nhiều gói tin TS được nhóm chung vào một mẫu. Những ứng dụng cụ thể, chẳng hạn một vài ứng dụng IPTV, truyền các gói TS trong RTP. Chỉ cần một nhãn thời gian nhậnlà có thể suy ra được tất cả các gói TS trong một gói RTP.Một ứng dụng khác dùng để chứa các gói TS trong một mẫu là SPTS, trong đó một mẫu chứa tất cả các gói TS cho một GoP.Trong trường hợp này tất cả các mẫu là một điểm truy nhập ngẫu nhiên.

Lưu ý rằng việc truy nhập ngẫu nhiên tới tất cả các gói TS là không khả thi nếu dùng định dạng tệp tin này khi sử dụng nhiều gói TS trong một mẫu.

Trong trường hợp MPTS, chỉ một gói trên một mẫu được sử dụng. Điều này giúp thuận lợi trong việc sử dụng cơ chế nhóm mẫu trên cơ sở per-packet.

#### Các điểm truy nhập ngẫu nhiên

Một mẫuđồng bộlà một điểm ở đó việc xử lý một rãnh có thể bắt đầu mà không có lỗi. Cả MPTS và SPTS đều được hỗ trợ bởi các MPEG-2 TS rãnh chỉ dẫn, tuy vậy một điểm truy nhập ngẫu nhiên, được đánh dấu như một mẫu đồng bộ, thông thường được định nghĩa cho SPTS, nơi nó xác định mở đầu của một gói có chứa byte đầu tiên của một đơn vị truy nhập đa phương tiên có thể giải mã độc lập được (ví dụ hình ảnh MPEG-2 I-frame hoặc các ảnhMPEG-4 AVC IDR) của một luồng sử dụng differential coding. Đối với MPTS, bảng mẫu đồng bộ thông thường sẽ xuất hiện nhưng để trống, biểu thị rằng không có điểm nào trên rãnh mà tại đó việc xử lý toàn bộ rãnh có thể bắt đầu mà không gặp lỗi. Khuyến nghị đưa PSI/SI vào trong Sample Description như thế thì việc truy nhập ngẫu nhiên thực sự với chỉ dữ liệu đa phương tiện có thể diễn ra.

Lưu ý rằng trong trường hợp của MPTS, có tồn tại bảng mẫu đồng bộ nhưng đó là bảng mẫu trống (điều đó có nghĩa rằng không có mẫu nào là mẫu đồng bộ).

Cũng lưu ý rằng trong trường hợp SPTS, các mẫu bao gồm cả các gói TS cần phải có một điểm đồng bộ (ví dụ GoP boundary0 ở khởi đầu của mẫu. Bảng mẫu đồng bộ sau đó sẽ đánh dấu các mẫu có điểm đồng bộ (chẳng hạn như bắt đầu của các GoP); nếu bảng mẫu đồng bộ không xuất hiện, tất cả các mẫu đều là các điểm đồng bộ. Nếu bảng mẫu đồng bộ xuất hiện nhưng trống, các vị trí mẫu đồng bộ là không xác định và có thể không ở phần đầu của các mẫu.

LƯU Ý: Một ứng dụng đang tìm kiếm một key frame có thể bắt đầu đọc ở vị trí đó, nhưng thông thường nó cũng phải đọc các gói MPEG-2 TS sâu hơn (theo định dạng tệp tin này thì chúng là các subsequent mẫu) như thế thì bộ giải mã có thể giải mã hoàn toàn frame.

#### Cách dùng rãnh chỉ dẫntiếp nhận

Các rãnh chỉ dẫn tiếp nhậncó thể được sử dụng khi một hoặc nhiều gói tin củaluồng dữ liệu được ghi lại. Chúng thể hiện thứ tự, thời gian nhận và nội dung của các gói tin đã nhận được cùng với những thông tin khác.

LƯU Ý 1: Các thiết bị trình diễncó thể tái tạo lại luồng dữ liệu nhận được đó dựa trên các rãnh chỉ dẫn tiếp nhậnvà xử lý luồng dữ liệu được tái tạo lại đó như thế nó là mới được nhận.

Các rãnh chỉ dẫn tiếp nhậncó cùng cấu trúc như các rãnh chỉ dẫncho các máy chủ.

Định dạngcủa các mẫu chỉ dẫn tiếp nhận được biểu thị bằng miêu tả mẫu cho rãnh chỉ dẫn tiếp nhận. Mỗi một giao thức có định dạng mẫu chỉ dẫn tiếp nhận và tên của riêng nó.

LƯU Ý 2: Các máy chủ sử dụng các rãnh chỉ dẫn tiếp nhậnnhư là các chỉ dẫn cho việc gửi các luồng nhận được cần phải xử lý những suy giảm chất lượng có thể xảy ra đối với các luồng đã nhận được, chẳng hạn như jitter và mất gói của đường truyền và đảm bảo rằng những giới hạn của các giao thức đó và các dịnh dạng dữ liệu được lưu giữ phải được tuân thủ, bất chấp những suy giảm chất lượng có thể xảy ra với các luồng đã nhận được.

LƯU Ý 3: Với những rãnh chỉ dẫn máy chủ, định dạng mẫu của rãnh chỉ dẫn tiếp nhậncó thể cho phép việc xây dựng các gói bằng cách lấy dữ liệu từ các rãnh khác theo tham chiếu. Những rãnh khác có thể là các rãnh chỉ dẫnhoặc các rãnh đa phương tiện. Form cơ bản xác của các con trỏ đó được định nghĩa bởi định dạng mẫu cho giao thức đó, nhưng nhìn chung chúng bao gồm bốn thông tin: một chỉ số tham khảo rãnh, một số mẫu, một độ dịch chuyển và một length. Một vài trong số chúng có thể được ngầm định với một số giao thức nhất định. Các 'con trỏ' này luôn trỏ tới nguồn dữ liệu thực sự, có nghĩa là các dữ liệu tham khảo gián tiếp không được cho phép. Nếu một rãnh chỉ dẫnđược xây dựng trên nền ('on top') của một rãnh chỉ dẫnkhác, thì rãnh chỉ dẫnthứ hai phải có tham chiếu trực tiếp tới (các) rãnh đa phương tiệnđược sử dụng bởi rãnh chỉ dẫnthứ nhất nơi dữ liệu từ những rãnh đa phương tiệnđó được đặt trong luồng.

Nếu dữ liệu nhận được được rút ra cho các rãnh đa phương tiện,quá trình hủy chỉ dẫn (de-hinting) phải đảm bảo rằng các luồng đa phương tiện là hợp lệ, chẳng hạn như các luồng phải không có lỗi (error-*free*) (yêu cầu phải có biện pháp cô lập lỗi.)

Mẫu với kích cỡ bằng không được cho phép trong các rãnh chỉ dẫn tiếp nhậnvà các mẫu như thế có thể bị bỏ qua.

### Định dạng miêu tả mẫu

#### Giới thiệu

Miêu tả mẫu cho một MPEG-2 TS rãnh chỉ dẫn tiếp nhậnchứa tất cả các siêu dữ liệu tĩnh miêu tả luồng hoặc một phần của nó, đặc biệt là các bảng PSI/SI. Các MPEG-2 TS rãnh chỉ dẫn tiếp nhậnsử dụng một entry-format trong miêu tả mẫu 'rm2t' (biểu thị *MPEG-2 Transport Stream*) Entry-format cho MPEG-2 TS rãnh chỉ dẫn máy chủlà 'sm2t'.

Các tài liệu siêu dữ liệu tĩnh, chẳng hạn như các bảng PSI/SI. Sự có mặt của siêu dữ liệu tĩnh là tùy chọn. Khi chúng xuất hiện, siêu dữ liệu tĩnh là hợp lệ với các gói MPEG-2 TS nó miêu tả. Hệ quả là, nếu một siêu dữ liệu tĩnh thay đổi trong luồng, một mẫu đầu tiên sẽ cần tới một mục mẫu mới vào lúc thay đổi hoặc sau đó. Nếu siêu dữ liệu tĩnh không xuất hiện trong mục mẫu, các kiến trúc, chẳng hạn như các bảng PSI/SI, được lưu giữ trong các gói MPEG-2 TS là hợp lệ và cần phải quét luồng đó để tìm ra giá trị nào của siêu dữ liệu tĩnh là hợp lệ với một mẫu cụ thể.

#### Cú pháp

*class MPEG2TSReceptionSampleEntry extends MPEG2TSSampleEntry(`rm2t´) {*

*}*

*class MPEG2TSServerSampleEntry extends MPEG2TSSampleEntry(`sm2t´) {*

*}*

*class MPEG2TSSampleEntry(name) extends HintSampleEntry(name) {*

*uint(16) hinttrackversion = 1;*

*uint(16) highestcompatibleversion = 1;*

*uint(8)precedingbyteslen;*

*uint(8)trailingbyteslen;*

*uint(1)precomputed\_only\_flag;*

*uint(7)reserved;*

*box additionaldata[];*

*}*

#### Ngữ nghĩa

*hinttrackversion* hiện tại có giá trị là 1; trường *highestcompatibleversion* biểu thị phiên bản cũ nhất mà rãnh này có thể tương thích ngược.

*precedingbyteslen* biểu thị số lượng byte gắn trước mỗi gói MPEG-2 TS (đó có thể là một time-code từ một thiết bị thu bên ngoài).

*trailingbyteslen* biểu thị số lượng byte ở cuối của mỗi gói MPEG2-TS (có thể chứa checksum hoặc các dữ liệu khác được các thiết bị thu thêm vào).

*precomputed\_only\_flag* biểu thị các mẫu gắn cùng được tính toán trước nếu nó được thiết lập giá trị 1.

*additionaldata* là một tập các hộp. Tập này có thể chứa các hộp miêu tả một phiên bản phổ biến/chung của các bảng PSI/SI thông qua hộp 'tPAT' hoặc hộp 'tPMT' hoặc các dữ liệu khác, ví dụn như các hộp chỉ hợp lệ cho một mẫu (ở đó có chứa nhiều gói tin) và miêu tả những điều kiện ban đầu của STC hoặc các hộp định nghĩa nội dung của các dữ liệu gắn trước hoặc gắn sau. Các hộp PATBox, TSTimingBox, InitialSampleTimeBox chỉ được phép xuất hiện nhiều nhất một lần trong *additionaldata*.

Những hộp tùy chọn sau đây cho *additionaldata* được định nghĩa:

*aligned(8) class PATBox() extends Box(‘tPAT’) {*

*uint(3) reserved;*

*uint(13)PID;*

*uint(8) sectiondata[];*

*}*

*aligned(8) class PMTBox() extends Box(‘tPMT’) {*

*uint(3) reserved;*

*uint(13)PID;*

*uint(8) sectiondata[];*

*}*

*aligned(8) class ODBox () extends Box (‘tOD ’) {*

*uint(3) reserved;*

*uint(13)PID;*

*uint(8) sectiondata[];*

*}*

*aligned(8) class TSTimingBox() extends Box(‘tsti’) {*

*uint(1) timing\_derivation\_method;*

*uint(2) reserved;*

*uint(13)PID;*

*}*

*aligned(8) class InitialSampleTimeBox() extends Box(‘istm’) {*

*uint(32) initialsampletime;*

*uint(32) reserved;*

*}*

Hộp 'tPAT' chứa dữ liệu thành phần của PAT và mỗi hộp 'tPMT' chứa dữ liệu thành phần của một trong các PMT.

Trong trường hợp đối với một SPTS, khuyến nghị rằng hộp 'tPMT' có mặt trong *additionaldata*. Nếu PMT không xuất hiện trong dữ liệu mẫu, thì nó phải xuất hiện trong *additionaldata*. Nếu hộp 'tPMT'xuất hiện, nó sẽ là PMT cho chương trình được chứa trong dữ liệu mẫu (cho dù luồng có thể chứa các chương trình khác và là một MPTS).

*PID* là PID của các gói MPEG2-TS mà dữ liệu được trích xuất ra từ đó. Trong trường hợp của hộp 'tPAT' giá trị này luôn luôn là 0.

*sectiondata* mở rộng đến cuối hộp và là bảng MPEG2-TS hoàn thiện, chứa các phần kế tiếp nhau đối với một phiên bản cụ thể.

*initialsampletime* định ra giá trị ban đầu của thời gian mẫu trong trường hợp các mẫu time không bắt đầu từ 0. Không như các rãnh đa phương tiện, MPEG-2 TS rãnh chỉ dẫnthường có các mẫu time không bắt đầu từ 0, ví dụ như PCR times và thời gian nhận. Bởi vì '*stts*' chỉ lưu giữ hiệu (delta) giữa các mẫu time, trường này chỉ được yêu cầu cho việc cấu trúc lại các mẫu time ban đầu:

Trong trường hợp PCR time được sử dụng cho thời gian mẫu, thời gian mẫu được xây dựng lại có thể được sử dụng để khởi tạo STC khi mẫu được truy nhập ngẫu nhiên. Lưu ý rằng trường này có thể cần cập nhật lại sau khi sửa.

*timing\_derivation\_method* là một cờ chỉ ra phương pháp được sử dụng để đặt thời gian mẫu cho một PID cho trước. Giá trị của*timing\_derivation\_method* được xác định như sau:

*0x0* reception time: thời gian mẫu được suy ra từ thời gian nhận. Không đảm bảo chắc chắn rằngSTC được phục hồi từ việc suy ra từthời gian nhận.

*0x1* piecewise giữa các PCR: thời gian mẫu được suy ra từ một STC được xây dựng lại cho chương trình này. Giả sử tuyến tính piecewise giữa các PCR gần nhau và tất cả các gói TS trong các mẫu có constant duration trong khoảng này.

### Định dạngmẫu

Mỗi mẫu của một MPEG-2 TS rãnh chỉ dẫnbao gồm một tập hợp của :

* Các gói được tính toán trước: một hoặc nhiều gói MPEG-2 TS với các mào đầu và đoạngiới thiệu ngắn kèm theo.
* Các gói được xây dựng: các lệnh để tổng hợp một hoặc nhiều gói MPEG-2 TS cùng với các mào đầu và đoạn giới thiệu ngắn kèm theo bằng cách trỏ tới dữ liệu của một rãnh khác.

Lưu ý rằng mỗi gói tin MPEG-2 TS trong mẫu đó có thể được gắn trước bởi một tiền mào đầu (*precedingbytes*), hoặc theo sau bởi một posttrailer (*trailingbytes*), như được miêu tả chỉ tiết trong định dạng mô tả mẫu. Kích cỡ của tiền mào đầu và sau đoạn giới thiệu ngắn được quy định tương ứng bởi *precedingbyteslen* và *trailingbyteslen* trong miêu tả mẫu để cho phép các bảng mẫu nhỏ gọn ít đoạn hơn.

Các mẫu được tính toán trước và được xây dựng có thể xuất hiện trong cùng rãnh. Nếu yêu cầu padding gói tin luồng truyền tải, thì có thể thực hiện bằng *adaptation\_field* hoặc một cách tường minh bằng cách sử dụng *MPEG2TSImmediateConstructor* nếu phù hợp.

LƯU Ý 1:số lượng của các gói MPEG-2 TStrong mẫu đó có thể được suy ra trực tiếp từ bảng kích cỡ mẫu nếu mẫu chỉ chứa các gói đã được tính toán trước nếu như cờ *precomputed\_only\_flag* đã được đặt trong mục mẫu. Số lượng của các gói MPEG-2 TStrong mẫu có thể ở dạng có thể thay đổi hoặc bị giới hạn, ví dụ các mở rộng của định dạng tệp tin này có thể định nghĩa một mẫu để chứa chỉ đúng một gói tin.

LƯU Ý 2: có thể thu gọn các chuỗi phổ biến của các byte trong các gói truyền tải bằng bách đưa các byte đó trực tiếp vào một hoặc nhiều các gói, ví dụ trong *precedingbytes* hoặc *trailingbytes*section của chúng và rồi sử dụng MPEG2TSSampleConstructor ở những vị trí khác để tham chiếu đến chúng; việc này đặc biệt phù hợp để chạy các byte 0xFF.

#### Cú pháp

*// Constructor format*

*aligned(8) abstract class MPEG2TSConstructor (uint(8) type) {*

*uint(8) constructor\_type = type;*

*}*

*aligned(8) class MPEG2TSImmediateConstructor extends MPEG2TSConstructor(1) {*

*uint(8) immediatedatalen;*

*uint(8) data[immediatedatalen];*

*}*

*aligned(8) class MPEG2TSSampleConstructor extends MPEG2TSConstructor(2) {*

*uint(8) sampledatalen;*

*uint(16) trackrefindex;*

*uint(32) samplenumber;*

*uint(32) sampleoffset;*

*}*

*// Packet format*

*aligned(8) class MPEG2TSPacketRepresentation {*

*uint(8) precedingbytes[precedingbyteslen];*

*uint(8) sync\_byte;*

*if (sync\_byte == 0x47) {*

*uint(8) packet[187];*

*} else if (sync\_byte == 0x00 || sync\_byte == 0x01) {*

*uint(8)headerdatalen;*

*uint(4)reserved;*

*uint(4)num\_constructors;*

*bit(1) transport\_error\_indicator;*

*bit(1) payload\_unit\_start\_indicator;*

*bit(1) transport\_priority;*

*bit(13) PID;*

*bit(2) transport\_scrambling\_control;*

*bit(2) adaptation\_field\_control;*

*bit(4) continuity\_counter;*

*if (sync\_byte == 0x00 && (adaptation\_field\_control == ´10´ || adaptation\_field\_control == ´11´)) {*

*uint(8) adaptation\_field[headerdatalen-3];*

*}*

*MPEG2TSConstructor constructors[num\_constructors];*

*} else if (sync\_byte == 0xFF) {*

*// implicit null packet that has been removed*

*}*

*uint(8) trailingbytes[trailingbyteslen];*

*}*

*// Sample format*

*aligned(8) class MPEG2TSSample {*

*MPEG2TSPacketRepresentation sample[];*

*}*

#### Ngữ nghĩa

*precedingbytes* chứa bất cứ dữ liệu phụ trợ nào kèm trước các gói, thường được gắn vào bởi thiết bị thu. Ví dụ, nó có thể chứa một nhãn thời gian.

*sync\_byte*: nếu giá trị này bằng 0x47 thì việc biểu diễn gói tinsẽ chứa một gói luồng truyền tải (mộtmẫu rãnh chỉ dẫn tiếp nhận được tính toán trước), cùng với các byte còn lại sau trong trường gói. Các giá trị 0x00 và 0x01 được sử dụng cho (các) bản trình diễn gói có cấu trúc. Nếu MPEG2TSSampleConstructor được sử dụng để cấu tạo nên (các) bản trình diễn gói, nó sẽ trỏ tới một rãnh được đánh số bởi *trackrefindex* trong hộp tham chiếu rãnh với kiểu tham chiếu 'hint'. Nếu giá trị này bằng 0xFF, điều đó có nghĩa một gói trống đã được kiểu bỏ ở vị trí này. Tất cả các giá trị khác hiện tại đều được dự phòng.

*trackrefindex* đánh số Hộp Tham chiếu Rãnh vớikiểu tham chiếu 'hint' cho biết rãnh đa phương tiệnmà mẫu hiện thời đang gắn với.Các trường*samplenumber*và*sampleoffset* trong *MPEG2TSSampleConstructor* trỏ tới rãnh đa phương tiệnnày. *trackrefindex* bắt đầu từ giá trị 1. Giá trị 0 được để dành cho việc sử dụng trong tương lai.

*packet*: gói MPEG-2 TS, ngoại trừ byte đồng bộ (0x47).

Mảng *MPEG2TSConstructor* là tập hợp một hoặcnhiều constructor entry, nhằm cho phép nhiều unit truy nhập trong một gói tin luồng truyền tải. Một *MPEG2TSImmediateConstructor* có thể chứa cả mào đầu PES. *MPEG2TSSampleConstructor* tham chiếu dữ liệu trong các rãnh đa phương tiệnkèm theo. Tổng của các trường *headerdatalen* và *datalen* của tất cả các constructor của một *MPEG2TSPacket* phải bằng với độ dài của gói tin luồng truyền tải đang được tạo dựng, trừ đi 1 byte, đó là 187.

*trailingbytes* chứa bất cứ thông tin phụ trợ nào kèm theo sau gói tin. Ví dụ, nó có thể bao gồm một checksum.

*samplenumber* biểu thị sample trong referred rãnh chứa trong gói và *sampleoffset* biểu thị vị trí byte khởi đầu của mẫu đa phương tiệntham chiếu được chứa trong gói có chứa các byte *sampledatalen*. *sampleoffset* bắt đầu từ giá trị 0.

*immediatedatalen* biểu thị số lượng các byte trong trường *data* được chứa trong mẫu hơn là dữ liệu được chứa trong mẫu được tham chiếu tới một rãnh đa phương tiện.

*headerdatalen* biểu thị độ dài của mào đầu gói tin TS (không tính byte đồng bộ) bằng đơn vị byte. Trường này có giá trị bằng 3 nếu như không *adaptation\_field* không xuất hiện hoặc giá trị *(adaptation\_field\_length+3)*, với *adaptation\_field\_length* là octet đầu tiên của *adaptation\_field* như được định nghĩa trong ISO/IEC 13818-1.

Định dạng của cả *precedingbytes* và *trailingbytes* đều không được định nghĩa bởi bản đặc tả này.

Các trường còn lại của tổng hợpmẫu*(transport\_error\_indicator, payload\_unit\_start\_indicator, transport\_priority, PID, transport\_scrambling\_control, adaptation\_field\_control, continuity\_counter, adaptation\_field)* chứa một bản sao của mào đầu gói tin của gói TS, được định nghĩa trong ISO/IEC 13818-1.

### Rãnh chỉ dẫnluồng truyền tải MPEG-2 được bảo vệ

#### Giới thiệu

Tiểu mục này định ra một cơ chế để đánh dấu các luồng đa phương tiệnđược bảo vệ. Việc này được thực hiện bằng cách thay đổi mã bốn ký tự của SampleEntry, và gắn thêm những hộp có chứa chi tiết của cơ chế bảo vệ và mã bốn ký tự ban đầu. Tuy vậy, trong trường hợp này rãnh không được bảo vệ; nó là một rãnh chỉ dẫn'in the clear' có chứa dữ liệu được bảo vệ. Tiểu mục này miêu tả cách thức đánh dấu các rãnh chỉ dẫnmang dữ liệu được bảo vệ, sử dụng một cơ chế tương tự, và tận dụng cùng các hộp đó.

#### Cú pháp

*class ProtectedMPEG2TransportStreamSampleEntryextends MPEG2TransportStreamSampleEntry(‘pm2t’) {*

*ProtectionSchemeInfoBoxSchemeInformation;*

*}*

#### Ngữ nghĩa

Hộp*SchemeInformation ("sinf “)*(được định nghĩa trong 8.12) chứa các chi tiết của cơ chế bảo vệ được áp dụng. Nó cần phải bao gồm *OriginalFormatBox* trong đó chứa mục mẫu type ban đầu của MPEG-2 Hộp StreamSampleEntry.

## Rãnh chỉ dẫn tiếp nhận RTP, RTCP, SRTP và SRTCP

### Rãnh chỉ dẫn tiếp nhận RTP

#### Giới thiệu

Tiểu mục này quy định định dạng của rãnh chỉ dẫn tiếp nhậncho giao thức vận tải thời gian thực (RTP), như định nghĩa trong IETF RFC 3550.

RTP được sử dụng cho đa phương tiện transport thời gian thực qua giao thức Internet (Internet Protocol). Mỗi luồng RTP mang một đa phương tiện type, và mỗi rãnh chỉ dẫnRTP reception mang một luồng RTP.Bởi thế, việc thu một chương trình gồm cả hình và tiếng cần ít nhất hai rãnh chỉ dẫn tiếp nhận RTP.

Thiết kế của định dạng rãnh chỉ dẫn tiếp nhận RTPtheo sát thiết kế của định dạng rãnh chỉ dẫn máy chủ RTPhết mức có thể. Thiết kế đó cần phải đảm bảo rằng việc truyền gói RTP hoạt động hết sức tương tự bất chấp nó dựa trên các rãnh chỉ dẫn tiếp nhận RTPhay rãnh chỉ dẫn máy chủ RTP. Thêm vào đó, số lượng của các kiến trúc dữ liệu mới trong định dạng tệp tin, bởi vì đó, phải được giữ nhỏ nhất có thể.

Định dạng của các rãnh chỉ dẫn tiếp nhận RTPcho phép việc lưu trữ dữ liệu gói tin trong các mẫu chỉ dẫn, hoặc biến đổi dữ liệu gói tinRTP sang các mẫu đa phương tiện và thêm chúng vào các mẫu chỉ dẫn thông qua tham chiếu, hoặc kết hợp cả hai cách vừa nêu.Như đã lưu ý ở các phần trước, việc chuyển đổi các luồng nhận được sang các rãnh đa phương tiệncho phép các thiết bị đọc hiện tại tương thích với các phiên bản trước của các định dạng tệp tinđa phương tiệncơ sở ISO để xử lý các tệp tinthu được nếu các định dạng đa phương tiện cũng được hỗ trợ. Việc lưu giữ các mào đầu RTP gốc mang lại các thông tin quý giá cho việc cô lập lỗi và tái tạo luồng RTP ban đầu.Lưu ý rằng việc chuyển đổi cácdữ liệu gói tinsang các mẫu đa phương tiện có thể diễn ra “off-line” sau khi việc ghi các luồng trong các rãnh chỉ dẫn tiếp nhậnRTP được tính toán trước đã hoàn thành.

#### Định dạngmiêu tả mẫu

Entry-format trong miêu tả mẫu cho rãnh chỉ dẫn tiếp nhận RTPlà 'rrtp'. Cú pháp của mục mẫu cũng giống như các rãnh chỉ dẫn máy chủ RTPcó định dạng đầu vào mục 'rtp'.

*class ReceivedRtpHintSampleEntry() extends SampleEntry (‘rrtp‘) {*

*uint(16)hinttrackversion = 1;*

*uint(16)highestcompatibleversion = 1;*

*uint(32) maxpacketsize;*

*boxadditionaldata[];*

*}*

Định danh entry-format trong miêu tả mẫu của rãnh chỉ dẫn tiếp nhận RTPkhác với entry-format trong miêu tả mẫu của rãnh chỉ dẫn máy chủ RTP, nhằm tránh việc sử dụng một rãnh chỉ dẫn tiếp nhận RTPchứa lỗi như một rãnh chỉ dẫn máy chủhợp lệ.

Tập các hộp *additionaldata* có thể bao gồm đầu vào mục thang thời gian ('tims') và các hộp độ dịch chuyển thời gian ('tsro'). Thêm vào đó, *additionaldata* có thể chứa một Hộp Đồng bộ Nhãn thời gian

Hộp Đầu vào mục Thang thời gian (‘*tims’*) sẽ xuất hiện và giá trị của thang thời gianđược thiết lập để phù hợp với tần số đồng hồ của nhãn thời gian RTP của luồngđã thu được trong rãnh chỉ dẫn tiếp nhận.

Hộp Độ dịch chuyển Thời gian(‘*tsro’*) cũng có thể xuất hiện.Nếu như hộpđộ dịch chuyển thời gian không xuất hiện, giá trị của trường độ dịch chuyển được suy ra bằng 0. Giá trị của trường độ dịch chuyển được sử dụng để suy ra nhãn thời gian RTP, như đã nêu trong 9.4.1.4.

Về cơ bản nhãn thời gian RTP không bắt đầu từ không, đặc biệt là nếu bên thu RTP điều chỉnh luồng. Vì thế hộp độ dịch chuyển thời gian cần phải xuất hiện trong các rãnh chỉ dẫn tiếp nhận RTPvà giá trị của độ dịch chuyển trong hộp độ dịch chuyển thời gian cần được đặt bằng với nhãn thời gian RTP đầu tiên của luồng RTP trong thứ tự nhận.

Các hộp 0 hoặc1 *timestampsynchrony* có thể xuất hiện trong *additionaldata* trong mục mẫu cho rãnh chỉ dẫn tiếp nhận RTP. Nếu hộp *timestampsynchrony* không xuất hiện, giá trị của *timestamp\_sync* được suy ra là bằng 0.

*class timestampsynchrony() extends Box(‘tssy’) {*

*unsigned int(6) reserved;*

*unsigned int(2) timestamp\_sync;*

*}*

*timestamp\_sync*bằng 0 biểu thị rằng các nhãn thời gian RTP của rãnh chỉ dẫn tiếp nhận RTPhiện tại được suy ra từ công thức trong 9.4.1.4 có thể hoặc không được đồng bộ với các nhãn thời gian RTP hoặc các rãnh chỉ dẫn tiếp nhận RTPkhác.

*timestamp\_sync* bằng 1 biểu thị rằng các nhãn thời gian RTP của rãnh chỉ dẫn tiếp nhận RTPhiện tại được suy ra từ công thức trong 9.4.1.4 phản ánh đúng nhãn thời gian RTP nhận được (mà không cần sửa đồng bộ với bất cứ rãnh chỉ dẫn tiếp nhận RTPnào khác).*timestamp\_sync* bằng 2 cho thấy rằng các nhãn thời gian RTP của rãnh chỉ dẫn tiếp nhận RTPhiện tại được suy ra từ công thức trong 9.4.1.4 được đồng bộ với các nhãn thời gian RTP của các rãnh chỉ dẫn tiếp nhận RTPkhác.

Khi*timestamp\_sync* có giá trị 0 hoặc 1, thiết bị đọc cần điều chỉnh đồng bộnội luồng (inter-stream) thông qua các báo cáo bên gửi RTCP đã được. Khi*timestamp\_sync*bằng 2, đa phương tiện chứa trong rãnh chỉ dẫn tiếp nhận RTPcó thể được phát đồng bộ theo các nhãn thời gian RTP được tái tạo lại mà không cần điều chỉnh đồng bộ thông qua báo cáo bên gửi RTCP. Nếu như rãnh chỉ dẫn tiếp nhận RTPđược sử dụng để gửi tiếp luồng RTP đã được thu, khuyến nghị đặt giá trị 0 hoặc 1 cho *timestamp\_sync*, bởi vì có thể tái sử dụng các bản báo cáo bên gửi RTCP đã lưu.

*timestamp\_sync* bằng 3 được để dành.

Giá trị *timestamp\_sync* có thể giống nhau cho tất cả các rãnh chỉ dẫn tiếp nhận RTPcó mặt trong một tệp tin.

Khi RTCP cũng được lưu giữ, việc sử dụng một RTCP rãnh chỉ dẫn, mối quan hệ nhãn thời gian giữa các các rãnh chỉ dẫn RTP và RTCP chỉ có thể được duy trì nếu như các nhãn thời gian RTP được neo bằng cách sử dụng một tập độ dịch chuyển thời gian ('tsro') trong rãnh RTP, và bởi thế độ dịch chuyển thời gian là bắt buộc nếu như RTCP được lưu trong rãnh chỉ dẫn RTCP.

*ReceivedSsrcBox*không hoặc một. được xác định bởi mã bốn ký tự 'rssr' phải xuất hiện trong *additionaldata* của một đầu vào mục miêu tả mẫu của một rãnh chỉ dẫn tiếp nhận RTP:

*class ReceivedSsrcBox extends Box(‘rssr’) {*

*unsigned int(32)SSRC*

*}*

Giá trị *SSRC* phải bằng với giá trị SSRC trong mào đầu của tất cả các gói SRTP đã được lưu giữ được miêu tả bởi bản miêu tả mẫu.

#### Định dạng mẫu

Định dạng mẫu của rãnh chỉ dẫn tiếp nhận RTPgiống như cú pháp của định dạng mẫu của rãnh chỉ dẫn máy chủ RTP. Mỗi một mẫu trong rãnh chỉ dẫn tiếp nhậnbiểu thị một hoặc nhiều gói RTP đã nhận. Nếu các khung hình đa phương tiện không phân tán và đan xen trong một luồng RTP, khuyến nghị rằng mỗi mẫu biểu thị tất cả các gói RTP nhận được có cùng nhãn thời gian RTP, tức là các gói liên tiếp nhau trong thứ tự RTP sequence number với nhãn thời gian RTP chung.

Mỗi mẫu chỉ dẫn tiếp nhận RTP chứa hai vùng: các lệnh để hình thành nên gói, và bất cứ dữ liệu cần thiết nào cho việc cấu thành nên gói, chẳng hạn như bản sao của dữ liệu gói. Lưu ý rằng kích thước của mẫu được biết từ bảng kích thước mẫu.

Bởi vì thời gian nhậncủa các gói tin có thể thay đổi, sự thay đổi này có thể được báo cho mỗi gói như được định ra sau đây.

Mẫu với kích cỡ bằng không được cho phép trong các rãnh chỉ dẫn tiếp nhậnvà các mẫu như thế có thể bị bỏ qua.

#### Định dạngđầu vào mục gói tin

Mỗi gói tin trong bảng đầu vào mục gói tin đều có cùng một cấu trúc cho các rãnh chỉ dẫn máy chủ (transmission), trong 9.1.3.1.Với *i* là số mẫu của một mẫu, tổng số thời gian mẫu DT(i) được xác định trong 8.6.1.2. và*relative\_time* biểu thị thời gian nhận của gói tin.Nguồn đồng hồ củathời gian nhận chưa được xác định và có thể, lấy ví dụ, đồng hồ của bên nhận.Nếu dải thời gian nhận của một rãnh chỉ dẫn tiếp nhậnchồng lấn hoàn toàn hoặc một phần với dải thời gian nhận của một rãnh chỉ dẫn tiếp nhậnkhác, thì nguồn đồng hồ củacác rãnh chỉ dẫnnày phải giống nhau.

Khuyến nghị rằng các bộ thu có thể sử dụng một hằng số cho *sample\_delta* trong hộp thời gian giải mã mẫu ('*stts*') với giá trị ở mức hợp lý và phù hợp với việc lập lịch gói tin vàbiến động trễ end-to-end bằng cách đặt *relative\_time* một cách tương ứng trong các mẫu chỉ dẫn tiếp nhận đã được lưu giữ. Việc thiết đặt các giá trị *sample\_delta* và *relative\_time* có thể làm giảm bớt thời gian giải mã tới hộp mẫu. Trong trường hợp đó, *timestamp\_sync* được đặt bằng 1, các mẫu duration hầu như không thay đổi, và độ dịch chuyển thời gian ('tsro' được lưu giữ trong mục mẫu.

Các giá trị của*RTP\_version, P\_bit, X\_bit, CSRC\_count, M\_bit, payload\_type*, và *RTPequenceseed*phải được đặt bằng V, P, X, CC, M, PT và các trường số của chuỗi của gói RTP thu được trong mẫu đó.

Các trường *bframe\_flag* và *repeat\_flag* được để dành trong các rãnh chỉ dẫn tiếp nhậnvà phải bằng không.

Ngữ nghĩa của *extra\_flag*và *extra\_information\_length* giống hoàn toàn với những giá trị được quy định cho các rãnh chỉ dẫn máy chủ RTP.

Các hộp TLV sau đây được định nghĩa: *rtphdrextTLV, rtpoffsetTLV, receivedCSRC*.

Nếu giá trị *X\_bit* đã được thiết lập thì phải có một hộp *rtphdrextTLV* để lưu giữ mở rộng mào đầu RTP đã nhận được.

*aligned(8) class rtphdrextTLV extends Box(‘rtpx’) {*

*unsigned int(8) data[];*

*}*

*data* là mở rộng mào dầu RTP thô tùy thuộc theo các ứng dụng.

Cú pháp của hộp*rtpoffsetTLV* được quy định trong 9.1.3.1.

*offset* biểu thị một giá trị độ dịch chuyển 32-bit nguyên có dấu với nhãn thời gian RTP của gói RTP đã nhận được. Gọi *i* là số mẫu của một mẫu, DT(i) bằng với DT được quy định trong 8.6.1.2 cho mẫu thứ i, *tsro.offset* là giá trị độ dịch chuyển trong hộp 'tsro' của đầu vào mục mẫu chỉ dẫn tiếp nhận được tham khảo đến, và % là phép toán modulo. Giá trị của *offset* phải đảm bảo thỏa biểu thức sau đây là đúng:

công thức (1) tính toán nhãn thời gian RTP

LƯU Ý 1: Khi mỗi mẫu chỉ dẫn tiếp nhậnbiểu thị tất cả các gói RTP nhận được có cùng nhãn thời gian RTP, giá trị của *sample\_delta* trong thời gian giải mã tới hộp mẫu có thể được đặt để phù hợp với nhãn thời gian RTP. Nói cách khác, DT(i), như được định nghĩa ở trên, có thể được dặt bằng với (*RTP timestamp - tsro.offset - offset*) (giả sử rằng giá trị kết quả sẽ lớn hơn hoặc bằng 0). Khuyến nghị thực hiện cách này.

LƯU Ý 2: Các nhãn thời gian RTP không nhất thiết phải tăng lên nhu một hàm của số chuỗi RTP trong tất cả luồng RTP, chẳng hạn như, thứ tự truyền phát và thứ tự chơi lại các gói tin có thể không hoàn toàn giống nhau. Lấy ví dụ, rất nhiều hình ảnhlược đồ mã hóa cho phép dự đoán hai chiều từ các hình ảnh đến trước và đến sau trong thứ tự chơi lại. Khi các mẫu xuất hiện trong các rãnh theo đúng thứ tự giải mã của chúng, chẳng hạn như theo thứ tự tiếp nhận trong trường hợp của rãnh chỉ dẫn tiếp nhận RTP, độ dịch chuyển trong hộp *rtphdrextTLV* có thể được sử dụng để warp nhãn thời gian RTP khỏi thời gian mẫu DT(i).

Nhằm mục đích điều chỉnh trong các Hộp Danh sách Biên tập, thời gian tổng hợp của một gói RTP đã nhận được suy ra từ tổng của các mẫu time DT(i) và độ dịch chuyển như được định nghĩa ở trên.

Nếu giá trị của *CSRC\_count*không bằng không, một hộp receivedCSRC có thể xuất hiện để lưu giữ các trường mào đầu CSRC nhận được cho mỗi gói RTP. Hộp receivedCSRC được xác định với mã bốn ký tự *‘rcsr*'.

*aligned(8) class receivedCSRC extends Box(‘rcsr') {*

*unsigned int(32)CSRC[]; //to end of the box*

*}*

Số lượng của các đầu vào mục trong *CSRC[]* bằng với giá trị *CC* value của các gói SRTP nhận được. Đầu vào mục thứ n của *CSRC[]* phải bằng với giá trị CSRC thứ n của mào đầu gói tin RTP.

#### Thông tin SDP

Cả movie và thông tin rãnh SDP có thể xuất hiện, như đã được quy định trong 9.1.4.

### Rãnh chỉ dẫn tiếp nhận RTCP

#### Giới thiệu

Tiểu mục này quy định định dạng của rãnh chỉ dẫn tiếp nhậncho giao thức điều khiển thời gian thực (RTCP), như định nghĩa trong IETF RFC 3550.

RTCP được sử dụng cho vận tải thời gian thực của thông tin điều khiển cho phiên RTP qua giao thức Internet. Trong quá trình streaming, mỗi luồng RTP thông thường có một luồng RTCP kèm theo chứa các thông tin điều khiển cho luồng RTP đó. Một rãnh chỉ dẫn tiếp nhận RTCPmang theo một luồng RTCP và được gắn với rãnh chỉ dẫn tiếp nhận RTPtương ứng thông qua một tham chiếu rãnh.

Định dạng của các rãnh chỉ dẫn tiếp nhận RTCPcho phép việc lưu giữ của các báo cáo của bên gửi RTCP trong các mẫu chỉ dẫn.

Các báo cáo của bên gửi RTCP là đối tượng quan tâm của việc ghi luồng, bởi vì chúng phản ánh trạng thái hiện tại của các máy chủ, chẳng hạn như mối quan hệ của thời gian đa phương tiện (nhãn thời gian RTP của các gói âm thanh/hình ảnh) tới máy chủ thời gian (thời gian tuyệt đối trong định dạng NTP). Thông tin về mối quan hệ này cũng cần thiết cho việc chơi lại của các rãnh chỉ dẫn tiếp nhận RTPđã được ghi lại để có thể dò tìm và sửa chữa trôi đồng hồ và jitter.

Hộp Đồng bộ Nhãn thời gian được định nghĩa trong 9.4.1.2 cho phép việc sửa lỗi trôi đồng hồ và jitter trước khi play một tệp tin, và bởi thế việc thu các luồng RTCP là tùy chọn.

Không có rãnh chỉ dẫn máy chủtương ứng cho rãnh chỉ dẫn tiếp nhận RTCP, bởi vì các bản tin RTCP được khởi tạo ngay trong quá trình truyền dẫn (on-the-fly).

#### Tổng quan

Sẽ không có hoặc có một rãnh chỉ dẫn tiếp nhận RTCPcho mỗi rãnh chỉ dẫn tiếp nhận RTP. Một rãnh chỉ dẫn tiếp nhận RTCPchứa một Hộp Tham chiếu Rãnh bao gồm một tham chiếu kiểu *'cdsc'* tới một rãnh chỉ dẫn tiếp nhận RTPliên quan.

Với i là số mẫu của một mẫu, thời gian mẫu DT(i) được định ra trong 8.6.1.2 biểu thị thời gian nhận của gói tin đó. Nguồn đồng hồ cho thời gian nhận phải bằng với nguồn dành cho rãnh chỉ dẫn tiếp nhận RTPliên quan đó. Giá trị của thang thời gian (timescale) trong Hộp Mào đầu Đa phương tiện của một rãnh chỉ dẫn tiếp nhận RTCPphải bằng với giá trị thang thời gian trong Hộp Mào đầu Đa phương tiện củarãnh chỉ dẫn tiếp nhận RTPtương ứng.

#### Định dạngmiêu tả mẫu

entry-format trong miêu tả mẫu dành cho các rãnh chỉ dẫn tiếp nhận RTCPlà 'rtcp'. Ngoại trừ điều đó, cấu trúc hoàn toàn giống với định dạng mẫu đầu vào mục cho RTP.Không có hộp nào được định nghĩa cho trường *additionaldata*.

#### Định dạng mẫu

##### Giới thiệu

Mỗi mẫu trong rãnh chỉ dẫn tiếp nhậnbiểu thị một hoặc nhiều gói tin RTCP nhận được. Mỗi mẫu bao gồm hai khu vực: các gói tin raw RTCP và bất cứ thông tin bổ sung cần thiết nào. Lưu ý rằng kích thước của mẫu đã được biết từ bảng kích cỡ mẫu, và rằng kích cỡ của một gói RTCp được biểu thị trong bản thân gói tin đó (như được quy định trong RFC 3550), ít hơn một so với số lượng các word 32 bit trong gói tin đó.

##### Cú pháp

*aligned(8) class receivedRTCPpacket {*

*unsigned int(8)data[];*

*}*

*aligned(8) class receivedRTCPsample {*

*unsigned int(16)packetcount;*

*unsigned int(16)reserved;*

*receivedRTCPpacketpackets[packetcount];*

*}*

##### Ngữ nghĩa

*data* chứa một gói tin RTCP thô, bao gồm mào đầu báo cáo RTCP, khối thông tin bên gửi 20 byte và số lượng khối báo cáo không giới hạn.Lưu ý rằng kích cỡ của mỗi gói RTCP có thể biết bằng cách phân tách các trường độ dài 16 bit của mào đầu RTCP.

*packetcount* biểu thị số lượng của các gói tin RTCP nhận được được lưu giữ trong mẫu.

*packets* chứa các gói tin RTCP đã nhận được.

### Rãnh chỉ dẫn tiếp nhận SRTP

#### Giới thiệu

Tiểu mục này quy định các định dạng của rãnh chỉ dẫn tiếp nhậncho giao thức vận tải thời gian thực an toàn (SRTP), như định nghĩa trong IETF RFC 3711.

SRTP là phần mở rộng an toàn cho đa phương tiện transport thời gian thực (RTP) qua giao thức Internet (Internet Protocol). Mỗi luồng SRTP mang một dạng đa phương tiện, và mỗi rãnh chỉ dẫn tiếp nhận RTPmang một luồng RTP. Bởi thế, thu một chương trình gồm cả âm thanh hình ảnh cho ra ít nhất hai rãnh chỉ dẫn tiếp nhận RTP.

Thiết kế của định dạng Rãnh chỉ dẫn SRTPtuân theo thiết kế của các rãnh chỉ dẫn tiếp nhận RTPvà sử dụng lại hầu hết khung làm việc cung cấp bởi các rãnh chỉ dẫn tiếp nhận RTP. Điểm khác biệt chủ yếu giữa RTP và rãnh chỉ dẫn tiếp nhận RTPlà nội dung đa phương tiện thực sự được lưu trữ dưới dạng mã hóa trong rãnh chỉ dẫn tiếp nhận RTP, trong khi nó là dạng không mã hóa cho các rãnh chỉ dẫn tiếp nhận RTP. Các rãnh chỉ dẫn tiếp nhận RTPcung cấp các hộp phụ trợ đểlưu giữ các thông tin cần thiết cho việc giải mã các nội dung đã được mã hóa trong quá trình phát lại. Thêm vào đó, tất cả các trường mào đầu của mào đầu gói tin SRTP phải được lưu trữ cùng với dữ liệu, bởi vì thông tin này cần thiết cho việc kiểm tra tính toàn vẹn của dữ liệu nhận được. Các rãnh chỉ dẫn tiếp nhận RTPthông thường được sử dụng cùng với các rãnh chỉ dẫn tiếp nhận RTCP.

Các rãnh chỉ dẫn tiếp nhận RTCPcó thể, lấy ví dụ, được sử dụng để lưu trữ các nội dung mobile TV được bảo vệ.

#### Định dạngmiêu tả mẫu

##### Đầu vào mục miêu tả mẫu

Định dạng mẫu của các rãnh chỉ dẫn tiếp nhận RTPgiống như định dạng mẫu của các rãnh chỉ dẫn tiếp nhận RTPngoại trừ rằng các tên của mục mẫu được đổi từ 'rrtp' sang 'rsrp' và nó có thể chứa các hộp bổ sung:

*class ReceivedSrtpHintSampleEntry() extends SampleEntry (‘rsrp‘) {*

*uint(16)hinttrackversion = 1;*

*uint(16)highestcompatibleversion = 1;*

*uint(32)maxpacketsize;*

*box additionaldata[];*

*}*

Các trường và các hộp giống như của *ReceivedRtpHintSampleEntry*(‘*rrtp’*). *addtionaldata[]*của mỗi đầu vào mục miêu tả mẫu của một rãnh chỉ dẫn tiếp nhận SRTPphải chứa duy nhất một hộp *ReceivedSsrc* (*‘rssr’*).

Thêm vào đó, *additionaldata[]* có thể chứa Hộp ID Ngữ cảnh Mật mã Nhận và Hộp Bộ đếm Luân hồi được định nghĩa dưới đây. Hơn nữa, một hộp xử lý SRTP cũng có thể được thêm vào như là một trong những hộp *additionaldata*. Bởi vì nội dung được lưu trữ ở dạng mã hóa, tính toàn vẹn và các trường thuật toán mã hóa trong Hộp Xử lý SRTP định ra thuật toán sẽ được áp dụng đối với luồng nhận được. Một đầu vào mục với bốn dấu trắng ($20$20$20$20) có thể được sử dụng để biểu thị rằng thuật toán đó được định nghĩa bằng các phương thức bên ngoài phạm vi của tài liệu này.

##### Hộp IDNgữ cảnh Mật mã Nhận (Received Cryptographic Context ID Box)

Có thể không xuất hiệnhoặc xuất hiện một *ReceivedCryptoContextIdBox*, được xác định với mã bốn ký tự 'ccid', trong *additionaldata* của một đầu vào mục miêu tả mẫu của một rãnh chỉ dẫn tiếp nhận RTP.Thông tin để phục hồi ngữ cảnh mã hóa của luồng RTP nhận được có thể được lưu trữ ở đây.

*aligned(8) class ReceivedCryptoContextIdBox extends Box (‘ccid’) {*

*unsigned int(16)destPort;*

*unsigned int(8) ip\_version;*

*switch (ip\_version) {*

*case 4: // IPv4*

*unsigned int(32)destIP;*

*break;*

*case 6: // IPv6*

*unsigned int(64)destIP;*

*break;*

*}}*

Các thông số *destPort*và*destIP* chứa số cổngvà địa chỉ IP(như xuất hiện trong các gói IPv4 và IPv6 nhận được) của phiên SRTP mà thông qua đó các gói tin SRTP đã được ghi đã được nhận. *ip\_version*có giá trị 4 hoặc 6 tương ứng với IPv4 hay IPv6.

##### Hộp Bộ đếm Luân hồi(Box Rollover Counter)

Có thể tồn tại hoặc không tồn tài *RolloverCounterBox*, được xác định bởi mã bốn ký tự 'sroc', trong *additionaldata* của một đầu vào mục miêu tả mẫu của một rãnh chỉ dẫn tiếp nhận RTP. Thông thường, giá trị bộ đếm luân hồi thay đổi sau 65536 gói SRTP.

*aligned(8) class RolloverCounterBox extends Box (‘sroc’) {*

*unsigned int(32)rollover\_counter;*

*}*

*rollover\_counter*có kiểu số nguyên khác không cung cấp một giá trị cho trường ROC của tất cả các gói SRTP liên quan nhận được.

LƯU Ý:bộ đếm luân hồi (ROC) là một thành phần của ngữ cảnh mật mã của một luồng RTP và phụ thuộc vào vị trí tuyệt đối của một gói tin trong một luồng RTP. Thông tin về giá trị ROC là cần thiết để giải mã gói tin SRTP nhận được. Việc sử dụng Hộp Bộ đếm Luân hồi là tùy chọn vìRFC 4771 định nghĩa như là một cơ chế tùy chọn để báo hiệu giá trị ROC một cách rõ ràng trong quá trình xác thực nhãn của một gói SRTP.

#### Định dạng mẫu và Packet Entry

Cả hai, định dạng mẫu và định dạng đầu vào mục gói tin cho các rãnh chỉ dẫn tiếp nhận RTPlà đồng nhất với của rãnh chỉ dẫn tiếp nhận RTP, được định nghĩa trong 9.4.1.3 và 9.4.1.4. Dữ liệu trong gói tin được lưu trữ như các gói tin SRTP nhận được, chẳng hạn như, tất cả các thông tin nhận được trong gói tin SRTP ngoại trừ mào đầu hoặc, nói cách khác, dữ liệu được mã hóa cùng với định danh khóa (MKI) và nhãn xác thực.

Nếu giá trị của *CSRC\_count* không bằng không đối với một gói SRTP đã nhận được, thì *extra\_data\_tlv* tương ứng với*receivedSRTPpacket* này phải chứa cơ bản xác một hộp receivedCSRC (*‘rcsr'*).

### Rãnh chỉ dẫn tiếp nhận SRTCP

#### Giới thiệu

Tiểu mục này quy định các định dạng của rãnh chỉ dẫn tiếp nhậncho giao thức điều khiển vận tải thời gian thực an toàn (SRTCP), như định nghĩa trong IETF RFC 3711.

SRTCP được sử dụng cho vận tải thời gian thực của thông tin điều khiển cho phiên SRTP qua Internet Protocol. SRTCP có vai trò đối với SRTP cũng như vai trò của RTCP với RTP, so sánh với 9.4.2. Trong quá trình streaming, mỗi luồng RTP thông thường có một luồng RTCP kèm theo mang thông tin điều khiển cho luồng RTP. Một SRTCPrãnh chỉ dẫn tiếp nhậnmang một luồng RTCP và gắn với SRTCPrãnh chỉ dẫn tiếp nhậntương ứng thông qua một rãnhtham chiếu.

Định dạng của các rãnh chỉ dẫn tiếp nhận RTCPcho phép việc lưu giữ của các gói tin SRTCP trong các mẫu chỉ dẫn, chẳng hạn như của cácBáo cáo của bên gửi SRTCP.

Các báo cáo của bên gửi SRTCP là đối tượng quan tâm của việc ghi luồng, bởi vì chúng phản ánh trạng thái hiện tại của các máy chủ, chẳng hạn như mối quan hệ của thời gian đa phương tiện (SRTP nhãn thời gian của các gói âm thanh/ hình ảnh) tới máy chủ thời gian (thời gian tuyệt đối trong định dạng NTP). Thông tin về mối quan hệ này cũng cần thiết cho việc chơi lại của các rãnh chỉ dẫn tiếp nhận RTPđã được ghi lại để có thể dò tìm và sửa chữa trôi đồng hồ và jitter.

Hộp Đồng bộ Nhãn thời gian được định nghĩa trong 9.4.1.2 cho phép việc sửa lỗi trôi đồng hồ và jitter trước khi play một tệp tin, và bởi thế việc thu các luồng RTCP là tùy chọn.

Không có rãnh chỉ dẫn máy chủtương ứng cho rãnh chỉ dẫn tiếp nhận RTCP, bởi vì các bản tin RTCP được khởi tạo ngay trong quá trình truyền dẫn (on-the-fly).

#### Tổng quát

Sẽ có một hoặc không có rãnh chỉ dẫn tiếp nhận RTCPcho mỗi rãnh chỉ dẫn tiếp nhận RTP. Mỗi rãnh chỉ dẫn tiếp nhận RTCPchứa một Hộp Tham chiếu Rãnh bao gồm một tham chiếukiểu '*cdsc*' tới rãnh chỉ dẫn tiếp nhận RTPtương ứng.

Với i là số mẫu của một mẫu, thời gian mẫu DT(i) được định ra trong 8.6.1.2 biểu thị thời gian nhận của gói tin đó. Nguồn đồng hồ cho thời gian nhận phải bằng với nguồn dành chorãnh chỉ dẫn tiếp nhận RTPliên quan đó. Giá trị của thang thời gian trong Hộp Mào đầu Đa phương tiện của một rãnh chỉ dẫn tiếp nhận RTCPphải bằng với giá trị thang thời gian trong Hộp Mào đầu Đa phương tiện củarãnh chỉ dẫn tiếp nhận RTPtương ứng.

#### Định dạngmiêu tả mẫu

entry-format trong miêu tả mẫu cho rãnh chỉ dẫn tiếp nhận RTPlà 'stcp'.Ngoại trừ điều đó, cấu trúc hoàn toàn giống với định dạng mẫu đầu vào mục cho RTCP. Phương pháp mã hóa và xác thực của các Rãnh chỉ dẫn SRTPđược định nghĩa bởi các đầu vào mục tương ứng trong hộp SRTP Process của Rãnh chỉ dẫn SRTPtương ứng.

LƯU Ý: không nhất thiết phải có một sự tương ứng giữa các hộp ROC định nghĩa cho SRTP với SRTCP, bởi gói tin SRTCP chứa vector khởi tạo báo hiệu.

#### Định dạng mẫu

Định dạng mẫu là định dạng mẫu cho rãnh chỉ dẫn tiếp nhận RTCPnhư được định nghĩa trong 9.4.2.4.

### Rãnh chỉ dẫn tiếp nhận RTP được bảo vệ

#### Giới thiệu

Bản đặc tả này định nghĩa một cơ chế để đánh dấu đã được bảo vệ cho các luồng đa phương tiện. Việc này được thực hiện bằng cách thay đổi mã bốn ký tự của SampleEntry, và dán những hộp có chứa chi tiết của cơ chế bảo vệ và mã bốn ký tự ban đầu. Tuy vậy, trong trường hợp này rãnh không được bảo vệ; nó là một rãnh chỉ dẫn'in the clear' có chứa dữ liệu được bảo vệ. Tiểu mục này miêu tả cách thức đánh dấu các rãnh chỉ dẫn tiếp nhậnkhi chúng mang dữ liệu được bảo vệ, sử dụng một cơ chế tương tự, và tận dụng cùng những hộp đó.

#### Cú pháp

*Class ProtectedRtpReceptionHintSampleEntry extends RtpReceptionHintSampleEntry (‘prtp‘) {*

*ProtectionSchemeInfoBoxSchemeInformation;*

*}*

#### Ngữ nghĩa

Hộp*SchemeInformation*(‘*sinf*’)chứa thông tin chi tiết của cơ chế bảo vệ được áp dụng. Nó sẽ bao gồm *OriginalFormatBox* có chứa mã bốn ký tự 'rrtp' (mã bốn ký tự của hộp RTPReceptionHintSampleEntry ban đầu).

### Thủ tục ghi

Xem phụ lục G.

### Thủ tục phân tích

Xem phụ lục G.

# Các nhóm mẫu

## Các điểm khôi phục truy nhập ngẫu nhiên

### Định nghĩa

Trong vài hệ thống mã hóa, có thể truy nhập ngẫu nhiên vào trong một luồng (lưu lượng) và đạt được việc giải mã cơ bản xác sau khi vừa thực hiện giải mã một số lượng các mẫu. Điều này được biết đến như là làm mới việc giải mã theo từng bậc. Lấy ví dụ, trong hình ảnh, bộ mã hóa có thể mã hóa các khối marco được mã hóa nội trong luồng lưu lượng, như vậy bộ mã hóa biết rằng trong một chu kỳ nhất định toàn bộ hình ảnh bao gồm các điểm ảnh mà chỉ phụ thuộc vào các khối đoạn mã macro được mã hóa nội đã được cung cấp trong suốt chu kỳ đó.

Các mẫu có kiểu làm mới từng bậc có thể được đánh dấu là một thành viên của một trong số các nhóm này. Việc định nghĩa các nhóm cho phép việc đánh dấu xuất hiện tại cả ở phần đầu và cuối của chu kỳ. Tuy nhiên, khi được sử dụng với một kiểuđa phương tiện cụ thể, việc sử dụng các nhóm này có thể bị giới hạn chỉ đánh dấu ở cuối chu kỳ (ví dụ việc hạn chế chỉ đối với các giá trị lặp lại dương hoặc âm). Một roll-group là một nhóm các mẫu có cùng khoảng cách lặp lại.

Các nhóm lặp lại có ý nghĩa như sau.

*VisualRollRecoveryEntry* lưu các mẫu cho phép kích hoạt các điểm đầu vào mục trong các luồng thay thế cho các mẫu đồng bộ.

*AudioRollRecoveryEntry* lưu các khoảng cách pre-roll cần thiết trong các luồng âm thanh, trong đó mỗi mẫu có thể được giải mã một cách độc lập, tuy nhiên đầu ra bộ giải mã chỉ đảm bảo cho kết quả đúng khi pre-rolling bởi số chỉ thị mẫu.

*AudioPreRollEntry* được dùng với các luồng âm thanh, trong đó không phải mẫu nào cũng là mẫu đồng bộ, việc giải mã chỉ bắt đầu tại mẫu đồng bộ, nhưng đầu ra bộ giải mã chỉ đảm bảo cho kết quả đúng khi pre-rolling bởi số chỉ thị mẫu. Điều đó có nghĩa để có được kết quả đầu ra đúng khi thực hiện truy nhập ngẫu nhiên thì trước tiên cần sao lưu khoảng cách pre-roll được chỉ thị, sau đó (để bắt đầu giải mã) tìm mẫu đồng bộ gần nhất hoặc trước vị trí cần truy nhập.

### Cú pháp

*class VisualRollRecoveryEntry() extends VisualSampleGroupEntry (’roll’) {*

*signed int(16) roll\_distance;*

*}*

*class AudioRollRecoveryEntry() extends AudioSampleGroupEntry (’ prol’) {*

*signed int(16) roll\_distance;*

*}*

*class AudioPreRollEntry() extends AudioSampleGroupEntry (’prol’) {*

*signed int(16) roll\_distance;*

*}*

### Ngữ nghĩa

*roll\_distance*có kiểu số nguyên có dấu cho biết số lượng các mẫu phải được giải mã để một mẫu được giải mã đúng. Giá trị dương cho biết số lượng các mẫu sau khi mẫu đó là thành viên nhóm mà nhóm nàyphải được giải mã tới khi mẫu cần khôi phục cuối cùng hoàn thành, ví dụ trường hợp mẫu cuối cùng được khôi phục đúng.Giá trị âm cho biết số lượng các mẫu trước khi mẫu là một thành viên nhóm phải được giải mã yêu cầu cho việc khôi phục được hoàn thành tại mẫu đã được đánh dấu. Giá trị 0 phải không được sử dụng; việc đồng bộ các điểm truy nhập ngẫu nhiên của bảng tài liệu mẫu thì không cần khôi phục khoảng cách cuốn.

## Các nhóm chia sẻ tỉ lệ

### Giới thiệu

Các chỉ thị tốc độ chia sẻ được sử dụng bởi nhiều bộ đọc và các máy chủ streaming để giúp đỡ việc phân bổ tốc độ bit động khi vài luồng cùng chia sẻ một băng thông tài nguyên chung. Các chỉ thị đã được lưu trữ trong tệp tin như nhóm các bản ghi mẫu và áp dụng khi mở rộng hoặc các rãnh luân phiên. Các hướng dẫn là độc lập thời gian như các mẫu trong một rãnh có thể được kết hợp với mẫu các nhóm bản ghi mẫu khác. Trong trường hợp đơn giản nhất, chỉ một đối tượng giá trị tỉ lệ chia sẻ được định nghĩa trên từng đa phương tiện và dải thời gian như lược đồ trong hình 5.



Hình 5 – Mối tương quan giữa tỉ lệ âm thanh hình ảnh và thời gian

Với mục tiêu điều tiết cho các giá trị tốc độ chia sẻ mà biến thiên với tốc độ bit có sẵn, nó có thể đặc tả nhiều hơn một dải vận hành. Một dải vận hành có thể cho ví dụ chỉ ra rằng âm thanh yêu cầu một tỉ lệ phần trăm cao hơn (là hình ảnh) tại tốc độ bit thấp có sẵn. Điều này được hoàn thành bởi việc đặc tả hai điểm hoạt động được chỉ ra trong Hình 6.



Hình 6 – Mỗi tương quan giữa tỉ lệ âm thanh và tốc độ bit sẵn có.

Các điểm hoạt động được định nghĩa trong các giới hạn tổng băng thông sẵn sàng. Đối với nhiều tình huống phức tạp hơn có thể đặc tả nhiều điểm hoạt động hơn.

Thêm vào các giá trị tỉ lệ chia sẻ mục tiêu, các điểm hoạt động cũng có thể đặc tả tốc độ cao nhất và tốc độ thấp nhất cho một kiểuđa phương tiện, cũng như bỏ qua giá trị ưu tiên.

### Đầu vào mục nhóm mẫu chia sẻ tỉ lệ

#### Định nghĩa

Mỗi mẫu trong một rãnh có thể được kết hợp với một trong số các nhóm miêu tả mẫu, một miêu tả mẫu là một bản ghi lưu thông tin chia sẻ tỉ lệ. Thông thường thông tin chia sẻ tỉ lệ đều được áp dụng tới nhiều mẫu liên tục và nó có thể do đó được đầy đủ để định nghĩa hai hoặc ba nhóm định nghĩa mẫu mà có thể được sử dụng tại nhiều khoảng thời gian tới khác nhau.

Kiểu nhóm ‘rash’ (từ viết tắt cho việc chia sẻtỉ lệ) được định nghĩa như là tiêu chí cho thông tin chia sẻ tỉ lệ. Có thể không có hoặc chỉ có một hộp ánh xạ mẫu vào nhóm (‘*sbgp*’) cho việc kiểunhóm ‘rash’ trong hộp bảng mẫu (‘*stbl*’) của một rãnh. Nó sẽ nằm trong rãnh chỉ dẫn, nếu một rãnh chỉ dẫn được sử dụng, nếu không nó nằm trong một rãnh đa phương tiện.

Tỉ lệ chia sẻ mục tiêu có thể được đặc tả cho một vài điểm hoạt động được định nghĩa trong các ngữ cảnh của tổng các tốc độ bit sẵn sàng, ví dụ tốc độ bit mà có thể được chia sẻ. Nếu chỉ một điểm hoạt động được định nghĩa, tỉ lệ chia sẻ mục tiêu áp dụng tới tất cả các tốc độ bit sẵn sàng. Nếu một vài điểm hoạt động được định nghĩa, thì từng điểm hoạt động đặc tả một tỉ lệ chia sẻ mục tiêu. Các giá trị tỉ lệ chia sẻ mục tiêu định nghĩa cho các điểm hoạt động đầu tiên và cuối cùng cũng định nghĩa các giá trị tỉ lệ chia sẻ mục tiêu tại các bitrate sẵn sàng tại mức thấp và mức cao, tương ứng. Tỉ lệ chia sẻ mục tiêu giữa các điểm hoạt động được đặc tả trong dải giữa các tốc độ mục tiêu chia sẻ của các điểm hoạt động đó. Giá trị có thể được ước lượng với phép suy tuyến tính.

#### Cú pháp

*class RateShareEntry() extends SampleGroupDescriptionEntry('rash') {*

*unsigned int(16) operation\_point\_count;*

*if (operation\_point\_count == 1) {*

*unsigned int(16) target\_rate\_share;*

*}*

*else {*

*for (i=0; i < operation\_point\_count; i++) {*

*unsigned int(32) available\_bitrate;*

*unsigned int(16) target\_rate\_share;*

*}*

*}*

*unsigned int(32) maximum\_bitrate;*

*unsigned int(32) minimum\_bitrate;*

*unsigned int(8) discard\_priority;*

*}*

#### Ngữ nghĩa

*operation\_point\_count* có kiểu số nguyên khác 0 cho biết số lượng các điểm hoạt động.

*available\_bitrate* có kiểu số nguyên dương định nghĩa một điểm hoạt động (giá trị tính bằng kb/s). Nó là tổng tốc độ bit sẵn sàng có thể được phân bổ trong các chia sẻcủa các rãnh. Mỗi đầu vào mục sẽ lớn hơn đầu vào mục trước đó.

*target\_rate\_share*có kiểu số nguyên. Giá trị khác 0 cho biết phần trăm của băng thông sẵn sàng sẽ được phân bổ tới đa phương tiện cho từng điểm hoạt động. Giá trị của điểm hoạt động đầu tiên (hay cuối cùng) áp dụng tới tốc độ bit sẵn sàng thấp hơn (hoặc cao hơn) là điểm hoạt động của cơ bản nó. Tỉ lệ chia sẻ mục tiêu giữa các điểm hoạt động được bao quanh bởi các tỉ lệ chia sẻ mục tiêu của các điểm hoạt động tương ứng. Giá trị 0 cho biết không có thông tin trên tỉ lệ chia sẻ phần trăm ưu tiên hơn nào được cung cấp.

*maximum\_bitrate*có kiểu số nguyên. Giá trị khác 0 cho biết (được tính bằng kb/s) một ngưỡng cao hơn cho băng thông mà sẽ được phân bổ tới đa phương tiện. Một tốc độ cao hơn giá trị tốc độ bit cao nhất sẽ chỉ được phân bổ nếu tất cả các đa phương tiện khác trong phiên đã bị chiếm hết tỉ lệ chia sẻ mục tiêu và tốc độ bit cao nhất, tương ứng. Giá trị 0 cho biết không thông tin về tốc độ bit cao nhất được cung cấp.

*minimum\_bitrate*có kiểu số nguyên. Giá trị không phải 0 cho biết (được tính bằng kb/s) một ngưỡng thấp hơn cho băng thông mà sẽ được phân bổ tới đa phương tiện. Nếu băng thông được phân bổ tương ứng tới một giá trị nhỏ hơn, sau đó không tốc độ bit nào được phân bổ. Thay vì được ưu tiên đem tới đa phương tiện khác trong phiên hoặc luân phiên các mã hoặc của cùng đa phương tiện. Tốc độ bit nhỏ nhất bằng 0 cho biết không có thông tin với tốc độ bit nhỏ nhất được cung cấp.

*discard\_priority* là một giá trị nguyên cho biết mức ưu tiên của rãnh khi các rãnh bị kiểu bỏ để đạt các thiết lập cưỡng bức bởi tỉ lệ chia sẻ mục tiêu, tốc độ bit cao nhất và tốc độ bit thấp nhất. Các rãnh bị kiểu bỏ trong yêu cầu ưu tiên kiểu bỏ và rãnh nào có giá trị mức ưu tiên kiểu bỏ cao nhất sẽ bị kiểu bỏ trước tiên.

### Mối quan hệ giữa các rãnh

Mục đích của việc định nghĩa thông tin tỉ lệ chia sẻ là để hỗ trợ một máy chủ hoặc một bộ đọc hình ảnh giải nén dữ liệu từ một rãnh trong sự kết hợp với nhiều rãnh khác. Lưu ý rằng một máy chủ/đầu đọc truyền theo luồng/đọc các rãnh cùng một lúc nếu chúng thuộc các nhóm luân phiên nhau và có thể chuyển đổi giữa các rãnh về cùng nhóm chuyển trong một nhóm luân phiên. Theo mặc định, tất cả các rãnh đều được phục vụ/đọc hình ảnh đồng thời nếu các nhóm luân phiên không được định nghĩa.

Thông tin tỉ lệ chia sẻ sẽ được cung cấp tới cho từng rãnh. Một rãnh mà không bao gồm thông tin tỉ lệ chia sẻ có một điểm hoạt động và có thể được đối xử như là một rãnh có tốc độ bit cố định với kiểu trừ ưu tiên 128. Chia sẽ tốc độ mục tiêu, các tốc độ bit nhỏ nhất và lớn nhất không được áp dụng trong trường hợp này.

Các rãnh được luân phiên cho từng rãnh khác (tại từng trường hợp theo thời gian) sẽ định nghĩa cùng số lượng điểm hoạt động tại cùng tập hợp của tổng hợp các tốc độ bit tồn tại và có cùng các mức ưu tiên kiểu bỏ. Lưu ý rằng số lượng và định nghĩa các điểm hoạt động có thể phụ thuộc vào thời gian. Các rãnh luân phiên có thể có các tỉ lệ chia sẻ mục tiêu khác, là các tốc độ bit nhỏ nhất và lớn nhất.

### Phân bổ tốc độ bit

Thông tin tỉ lệ chia sẻ dựa trên tốc độ bit lớn nhất, tốc độ bit nhỏ nhất, và chia sẻ tỉ lệ mục tiêu có thể kết hợp cho một rãnh. Nếu trường hợp này, việc chia sẻ tỉ lệ mục tiêu sẽ được áp dụng để tìm kiếm một tốc độ bit được phân bổ trước khi khoảng tốc độ bit lớn nhất và nhỏ nhất được xem xét.

Khi phân bổ băng thông cho một số rãnh, cần lưu ý:

1. Trong trường hợp tất cả các rãnh có các giá trị tỉ lệ chia sẻ mục tiêu rõ ràng và tổng của chúng không tới 100 phần trăm thì đối xử với các rãnh theo các tỉ trọng, ví dụ bình thường hóa các rãnh này.
2. Tổng phân bổ sẽ không vượt quá tổng tốc độ bit sẵn sàng.
3. Trong một lựa chọn giữa các rãnh luân phiên, rãnh được chọn sẽ là rãnh là nguyên nhân tạo nhóm luân phiên có một phân bổ gần nhất phù hợp với tỉ lệ chia sẻ mục tiêu của nó, hoặc rãnh mà tốc độ mong muốn cao nhất có thể được phân bổ mà không phải kiểu bỏ các rãnh khác (xem bên dưới).
4. Các rãnh phải có một phân bổ giữa các tốc độ bit nhỏ nhất và tốc độ bit lớn nhất, hoặc là bị kiểu bỏ.
5. Các rãnh có một phân bổ phù hợp với các chia sẻ tỉ lệ mục tiêu của chúng, nhưng có thể bị sai khác để cho phép một vài rãnh hoàn thành giá trị nhỏ nhất của chúng, hoặc trong trường hợp một vài rãnh tiến tới giá trị lớn nhất.
6. Nếu một phân bổ không được hoàn thành bao gồm một rãnh từ từng nhóm luân phiên, sau đó các rãnh sẽ bị kiểu bỏ trong yêu cầu kiểu bỏ theo ưu tiên.
7. Việc phân bổ phải được tính toán kiểu bất kể khi nào tập thi hành cho một rãnh hoạt động (một rãnh mà được lựa chọn từ một nhóm luân phiên) thay đổi hoặc thay đổi tốc độ bit tồn tại.

## Các chuỗi khởi động luân phiên

### Định nghĩa

Một trình tự khởi động luân phiên chứa một tập hợp con của các mẫu của một rãnh trong một chu kỳ nhất định khởi động từ một mẫu đồng bộ hoặc từ một mẫu được đánh dấu bởi nhóm mẫu theo ‘rap’, trong đó được thu thập được xem tới như là mẫu dưới đây. Bằng cách giải mã tập hợp con của các mẫu này, việc vẽ các mẫu có thể được bắt đầu sớm hơn trong trường hợp khi tất cả các mẫu được giải mã.

Một bản ghi miêu tả nhóm mẫu ‘alst’ chỉ ra số lượng các mẫu trong bất kỳ các chuỗi khởi động luân phiên tương ứng nào, sau khi tất cả các mẫu được xử lý.

Cả phiên bản 0 hoặc phiên bản 1 của Hộp Ánh xạ Mẫu và Nhóm có thể được sử dụng với nhóm mẫu khởi động tuần tự luân phiên. Nếu phiên bản 1 của Hộp Ánh xạ Mẫu và Nhóm được sử dụng, tham số group\_type*\_parameter* không được định nghĩa theo ngữ nghĩa nhưng có cùng thuật toán để chuyển hóa các chuỗi khởi động luân phiên sẽ được sử dụng nhất quán cho một giá trị riêng cảu tham số *grouping\_type \_parameter*.

Một đầu đọc sử dụng các chuỗi khởi động luân phiên có thể hoạt động như sau. Đầu tiên, một mẫu đồng bộ ban đầu để bắt đầu giải mã hóa được xác định bằng việc sử dụng Hộp Mẫu Đồng bộ, cờ *sample\_is\_non\_sync\_sample* cho các mẫu bao quanh trong các phân mảnh của rãnh, hoặc nhóm các mẫu ‘rap’. Sau đó, nếu mẫu đồng bộ ban đầu được kết hợp với bản ghi đặc tả nhóm mẫu của kiểu ‘alast’ nơi roll\_count thì lớn hơn là 0, bộ đọc có thể sử dụng chuỗi khởi động luân phiên. Đầu đọc sau đó giải mã chỉ những mẫu mà được ánh xạ tới chuỗi khởi động luân phiên cho đến khi một số lượng mẫu mà được giải mã bằng với giá trị *roll\_count*. Sau đó, tất cả các mẫu đều được giải mã.

### Cú pháp

*class AlternativeStartupEntry() extends VisualSampleGroupEntry (’alst’)*

*{*

*unsigned int(16) roll\_count;*

*unsigned int(16) first\_output\_sample;*

*for (i=1; i <= roll\_count; i++)*

*unsigned int(32) sample\_offset[i];*

*j=1;*

*do { // optional, until the end of the structure*

*unsigned int(16) num\_output\_samples[j];*

*unsigned int(16) num\_total\_samples[j];*

*j++;*

*}*

*}*

### Ngữ nghĩa

*roll\_count* chỉ số lượng các mẫu trong chuỗi khởi động luân phiên. Nếu giá trị roll\_count bằng 0, mẫu kết hợp không liên kết với bất kỳ chuỗi khởi động luân phiên nào và các cách giải thích ngữ nghĩa cụm từ *first\_output\_sample* không được đặc tả. Số lượng các mẫu được ánh xạ tới bản ghi nhóm mẫu trên từng chuỗi khởi động luân phiên sẽ bằng với giá trị *roll\_count*.

*first\_output\_sample*cho biết chỉ số của mẫu đầu tiên dự định cho đầu ra giữa các mẫu trong chuỗi khởi động luân phiên. Chỉ số của mẫu khởi tạo đồng bộ bắt đầu chuỗi khởi động luân phiên là 1, lần lượt chỉ số được tăng lên 1, để giải mã, từng mẫu một trong chuỗi khởi động luân phiên.

*sample\_offset[i]* chỉ giá trị thời gian delta của việc giải mã của mẫu thứ I trong chuỗi khởi động luân phiên liên quan tới thời gian giải mã thường xuyen của mẫu được lấy từ Thời gian giải mã tới Hộp mẫu (Decoding Time to Sample Box) hoặc theHộp Mào đầu Phân mảnh Rãnh. Mẫu khởi tạo đồng bộ bắt đầu chuỗi khởi động luân phiên là mẫu đầu tiên của nó.

*num\_output\_samples[j]* và *num\_total\_samples[j]* chỉ tốc độ mẫu đầu ra trong chuỗi khởi động tuần tự. Chuỗi khởi động tuần tự được phân chia thành k phần liên tục, mỗi phần có một tốc độ mẫu đầu ra không đổi mà tốc độ của mỗi phần thì không bằng với phần liền kề với nó. Phần đầu tiên bắt đầu từ mẫu là *first\_output\_sample*.

*num\_output\_sample[j]* chỉ số lượng các mẫu đầu ra của j phần của chuỗi khởi động luân phiên.

*num\_total\_samples[j]* chỉ số lượng tổng các mẫu, bao gồm những mẫu mà không trong chuỗi khởi động luân phiên, từ mẫu đầu tiên trong phần thứ j mà đầu ra tới một của mẫu mới hơn mà kết thúc chuỗi khởi động luân phiên và mẫu mà đến trước ngay lập tức của mẫu đầu ra đầu tiên của phần thứ (j+1).

### Một số ví dụ

Khả năng mở rộng phân cấp thời gian (ví dụ trong AVC và SVC) cải thiện hiệu năng nén nhưng lại làm tăng lên độ trễ giải mã do yêu cầu của việc giải mã hình ảnh từ yêu cầu mã hóa/ giải mã hóa theo yêu cầu của đầu ra. Các phân cấp thời gian sâu được chứng minh là hữu ích trong các khoảng thời gian nén hiệu quả trong một vài trường hợp nghiên cứu. Khi phân cấp thời gian sâu và tốc độ vận hành của bộ giải mã bị giới hạn (không nhanh hơn được so với xử lý thời gian thực), độ trễ ban đầu từ khi bắt đầu của việc giải mã tới khi bắt đầu việc phục dựng hình ảnh là đáng kể và có thể ảnh hưởng tiêu cực tới trải nghiệm của người dùng cuối.

Hình 7 minh họa một dòng bit mở rộng phân cấp điển hình với 5 mức thời gian. Hình 7a chỉ ra ví dụ chuỗi yêu cầu của đầu ra. Các giá trị đã bao xung quanh trong các hộp chỉ giá trị *frame\_num* của hình vẽ.

Các giá trị viết in nghiêng chỉ một hình ảnh không tham chiếu trong khi các hình ảnh khác là các hình ảnh tham chiếu. Hình 7b chỉ ra ví dụ chuỗi trong giải mã. Hình 7c chỉ ra ví dụ chuỗi trong đầu ra khi giả định rằng khoảng thời gian đầu ra trùng với khoảng thời gian giải mã và việc giải mã một hình ảnh kéo dài trong khoảng thời gian một khung hình. Có thể thấy rằng việc phát lại một luồng bắt đầu bằng 1 độ trễ sau khi 5 khung hình được giải mã bắt đầu. Nếu các hình ảnh có giá trị lấy mẫu là 25 Hz, độ trễ khung hình là 40ms, và việc phát bị trễ với giá trị là 0,2s.



Hình 7 – Trễ bộ đệm giải mã hình ảnh của một chuỗi 5 mức thời gian.

Nhờ hệ thống phân cấp thời gian, có thể giải mã chỉ một tập hợp con của cáchình ảnh tại điểm bắt đầu của chuỗi. Do đó, việc dựng hình có thể được bắt đầu nhanh hơn nhưng tốc độ hiển thị hình ảnh là thấp hơn tại lúc thời điểm ban đầu. Nói cách khác, một đầu đọc có thể thực hiện sự đánh đổi giữa thời gian trễ khởi động ban đầu và tốc độ hiển thị hình ảnh ban đầu. Hình 8 và Hình 9 cho thấy hai ví dụ về các chuỗi khởi động luân phiên nơi một tập hợp con của luồng bit của Hình 7 được giải mã.

Các mẫu được lựa chọn để giải mã và đầu ra bộ giải mã được trình bày trong Hình 8a và 8b hình, tương ứng. Hình ảnh tài liệu tham khảo có *frame\_num* bằng 4 và những hình ảnh không có tài liệu tham khảo *frame\_num* bằng 5 không giải mã. Trong ví dụ này, cung cấp hình ảnh bắt đầu bốn khoảng thời gian hình ảnh sớm hơn trong hình 7. Khi tỷ lệ hình ảnh là 25 Hz, tiết kiệm trong sự chậm trễ khởi động là 160 mili giây. Tiết kiệm trong sự chậm trễ khởi động đi kèm với những bất lợi của tỷ lệ hình ảnh hiển thị thấp hơn ở đầu luồng bit.



Hình 8 – Ví dụ của một chuỗi khởi tạo thay thế.

Thay vì ví dụ trong Hình 9, một cách thức khác để lựa chọn các hình ảnh cho việc giải mã được giới thiệu. Việc giải mã các hình ảnh mà phụ thuộc vào hình ảnh với *frame\_num* bằng 3 được bỏ qua và việc giải mã các hình ảnh không tham chiếu trong nửa giây của nhóm các hình ảnh đầu tiên cũng được bỏ qua. Kết quả của việc giải mã hình ảnh từ mẫu với *frame\_num* bằng 2 là hình ảnh đầu tiên tại đầu ra. Như một kết quả, tốc độ hình ảnh đầu ra của nhóm các hình ảnh đầu tiên bằng một nửa tốc độ hình ảnh bình thường, nhưng việc xử lý hình ảnh bắt đầu với thời gian bằng 2 khung hình (80ms tại tốc độ quét 25Hz) sớm hơn so với giải pháp thông thường minh họa trong Hình 7.



Hình 9 - Ví dụ của một chuỗi khởi tạo thay thế.

## Nhóm mẫu điểm truy nhập ngẫu nhiên

### Định nghĩa

Một mẫu đồng bộ được đặc tả là một điểm truy nhập ngẫu nhiên sau khi tất cả các mẫu trong yêu cầu giải mã có thể được giải mã đúng. Tuy nhiên, có thể giải mã một điểm truy nhập ngẫu nhiên “mở “, sau khi tất cả các mẫu tại đầu ra có thể được giải mã đúng, nhưng một vài mẫu theo sau điểm truy nhập ngẫu nhiên để giải mã và đứng trước điểm truy nhập ngẫu nhiên trong yêu cầu không cần thiết đầu ra phải có khả năng giải mã đúng. Lấy ví dụ, một hình ảnh nội tại bắt đầu một nhóm các ảnh mở có thể được theo sau trong việc giải mã bởi các hình ảnh tiên đoán trước mà tuy nhiên hình ảnh nội suy trước trong yêu cầu tại đầu ra; mặc dù chúng có thể không thể được giải mã đúng nếu việc giải mã bắt đầu từ ảnh nội suy, chúng là không được cần.

Đối với các mẫu truy nhập ngẫu nhiên “mở “có thể được đánh dấu bởi một thành viên đang tồn tại của nhóm này. Các mẫu được đánh dấu bởi nhóm này phải là các điểm truy nhập ngẫu nhiên, và có thể cũng là các điểm đồng bộ (đó là: nó không yêu cầu các mẫu được đánh dấu bởi bảng mẫu đồng bộ được kiểu trừ.

### Cú pháp

*class VisualRandomAccessEntry() extends VisualSampleGroupEntry (’rap ’)*

*{*

*unsigned int(1) num\_leading\_samples\_known;*

*unsigned int(7) num\_leading\_samples;*

*}*

### Ngữ nghĩa

*num\_leading\_samples\_known* bằng 1 chỉ rằng số lượng các mẫu hàng đầu được biết đến cho từng mẫu trong nhóm này, và số lượng được đặc tả bởi thông số *num\_leading\_samples*. Một mẫu dẫn đầu là một mẫu kết hợp với một điểm truy nhập ngẫu nhiên (RAP: Random Access Point)”mở “. Trước RAP giới thiệu và theo ngay sau RAP hoặc mẫu dẫn đầu khác trong việc giải mã, và khi giải mã bắt đầu từ RAP, mẫu không thể được giải mã cơ bản xác.

*num\_leading\_samples* đặc tả số lượng các mẫu dẫn đầu cho từng mẫu trong nhóm này. Khi giá trị *num\_leading\_samples\_known* bằng 0, trường này sẽ bị bỏ qua.

## Gộp nhóm mẫu theo mức thời gian

### Định nghĩa

Nhiều hình ảnh codec hỗ trợ khả năng mở rộng theo thời gian, nơi nó có thể trích xuất một hoặc nhiều tập hợp con của khung hình có thể được giải mã một cách độc lập. Một trường hợp đơn giản là đưa ra các khung hình I cho một luồng bit với một khoảng thời gian khung hình I thường xuyên, ví dụ IPPPIPPP …, cứ ảnh thứ 4 là một khung hình I. Cũng tập hợp con của những khung hình I có thể được trích xuất cho cả những tốc độ khung hình thấp hơn. Nhiều tình huống phức tạp với nhiều mức thời gian có thể được xây dựng bằng các khung phân cấp B hoặc P.

Nhóm mẫu MứcThời gian (Temporal Level) ('tele') cung cấp một nhóm mẫu có codec độc lập có thể được sử dụng cho nhóm mẫu (các đơn vị truy nhập) trong một rãnh (và các khả năng phân mảnh của rãnh) theo mức thời gian, nơi các mẫu của một cấp thời gian không có mã hóa phụ thuộc vào mẫu của các cấp thời gian cao hơn. Mức thời gian bằng với chỉ số miêu tả nhóm mẫu (lấy các giá trị 1, 2, 3,..). Luồng bit chỉ chứa các đơn vị truy nhập từ cấp thời gian đầu tiên đến một mức độ thời gian cao hơn còn lại phù hợp với tiêu chuẩn mã hóa.

Một nhóm theo mức độ thời tạo điều kiện chích xuất dễ dàng các chuỗi con thời gian, ví dụ bằng cách sử dụng hộp Subsegment Indexing mục trong 8.16.4.

### Cú pháp

*class TemporalLevelEntry() extends VisualSampleGroupEntry('tele')*

*{*

*bit(1) level\_independently\_decodable;*

*bit(7) reserved=0;*

*}*

### Ngữ nghĩa

Các mẫu theo cấp thời gian trong một nhóm mẫu bằng với chỉ số miêu tả nhóm mẫu.

*level\_independently\_decodable* là một cờ. Giá trị 1 chỉ ra rằng tất cả các mẫu của mức này không có các phụ thuộc mã hóa vào các mẫu của các cấp khác. Giá trị 0 chỉ ra rằng không có thông tin được cung cấp.

# Khả năng mở rộng

## Các đối tượng

Các đối tượng có giá trị định nghĩa trong tài liệu đặc tả này được xác định bằng một giá trị 32bit, giá trị này là một tập hợp của bốn ký tự in thường nằm trong tập hợp ký tự của ISO 8859-1.

Để cho phép người sử dụng mở rộng các định dạng, để lưu trữ các kiểu đối tượng mới, và để cho phép việc liên hoạt động của các tệp tin được định dạng theo đặc tả của tài liệu này với các môi trường tính toán được phân tán nhất định, có một kiểu bảng ánh xạ và một kiểu cơ chế mở rộng mà ghép cặp được với nhau.

Các đối tượng thường được sử dụng trong tính toán phân tán là UUID (Universal Unique Identifiers: Định danh duy nhất toàn cầu), có 16 byte. Bất kỳ kiểu đặc tả quy phạm nào ở đây đều có thể ánh xạ trực tiếp vào không gian UUID bằng cách ghép kết hợp 4 kiểu giá trị byte,với giá trị 12 byte ISO được dành riêng, 0xXXXXXXXX-0011-0010-8000-00AA00389B71. Bốn mã ký tự thay thế XXXXXXXX trong số liền trước. Những kiểu được xác định theo tiêu chuẩn ISO như là các kiểu đối tượng được sử dụng trong tài liệu đặc tả này.

Các đối tượng người dùng sử dụng kiểu thoát ‘uuid’. Kiểu thoát này đã được miêu tả bằng tài liệu trong mục 6.2. Sau kích cỡ và các kiểu trường, đây là 1 giá trị UUID 16byte đầy đủ.

Các hệ thống mà có mong muốn đối xử với bất kỳ đối tượng nào như có một UUID có thể sử dụng các thuật toán sau đây:

*size := read\_uint32();*

*type := read\_uint32();*

*if (type==‘uuid’)*

*then uuid := read\_uuid()*

*else uuid := form\_uuid(type, ISO\_12\_bytes);*

Tương tự như vậy khi tuyến tính hóa một tập các đối tượng vào các tệp tin định dạng đặc điểm kỹ thuật này, sau đây được áp dụng:

*write\_uint32*(*object\_size(object)*)*;*

*uuid := object\_uuid\_type(object);*

*if (is\_ISO\_uuid(uuid)*)

*write\_uint32*(*ISO\_type\_of(uuid)*)

*else { write\_uint32(‘uuid’); write\_uuid(uuid); }*

Một tệp tin có chứa các hộp từ đặc tả này đã được viết bằng cách sử dụng 'uuid' và UUID đầy đủ là không phù hợp; các hệ thống này không cần thiết để nhận ra các hộp tiêu chuẩn được viết bằng cách sử dụng 'uuid’ và một UUID chuẩn ISO.

## Các định dạng lưu trữ

Các tệp tincơ bản chứa các siêu dữ liệu có thể sử dụng các tệp tin khác để có phương tiện truyền thông dữ liệu. Những tệp tin khác có thể chứa các khai báo đánh đầu từ một loạt các tiêu chuẩn, trong đó có một này.

Tệp tincơ bản chứa siêu dữ liệu có thể sử dụng các tệp tin khác để chứa dữ liệu đa phương tiện. Những tệp tin khác chứa các khai báo mào đầu từ nhiều tiêu chuẩn, bao gồm tiêu chuẩn này.

Nếu một tệp tin thứ 2 có một tập khai báo siêu dữ liệu thìsiêu dữ liệu đó không phải là một phần của toàn bộ trình diễn. Điều này cho phép các tệp tintrình diễn nhỏ được gộp thành một trình diễn toàn bộ lớn hơn bằng cách xây dựng một siêu dữ liệu mới và tham chiếu dữ liệu đa phương tiện, hơn là sao chép nó.

Các tham chiếu vào những tệp tin khác không sử dụng tất cả dữ liệu trong những tệp tin đó; theo cách này, một tập hợp con siêu dữ liệu có thể được sử dụng, hoặc các mào đầu không được mong đợi bị bỏ qua.

## Các định dạng tệp tinphái sinh

Đặc tả này có thể được sử dụng làm cơ sở đặc tả định dạng tệp tin cho một mục đích hạn chế: lấy ví dụ, định dạng tệp tin MP4 cho MPEG-4 và định dạng tệp tin Motion JPEG 2000 đều bắt nguồn từ định dạng gốc. Khi một trích dẫn đặc tả được viết ra, phải xác định nhưsau:

Tên của định dạng mới, và các nhánh của nó và các kiểu tương thích cho Hộp Kiểu Tệp tin. Nói chung một tệp tin mở rộng mới sẽ được sử dụng, một kiểu MIME mới, và cũng là kiểutệp tin Macintosh, mặc dù việc định nghĩa và đăng ký của các định dạng mới nằm ngoài phạm vi của bản đặc tả này.

Bất kỳ các trường khuôn mẫu nào được sử dụng phải được khai báo rõ ràng; việc sử dụng phải được tuân theo các đặc tả ở đây.

Các định danh cơ bản xác “codingname “và “protocol” được sử dụng trong miêu tả mẫu phải được xác định. Định dạngcủa các mẫu mà các điểm mã hóa xác định cũng phải được xác định. Tuy nhiên, nó có thể có khả năng thích hợp để phù hợp với các hệ thống mã hóa mới vào một khung làm việc hiện tại (ví dụ như các hệ thống khung làm việc MPEG-4), hơn để xác định các điểm mã hóa mới ở mức này. Ví dụ, một định dạng âm thanh mới có thể sử dụng một codingname mới, hoặc có thể sử dụng 'mp4a' và đăng ký định danh mới trong khung làm việc âm thanh MPEG-4.

Các hộp mới có thể được định nghĩa, mặc dù điều này không được khuyến khích.

Nếu các đặc tả kỹ thuật đã dẫn cần một kiểu rãnh mới khác với hình ảnh, âm thanh, thì một kiểu xử lý mới phải được đăng ký. Mào đầuđa phương tiện cần thiết cho rãnh này phải được xác định. Nếu nó là một hộp mới, nó phải được định nghĩa và Kiểu hộp được đăng ký. Nói chung, kỳ vọng rằng hầu hết các hệ thống có thể sử dụng các kiểu rãnh hiện có.

Bất kỳ kiểu rãnh tham chiếu mới nào đều nên được đăng ký và định nghĩa. Theo định nghĩa trên, định dạng mô tả mẫu có thể được mở rộng với các hộp tùy chọn hoặc bắt buộc. Cú pháp thông thường để làm điều này sẽ được xác định một hộp mới với một tên cụ thể, mở rộng (ví dụ) Đầu vào mục Mẫu Hình ảnh và chứa các hộp mới.

# Phụ lục A (Tham khảo) Tổng quan

**A.1 Tổng quan**

Phụ lục A cung cấp cho người đọc những thông tin tổng quát về định dạng tệp tin âm thanh hình ảnh đồng thời bổ sung thêm thông tin tham khảo cho tài liệu chuẩn này.

**A.2 Những khái niệm cơ bản**

Đối với định dạng tệp tin âm thanh hình ảnh, dạng điển hình nhất là định dạng của một tệp tin movie. Về mặt logic, tệp tinmovie được chia thành các rãnh, mỗi rãnh chứa một chuỗi các thông tin được sắp xếp theo trình tự thời gian (ví dụ như các khung hình ảnh). Trong mỗi rãnh, mỗi đơn vị thông tin được sắp xếp theo thời gian được gọi là một mẫu, mẫu này có thể là một khung hình ảnh hoặc khung âm thanh được đánh số thứ tự. Một điểm cần lưu ý, một khung âm thanhcũng có thể được giải nén ra thành một chuỗi nhiều mẫu âm thanh; về tổng quan, tiêu chuẩn này sử dụng thuật ngữ mẫu khi nói đến một khung hoặc một đơn vị dữ liệu đã được đánh dấu thời gian. Mỗi rãnh có thể có một hoặc nhiều bảng miêu tả cho các mẫu (bảng mẫu), trong đó, mỗi mẫu trong rãnh gắn với một bảng miêu tả. Phần mô tả định nghĩa cách thức các mẫu có thể được giải mã (ví dụ nó chỉ ra thuật toán nén đã sử dụng).

Không giống với những định dạng tệp tin đa phượng tiện khác, định dạng này ngay từ đầu đã được chia thành một số phần có liên kết với nhau. Nắm được những phần riêng lẻ đó là chìa khóa để hiểu được định dạng tệp tin đề cập đến trong tài liệu này. Cụ thể:

Cấu trúc vật lý của tệp tin không nhất thiết phải gắn với cấu trúc vật lý của đa phương tiện. Ví dụ, rất nhiều định dạng tệp tin đóng khung dữ liệu đa phương tiện, bổ sung phần mào đầu và các dữ liệu khác ngay trước và sau khung hình ảnh, định dạng này không thực hiện theo phương thức đó.

Cả cấu trúc vật lý của tệp tin và cách sắp xếpđa phương tiện đều không bắt buộc phải gắn với thứ tự theo thời gian của đa phương tiện. Các khung hình ảnh cũng không cần phải sắp xếp theo trật tự thời gian trong tệp tin.

Điều này có nghĩa là có những cấu trúc tệp tin miêu tả vị trí và thứ tự thời gian của đa phương tiện, những cấu trúc này cho phép, nhưng không bắt buộccác tệp tin phải được sắp xếp theo trật tự thời gian.

Tất cả dữ liệu trong tệp tin hợp lệ đều được đóng gói trong trong các “hộp “(trước đây được gọi là nguyên tử - atom). Không có bất cứ dữ liệu nào nằm ngoài cấu trúc hộp này. Tất cả siêu dữ liệu và thông tin về thời gian và vị trí của đa phương tiện đều được đóng gói trong hộp. Tiêu chuẩn này cũng đưa ra định nghĩa “hộp “. Các dữ liệu đa phương tiện (ví dụ như các khung hình ảnh) đều được tham chiếu trong siêu dữ liệu. Dữ liệu đa phương tiện có thể thuộc cùng một tệp tin (trong một hoặc nhiều hộp) hoặc thuộc các tệp tin khác nhau, siêu dữ liệu cho phép tham chiếu đến các tệp tin khác nhau thông qua URL. Vị trí của dữ liệu đa phương tiện thuộc các tệp tin thứ cấp được miêu tả đầy đủ bởi siêu dữ liệu trong tệp tin sơ cấp. Định dạngcủa tệp tin thứ cấp có thể tuân theohoặc không tuân theo định dạng của được chỉ ra trong tiêu chuẩn này, vì vậy tệp tin thứ cấp có thể không có hộp.

Có thể có nhiều kiểu rãnh khác nhau, tuy nhiên có 3 kiểu rãnh quan trọng, đó là rãnh hình ảnh (video track), rãnh âm thanh (audio track) và rãnh chỉ dẫn (hint track). Rãnh hình ảnh chứa những mẫu hình ảnh, rãnh âm thanh chứa các mẫu âm thanh. Rãnh chỉ dẫn chứa những thông tin chỉ dẫn để chỉ cho máy chủ streaming biết cách thức đóng gói dữ liệu cho giao thức streaming từ các rãnh đa phương tiện có trong tệp tin. Rãnh chỉ dẫn chỉ liên quan đến streaming nên nó có thể sẽ không được dùng đến khi tệp tin được đọc ở chế độ cục bộ.

**A.3 Cấu trúc vật lý của đa phương tiện**

Các hộp thể hiệncách sắp xếp dữ liệu đa phương tiện được chỉ ra trong bảng mẫu. Các hộp này cũng chứathông tin tham chiếu dữ liệu, bảng kích thước mẫu, bảng ánh xạ mẫu vào đoạn, và bảng dịch chuyển đoạn. Những bảng này cho biết vị trí, kích thước của từng mẫu trong một rãnh.

Các thông tin tham chiếu dữ liệu cho phép xác định vị trí của đa phương tiện trong các tệp tin thứ cấp. Điều này cho phép hình thành nên một “thư viện “các tệp tinđa phương tiện riêng mà không cần sao chép đa phương tiện vào từng tệp tin riêng lẻ. Việc bố trí dữ liệu như vậy rất thuận tiện khi phải chỉnh sửa.

Các bảng trên còn được nén lại để tiết kiệm không gian. Ngoài ra, cơ chế đan xen được mong muốn không chỉ thực hiện theo từng mẫu mà còn thực hiện theo từng nhóm các mẫu liên tiếp nhau trong cùng một rãnh, hay từng nhóm các mẫu trong các rãnh khác nhau,... Một nhóm các mẫu liên tiếp trong một rãnh được gọi là đoạn (chunk). Mỗi đoạn có một giá trị dịch chuyển (offset) trong cơ bảntệp tin chứa chúng (tính từ điểm bắt đầu của tệp tin). Trong một đoạn, các mẫu được lưu giữ liên tiếp nhau,vì vậy, nếu một đoạn chứa 2 mẫu, vị trí của mẫu thứ 2 được xác định bằng cách cộng thêm kích thước của mẫu thứ nhất với giá trị dịch chuyển của đoạn đó. Bảng dịch chuyển đoạn chứa các giá trị dịch chuyển, bảng ánh xạ mẫu vào đoạn chứa thông tin ánh xạ giữa số thứ tự của mẫu sang số thứ tự của đoạn.

Một điểm cần lưu ý, giữa các đoạn có thể có “không gian chết “không được dữ liệu đa phương tiện tham chiếu đến. Vì vậy, trong quá trình chỉnh sửa, nếu tồn tại dữ liệuđa phương tiện không sử dụng, chỉ cần không tham chiếu đến và không cần sao chép lại dữ liệu đó khi xóa. Tương tự như vậy, nếu dữ liệuđa phương tiện trong tệp tin thứ cấp cókiểu định dạng tệp tin kiểu khác, có phần mào đầu hoặc cấu trúc theo định dạng khác đó thì dữ liệu đa phương tiện này có thể được bỏ qua.

**A.4 Cấu trúc thời gian của đa phương tiện**

Thời gian trong tệp tin có thể được xác định thông qua nhiều cấu trúc. Trong tệp tin movie, mỗi rãnh đều có một thang thời gian. Thang thời gian này có một trục thời gian được chia theo số lần lấy mẫu trong một giây. Có thể xác định cơ bản xác về thời gian nếu chọn được số lần phù hợp. Thông thường, với rãnh âm thanh, con số này cơ bản là tốc độ lẫy mẫu. Với hình ảnh thì lại sử dụng một con số hợp lý khác. Ví dụ, một thang thời gian đa phương tiện 30000 với khoảng thời gian lấy mẫu là 1001 sẽ tương ứng với chuẩn hình ảnh NTSC (thông thường có giá trị không hoàn toàn cơ bản xác là 29,97), cung cấp 19,9 giờ chiếu theo chuẩn 32 bit.

Cấu trúc thời gian của một rãnh có thể bị ảnh hưởng của một danh sách biên tập. Điều này dẫn đến 2 khả năng: sự dịch chuyển (và có thể tái sử dụng) của các phần trong dòng thời gian của rãnh trong toàn bộ movie và cũng có thể chèn các khoảng thời gian trống trong *rãnh*, được gọi là những hiệu chỉnh trống. Cần lưu ý đặc biệt là nếu một rãnh không bắt đầu trùng với điểm khởi đầu của một trình diễn thì cần phải có một hiệu chỉnh trống ban đầu.

Tổng thời gian của mỗi rãnh được xác định trong các mào đầu. Mỗi mẫu cũng có một khoảng thời gian xác định. Việc xác định cơ bản xác thời điểm trình diễn của một mẫu bất kỳ bằng tổng thời gian của các mẫu trước nó.

**A.5 Đan xen**

Cấu trúc vật lý của tệp tin có thể được sắp xếp theo cấu trúc thời gian. Điều này có nghĩa là cấu trúc vật lý của dữ liệu đa phương tiện được sắp xếp theo một trật tự thời gian. Ngoài ra, nếu dữ liệu đa phương tiện cho nhiều rãnh được chứa trong cùng một tệp tin, các dữ liệu này nên được sắp xếp đan xen. Thông thường, để đơn giản hóa việc đọc dữ liệu đa phương tiện trong một rãnh và để các bảng thông tin ngắn gọn thì cơ chế đan xen được thực hiện theo các khoảng thời gian (ví dụ là 1 giây) chứ không thực hiện theo các mẫu. Điều này làm giảm số lượng các đoạn, do đó làm giảm kích thước của bảng dịch chuyển đoạn.

**A.6 Sự kết hợp**

Nếu nhiều rãnh âm thanh được chứa trong cùng một tệp tin, các rãnh này được pha trộn với nhau để trình diễn. Việc pha trộn này ảnh hưởng từ hai yếu tố: độ lớn các rãnh và sự cân bằng giữa các rãnh.

Tương tự như vậy, các rãnh hình ảnh cũng được kết hợp với nhau dựa số thứ tự lớp (từ sau ra trước) và kiểu kết hợp. Ngoài ra, mỗi rãnh có thể được ánh xạ bằng cách sử dụng một ma trận, vì vậy toàn bộ movie cũng được ánh xạ theo ma trận. Điều này cho phép cả những hoạt động đơn giản (như gấp đôi điểm ảnh, xoay 90 độ) và hoạt động phức tạp (như thay đổi kích cỡ, xoay tùy biến).

Các tiêu chuẩn phái sinh có thể có kết hợp vượt khỏi sự tổng hợp mặc định này của âm thanh và hình ảnh với những hệ thống mạnh hơn (như MPEG-4 BIFS).

**A.7 Truy nhập ngẫu nhiên**

Nội dung phần này miêu tả phương thức tìm kiếm. Việc tìm kiếm được cơ bản thực hiện bằng cách sử dụng các hộp nhỏ trong Hộp Bảng Mẫu. Nếu sử dụng danh sách biên tập thì việc tìm kiếm phải tra cứu danh sách này.

Nếu muốn tìm kiếm một rãnh cho trước theo một thời gian T, trong đó T thuộc danh mục thời gian của Hộp Mào đầu Movie (Hộp Mào đầu Movie), có thể thực hiện theo các bước sau:

1. Nếu rãnh chứa danh sách biên tập, xác định hiệu chỉnh nào có thời gian T bằng cách duyệt qua các hiệu chỉnh. Thời gian bắt đầu chỉnh sửa trong phạm vi thời gian bộ phim sau đó phải được trừ vào thời gian T để tạo ra T ', thời gian vào chỉnh sửa trong phạm vi thời gian bộ phim. T ' tiếp tục chuyển đổi sang phạm vi thời gian của rãnh đa phương tiện để tạo ra T’'. Cuối cùng, thời gian trong phạm vi đa phương tiện được tính bằng cách thêm thời điểm bắt đầu phát đa phương tiện của chỉnh sửa đến T’’.
2. Hộp thời điểm mẫu của một rãnh chỉ ra các thời điểm của các mẫu của rãnh đó. Hộp này được dùng để tìm kiếm mẫu đầu tiên ứng với một thời gian cho trước.
3. Mẫu được đề cập đến trong bước 1 có thể không phải làmẫu đồng bộ. Bảng mẫu đồng bộ chỉ ra các mẫu trên thực tế là các điểm truy nhập ngẫu nhiên. Thông qua bảng này có thể xác định được đâu là mẫu đồng bộ ứng với một thời gian cụ thể. Trường hợp không có bảng mẫu đồng bộ có nghĩa là tất cả các mẫu đều là điểm đồng bộ, rất dễ gây ra lỗi. Truy vấn bảng mẫu đồng bộ để tìm mẫu mong muốn gần nhất, nhưng trước mẫu tìm được trong bước 1.
4. Đến đây ta đã có thể biết cách tìm mẫu truy nhập ngẫu nhiên. Việc sử dụng bảng liên hệ giữa mẫu vào đoạn cho ta biết mẫu mà ta tìm kiếm nằm trong đoạn nào.
5. Để biết đoạn nào chứa mẫu cần tìm ta sử dụng Hộp Độ dịch chuyển Đoạn, thông qua hộp này để biết đoạn đó bắt đầu từ đâu.
6. Bắt đầu từ sự dịch chuyển nêu trên ta có thể sử dụng thông tin trong hộp liên hệ giữa mẫu và đoạn cùng với Hộp Kích thước Mẫu để xác định mẫu nằm ở đâu trong đoạn.

**A.8Các tệp tin movie bị phân mảnh**

Phần này giới thiệu một kỹ thuật có thể được dùng trong các tệp tin ISO, đó là kiến trúc của mộtHộp Movie (Hộp Movie) trong một movie là khá lớn. Nó thường xảy ra ít nhất một trong các trường hợp sau:

* Trong quá trình ghi, tại một thời điểm, nếu việc đang ghi bị hỏng, bị hết dung lượng hoặc có một vài trường hợp xấu xảy ra, sau khi đã ghi được lượng lớn dữ liệu nhưng chưa ghi đượcHộp Movie thì dữ liệu ghi được đó cũng không sử dụng được. Điều này xảy ra là bởi vì định dạng tệp tin được nêu ở đây yêu cầu tất cả các siêu dữ liệu (Hộp Movie) phải được ghi trên phân vùng liên tục của cùng một tệp tin.
* Cũng trong quá trình ghi, đối với các thiết bị nhúng, đặc biệt với các máy ảnh tĩnh, không có RAM để làm bộ đệm choHộp Movie của kích thước bộ nhớ và việc tính toán lại khi hết movie đó là rất chậm. Nguy cơ tương tự cũng xảy ra đối với các trường hợp bị hỏng.
* Trong trường hợp HTTP khởi động nhanh. Nếu phim có kích thước hợp lý,Hộp Movie có thể được tải xuống trong khoảng thời gian không dự định được trước khi việc khởi động nhanh diễn ra.

Tham số cơ bản của phim được thiết lập ở phần đầu củaHộp MovieThuộc đối tượng: số lượng rãnh, các miêu tả về mẫu hiện có, độ rộng, độ cao, liên kết,... Tuy nhiênHộp Movie này không chứa thông tin về độ dài của toàn bộ movie, nhiều khi còn có ít hoặc không có mẫu nào trong các rãnh.

Trong trường hợp có rất ít hoặc chỉ là movie rỗng thì các mẫu khác sẽ được thêm vào, trong cấu trúc tệp tin thì đó cơ bản là các phân mảnh movie.

Triết lý thiết kế cơ bản của phân mảnh movie tương tự nhưHộp Movie, dữ liệu không được đóng khung. Tuy nhiên kiểu thiết kế này cũng được coi như là kiểu thiết kế có đóng khung nếu cần. Các cấu trúc được xây dựng để có thể dùngHộp Movie, do vậy mà một trình diễn kiểu phân mảnh có thể được hiểu như mộtHộp Movie riêng lẻ.

Việc tiếp cận này nghĩa là các giá trị mặc định được thiết lập cho mỗi mẫu, cả toàn cục (một lần cho mỗi rãnh) và trong mỗi phân mảnh. Chỉ những phân mảnh không có các giá trị mặc định thì mới cần các giá trị này. Điều này được áp dụng trong các trường hợp xảy ra thường xuyên, lặp đi lặp lại, cấu trúc nhỏ gọn, không làm mất đi việc xây dựng gia tăng của các movie có sự thay đổi.

Hộp Movie thông thường thiết lập nên cấu trúc của movie. Điều đó có thể thực hiện bất kỳ đâu trong tệp tin, tuy nhiên nếu đặt phía trước các phân mảnh là tốt nhất (Đó không phải là một quy tắc, những thay đổi nhỏ đối vớiHộp Movie mà lại đặt ở cuối tệp tin thì sẽ không thể thực hiện được). Vì vậyHộp Movie cần:

* Phải thể hiện một movie hợp lệ theo đúng nghĩa của nó (cho dù các rãnh không có một mẫu nào).
* Có một hộp để chỉ ra cách thức tìm kiếm và sử dụng các phân mảnh.
* Được dùng để chứa danh sách biên tập đầy đủ (nếu có).

Lưu ý rằng nếu phần mềm không hiểu được các phân mảnh thì sẽ chỉ chạy phần mở đầu của movie. Nếu phần mềm hiểu được các phân mảnh và gặp movie không phân mảnh thì sẽ không xử lý các phân mảnh vì không tìm thấy hộp chỉ thị phân mảnh.

# Phụ lục B (Tham khảo) Tuyên bố về quyền sáng chế

Tổ chức Tiêu chuẩn hóa Quốc tế (ISO) và Ủy ban Kỹ thuật Điện tử Quốc tế (IEC) đã tuyên bố rằng việc tuân thủ phần này của tiêu chuẩn ISO / IEC 14496 và ISO /IEC 15.444 có thể bao gồm việc sử dụng các bằng sáng chế.

ISO và IEC không có trách nhiệm liên quan đến các chứng cứ, tính hiệu lực và phạm vi của các bản quyền sáng chế này.

Chủ sở hữu các bản quyền sáng chế này đã đảm bảo với tổ chức ISO và IEC rằng họ sẵn sàng đàm phán về các giấy phép theo các điều khoản và điều kiện hợp lý, không phân biệt với các ứng viên trên khắp thế giới. Với các điều khoản này, các tuyên bố của những người nắm giữ các bằng sáng chế được đăng ký với ISO và IEC. Thông tin của các công ty nắm bản quyền được liệt kê dưới đây.

|  |  |
| --- | --- |
|  | **Company** |
| 1. | Apple, Inc. |
| 2. | Electronics & Telecommunications Research Institute |
| 3. | Matsushita Electric Industrial Co., Ltd |
| 4. | Nokia Corporation |
| 5. | Nokia Mobile Phones Ltd |
| 6. | QUALCOMM Incorporated |
| 7. | Technical Research Centre of Finland |
| 8. | Telefonaktiebolaget LM Ericsson |

Một số nội dung nằm trong phầncơ bảncủa ISO / IEC 14496 và ISO / IEC 15.444 có thể liên quan đến bản quyền sáng chế chứ không phải nằm trong phần phụ lục. ISO và IEC sẽ không chịu trách nhiệm trong việc xác định phần nào liên quan đến bản quyền sáng chế kiểu như vậy.

ISO ([www.iso.org/patents](file:///C:\Users\Friendly\AppData\Roaming\Microsoft\Downloads\www.iso.org\patents)) và IEC (<http://patents.iec.ch>) duy trì cơ sở dữ liệu trực tuyến của bằng sáng chế liên quan đến tiêu chuẩn của họ. Người dùng được khuyến khích tham khảo cơ sở dữ liệu liên quan đến các bằng sáng chế được cập nhật tương đối thường xuyên.

Lưu ý rằng các tuyên bố về quyền sáng chế được áp dụng cho Định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO (ISO Base Media Format File) được đề cập đến ở đây có thể không áp dụng cho tiêu chuẩn ISO / IEC 15.444-3 (Motion JPEG 2000). ISO / IEC 15.444-3 sử dụng một tập hợp con của ISO / IEC 15.444-12 (ISO Base Media Format File).

# Phụ lục C (Tham khảo) Hướng dẫn tạo định dạng tệp tin dựa trên tiêu chuẩn này

**C.1 Giới thiệu**

Phụ lục này cung cấp thông tin mô tả cách thức tạo một định dạng tệp tin cụ thể từ định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO được đề cập đến ở đây.

Định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO / IEC 14.496-12 | ISO / IEC 15.444-12 xác định cấu trúc cơ bản của các định dạng tệp tin. Các phần đa phương tiện cụ thể và sự mở rộng do người dùng quyết định có thể được cung cấp trong các thông số kỹ thuật khác có nguồn gốc từ định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO.

**C.2 Các nguyên tắc chung**

**C.2.1 Tổng quát**

Hiện có nhiều định dạng tệp tin sử dụng định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO, không chỉ có định dạng tệp tin MP4 MPEG-4 (ISO/IEC 14496-14) và Motion JPEG 2000 MJ2 (ISO / IEC 15.444-3). Khi xem xét một đặc điểm kỹ thuật mới có nguồn gốc từ định dạng tệp tin đa phương tiện cơ sở theo chuẩn ISO, tất cả các thông số kỹ thuật hiện tại nên được sử dụng như là các ví dụ và cũng là một nguồn để đưa ra các định nghĩa và công nghệ. Kiểm tra với đăng ký để tìm cái gì có thể đã tồn tại, và những thông số kỹ thuật tồn tại.

Đặc biệt, nếu một đặc điểm kỹ thuật hiện tại đã bao gồm việc làm thế nào một kiểuđa phương tiện đặc biệt được lưu trữ trong các định dạng tệp tin (ví dụ như hình ảnh MPEG-4 trong MP4), định nghĩa đó nên được sử dụng và không nên phát minh ra cái mới. Bằng cách này thông số kỹ thuật sử dụng chung công nghệ cũng sẽ chia sẻ định nghĩa làm thế nào công nghệ đó được ứng dụng.

Khi thực hiện phân lớp trong phần này, có một số đặc điểm như là ‘parameters’ với đặc điểm được nêu rõ trong trong tài liệu này. Tương tự, có một số đặc điểm của các định dạng tệp tin được chỉ rõ trong trong tài liệu này ở bên trong và được thảo luận thông qua các thông số kỹ thuật khác.

Các thông số kỹ thuật dẫn xuất được viết theo các thông số của các định dạng tệp tintrong tài liệu này; mẫu là gì, nhãn thời gian có nghĩa là gì, v.v. Đề cập đến các hộp cụ thể có trong mỗi đặc điểm kỹ thuật dẫn xuất thường có thể tạo ra lỗi, trừ một số trường hợp hạn chế (ví dụ như thêm một hộp dữ liệu người dùng, hoặc một hộp mở rộng).

**C.2.2 Hoạt động lớp cơ sở**

Có thể thực hiện một số hoạt động trên một tệp tintrong tài liệu này mà không cần quan tâm đến bất kỳ thông số kỹ thuật phái sinh nào. Các hoạt động này có thể bao gồm các việc đọc các rãnh, tìm kiếm dữ liệu, thời gian lấy mẫu và miêu tả mẫu, v.v. Ví dụ, điều này có thể được thực hiện, bằng một định dạng tệp tin inspector hoặc thư viện chung như các phần mềm tham chiếu.

Ít rõ ràng hơn là một lớp các thao tác của các tệp tin:

1. Đan xen lại dữ liệu; sắp xếp dữ liệu đa phương tiện theo thứ tự thời gian, với các mẫu cho các rãnh khác nhau nhóm lại thành khối có kích thước hợp lý, với các khối xen kẽ;
2. Làm cho các tệp tin sử dụng các tham chiếu dữ liệu trở nên độc lập bằng cách sao chép dữ liệu từ tệp tin bên ngoài vào tệp tin mới;
3. Kiểu bỏ các nguyên tử không gian miễn phí và nén các cấu trúc nguyên tử;
4. Kiểu bỏ dữ liệu từ nguyên tử '*mdat*' xuất hiện để được bỏ tham chiếu bởi các rãnh hoặc các nguyên tử siêu dữ liệu;
5. Kiểu bỏ các giá trị tính mẫu mà không có mẫu liên kết;
6. Kiểu bỏ các nhóm mẫu mà không có mẫu liên kết;
7. Chiết xuất một số rãnh và tạo một tệp tin mới với chỉ những cái kia (ví dụ như một rãnh âm thanh từ một biểu diễn âm thanh / hình ảnh);
8. Thêm hoặc kiểu bỏ các phân mảnh movie hoặc phân mảnh lại một movie.

Danh sách trên chưa hoàn toàn đầy đủ.

**C.3 Các hộp**

Có thể thêm các hộp vào định dạng tệp tin, nhưng cần lưu ývì nó có thể ảnh hưởng đếncác hộp khác. Đặc biệt, nếu có sự liên kết chéo (cross-link) giữa các hộp đang tồn tại thì khi thêm vào các tệp tin này có thể không tương thích với tiêu chuẩn trong tài liệu này.

Bạn phải đăng ký tất cả các hộp mới, ngoại trừ những người sử dụng kiểu “uuid. Tương tự như vậy, bạn nên đăng ký giải mã tên (nhập giá trị tính mẫu), các nhãn, các kiểu rãnh tham chiếu, xử lý (các kiểuđa phương tiện), các nhóm, và các kiểu sơ đồ bảo vệ. Khi sử dụng một trong những kiểu trên mà không đăng ký thì đó là một ý tưởng tồi, vì va chạm có thể xảy ra - hoặc người khác có thể đăng ký cùng một định danh với nghĩa khác nhau.

Bạn không nên viết một hộp bằng cách sử dụng ' UUID escape' nếu một mã bốn ký tự đơn giản có thể được sử dụng (các mô hình dành riêng ISO UUID 0xXXXXXXXX- 0011- 0010-8000-00AA00389B71, nơi mà các mã bốn kí tự thay thế X’s) và bạn không nên thiết kế để sử dụng một hộp UUID; tốt hơn là nên đặt dữ liệu của bạn ở ‘expansion points ‘ đã biết của các định dạng tệp tin nếu có thể, hoặc đăng ký một Kiểu hộp mới nếu thực sự cần thiết.

Tất cả dữ liệu trong các tệp tin ISO phải được, hoặc được chứa trong các hộp. Bạn có thể giới thiệu một chữ ký, nhưng nó phải 'giống như' một hộp.

Không yêu cầu bất kỳ hộp hiện tại hoặc mới, bạn xác định được ở một vị trí cụ thể, nếu có thể. Ví dụ, thông số kỹ thuật JPEG 2000 hiện tại đòi hỏi một ô chữ ký và nó là cái đầu tiên trong tệp tin. Nếu đặc điểm kỹ thuật khác cũng định nghĩa một ô chữ ký và cũng đòi hỏi rằng đó là cái đầu tiên, sau đó một tệp tin tuân thủ QTI cho cả hai thông số kỹ thuật không thể được xây dựng.

Nó phải có khả năng ‘walk’ ở mức cao nhất của một tệp tin bằng cách tìm độ dàihộp.Đừng quên rằng 'implied length’ được cho phép ở mứctệp tin.

Nếu không hoàn toàn không thể tránh khỏi, hộp nên chứa hoặc dữ liệu (ví dụ như trong các miền), hoặc các hộp khác, nhưng không phải cả hai. Tất cả các hộp có chứa dữ liệu cần là một hộp đầy đủ để cho phép các thay đổi sau đó với cú pháp và ngữ nghĩa. Hộp có chứa các hộp khác được gọi là các hộp chứa, và thường là một hộp đơn giản (không đầy đủ), vì ngữ nghĩa của nó sẽ không bao giờ thay đổi nếu nó được cung cấp tư liệu để chỉ chứa hộp.

**C.4 Nhãn định danh**

**C.4.1 Giới thiệu**

Phần này bao gồm việc sử dụng bộ nhận dạng nhãntrong Hộp Kiểu Tệp tin, bao gồm:

* Giới thiệu một nhãnmới.
* Hành vi của các thiết bị trình diễn phụ thuộc vào nhãn.
* Thiết lập nhãn vào việc tạo ra các tệp tinđa phương tiện cơ sở theo chuẩn ISO.

Nhãn xác định một đặc điểm kỹ thuật và thực hiện một bộ báo cáo đơn giản:

1. Tệp tin phù hợp với mọi yêu cầu của các đặc điểm kỹ thuật xác định;
2. Tệp tin không chứa gì trái ngược với các đặc điểm kỹ thuật xác định;
3. Một đầu đọc bổ sung mà đặc điểm kỹ thuật đơn có thể đọc, giải thích, và có thể biểu diễn các tệp tin, bỏ qua dữ liệu nó không nhận ra.

**C.4.2 Cách sử dụng của các nhãn**

Để xác định các thông số kỹ thuật mà các tệp tin tuân thủ, nhãn được sử dụng như các định dạng trong các định dạng tệp tin. Các nhãn này được thiết lập để thu Hộp Kiểu Tệp tin.

Ví dụ, một nhãn có thể cho thấy:

1. Các bộ mã hóa/giải mã có thể xuất hiện trong tệp tin,
2. Làm thế nào dữ liệu của từng codec được lưu trữ,
3. Hạn chế và mở rộng được áp dụng cho các tệp tin.

Nhãn mới có thể được đăng ký nếu cần thiết để tạo ra một đặc điểm kỹ thuật mới mà không hoàn toàn tuân thủ tiêu chuẩn hiện hành. Ví dụ, 3GPP cho phép sử dụng AMR và H.263 trong các định dạng tệp tin. Vì các codec không được hỗ trợ trong bất kỳ tiêu chuẩn tại thời điểm đó, 3GPP quy định việc sử dụng đầu vào mục mẫu và các trường khuôn mẫu trong định dạng tệp tin đa phươn tiện cơ sở theo chuẩn ISO cũng như xác định hộp mới mà các codec đề cập đến. Cân nhắc rằng định dạng tệp tin được sử dụng rộng rãi hơn trong tương lai và sẽ cần nhiều nhãn hơn.

Các nhãn không được thêm vào; chúng tồn tại một mình. Bạn không thể nói:”Nhãn này chỉ ra rằng hỗ trợ cho Y cũng được yêu cầu “bởi vì từ 'cũng' không có địa chỉ tham chiếu.

Hệ thống có các tệp tinghi lại cần kiểu bỏ các nhãn mà nó hệ thống không nhận diện được, vì hệ thống không biết liệu tệp tin vẫn còn phù hợp với các yêu cầu đó của nhãn hay không(ví dụ như chèn lại một tệp tin có thể làm mất đi sự tương hợp với một đặc điểm kỹ thuật đòi hỏi một kiểu chèn nhất định).

Lưu ý rằng nhãn lớn thường có phần mở rộng tệp tin, mà lần lượt kéo theo kiểu MIME. Nhưng đây không phải là những quy định. Ngoài ra, khi phục vụ dưới một định dạng MIME đừng quên rằng các kiểu MIME có thể lấy các thông số, và danh sách các nhãn tương thích thường sẽ có ích cho hệ thống tiếp nhận.

**C.4.3 Giới thiệu một nhãn mới**

Một nhãn mới có thể được xác định nếu phải chỉ ra sự tương thích với các đặc điểm kĩ thuật mới. Điều này thường có nghĩa là định nghĩa của một nhãn mới ít nhất một trong các điều kiện sau đây cần được thỏa mãn:

1. Sử dụng một codec mà không được hỗ trợ trong bất kỳ nhãn nào hiện có.
2. Sử dụng nhiều hơn một codec trong một hệ thống không được hỗ trợ trong bất kỳ nhãnnào hiện có. Ngoài ra, bộ phận phát lại của tệp tin chỉ được phép khi giải mã của tất cả các đa phương tiện trong các tệp tin được hỗ trợ bởi các thiết bị trình diễn.
3. Sử dụng phần hạn chế / hoặc phần mở rộng là người dung cụ thể (hộp, các miền mẫu, v.v)

Tuy nhiên, định dạng tệp tin có chứa cả một miền *major\_brand* và một dãy các nhãn tương thích. Các lĩnh vực này thuộc sở hữu của tác giả tệp tin và đặc điểm kỹ thuật trong tài liệu này. Đừng viết một đặc điểm kỹ thuật nói về các miền này, chỉ đơn thuần là về nhãn và Ngữ nghĩa của chúng. Đặc biệt, không yêu cầu các miền *major\_brand* ("tệp tin tương hợp với đặc điểm kỹ thuật này phải thiết lập *major\_brand* về XXXX“) như là một tệp tin không bao giờ tương thích với hai thông số kỹ thuật viết theo cách đó, và bạn cũng chặn một người nào đó lấy đi một đặc điểm kỹ thuật của bạn. Tuy nhiên, nhãn chỉ được cho phép như các nhãn tương thích có thể được xác định. Nhưng nhãn có thể được sử dụng như một đánh dấu. Đó là điều hoàn toàn hợp pháp để có một nhãn mà không có yêu cầu, và được đặt trong một tệp tin như là “I was there' (hoặc “nhãn này yêu cầu các tệp tin đã được viết cuối cùng bởi ZZZZ “).

**C.4.4 Hướng dẫn cho thiết bị trình diễn**

Nếu có nhiều hơn một nhãn có mặt trong danh sách các nhãn tương thích, và một hoặc nhiều nhãn được hỗ trợ bởi thiết bị đọc, thiết bị đọc sẽ chạy các dạng của tệp tin đó phù hợp với những thông số kỹ thuật. Trong trường hợp này, thiết bị đọc có thể không có khả năng giải mã đa phương tiện không được hỗ trợ.

**C.4.5 Hướng dẫn thiết kế**

Nếu tác giả muốn tạo ra một tệp tin phù hợp với nhiều hơn một đặc điểm kỹ thuật, áp dụng những yếu tố sau:

1. Không có gì trái ngược với các đặc điểm kỹ thuật xác định bởi một nhãn trong tệp tin. Ví dụ, nếu một đặc điểm kỹ thuật yêu cầu các tệp tin được khép kín, sau đó dấu hiệu cho thấy nhãncủa đặc điểm kỹ thuật không được sử dụng trên các tệp tin không khép kín.
2. Nếu tác giả cho rằng một thiết bị đọc phù hợp với chỉ một trong các thông số kỹ thuật, chỉ chạy phương tiện phù hợp với đặc điểm kỹ thuật đó, sau đó nhãn đó có thể được chỉ định.
3. Nếu tác giả yêu cầu phát các đa phương tiện từ nhiều hơn một đặc điểm kỹ thuật, sau đó sẽ có thể cần một nhãn mới vì điều này đại diện cho một yêu cầu mới cho thiết bị đọc.

**C.4.6 Ví dụ**

Trong phần này, chúng ta lấy ví dụ trường hợp khi một nhãnmới có thể được xác định. Trước hết, chúng tôi giải thích về hai nhãn hiện có. Nếu nhãn '3gp5' trong danh sách các các nhãn tương thích, nó chỉ ra rằng các tệp tin có chứa các đa phương tiện được định nghĩa trong 3GPP TS 26,234 (Release 5) theo cách tiêu chuẩn quy định. Ví dụ, các tệp tin của nhãn '3gp5' có thể bao gồm H.263. Tương tự như vậy, nếu nhãn 'mp42' trong danh sách của các nhãn tương thích, nó cho thấy rằng các tệp tin có chứa các đa phương tiện được định nghĩa trong ISO / IEC 14.496-14 một cách cụ thể. Ví dụ, các tệp tin của nhãn 'mp42' có thể bao gồm MP3. Tuy nhiên, MP3 không được hỗ trợ trong nhãn'3gp5'.

Các dẫn chứng đã chỉ ra các tệp tin có chứa H.263 và MP3, và có '3gp5' và 'mp42' là các nhãn tương thích. Nếu thiết bị đọc chỉ tuân thủ '3gp5' và không hỗ trợ MP3, thiết bị đọc chỉ nên phát H.263. Nếu tác giả của nội dung không mong đợi hoạt động như vậy, một nhãn mới được định nghĩa để cho thấy rằng cả H.263 và MP3 được hỗ trợ trong các tệp tin. Bằng cách xác định nhãn mới được xác định trong danh sách các các nhãn tương thích, nó có thể ngăn chặn các hành vi trên và tệp tin chỉ được phát khi thiết bị đọc hỗ trợ cả H.263 và MP3.

**C.5 Việc lưu các kiểuđa phương tiện mới**

Có hai lựa chọn trong định nghĩa làm thế nào một kiểuđa phương tiện mới sẽ được lưu trữ:

Đầu tiên, nếu cấu trúc hệ thống MPEG-4 được mong muốn hoặc được chấp nhận được, thì sau đó:

a) Một ObjectTypeIndication mới sẽ được yêu cầu và sử dụng;

b) Decoderspecificinformation cho codec này cần được xác định như là một bộ miêu tả MPEG-4;

c)Định dạng đơn vị truy nhập cần được xác định cho đa phương tiện này.

Các đa phương tiện sau đó sử dụng MPEG-4 mã điểm trong các định dạng tệp tin; Ví dụ, một codechình ảnh mới sẽ sử dụng một đầu vào mục mẫu của kiểu 'MP4V.

Nếu hệ thống lớp MPEG-4 là không phù hợp hoặc không mong muốn, sau đó:

a) Một mã bốn kí tự *sampleentry* mới sẽ được yêu cầu và sử dụng;

b) Bất kỳ thông tin bổ sung cần thiết cho các bộ giải mã cần được xác định như hộp phải được lưu trữ trong các *sampleentry*;

c) Các định dạng mẫu định dạng tệp tin phải được xác định cho các đa phương tiện này.

Lưu ý rằng trong trường hợp thứ hai, cơ quan đăng ký cũng sẽ phân bổ một objecttypeindication để sử dụng trong hệ thống MPEG-4

**C.6 Sử dụng các trường mẫu**

Các trường khuôn mẫu được quy định trong các định dạng tệp tin. Nếu bất kì cái nào được sử dụng trong một đặc điểm kỹ thuật dẫn xuất, việc sử dụng phải tương thích với các định nghĩa cơ bản, và sử dụng tài liệu đó một cách rõ ràng.

**C.7 Rãnh**

**C.7.1 Đinh vị dữ liệu**

Một rãnh là một chuỗi theo thời gian của các mẫu; mỗi mẫu được xác định bởi dữ liệu, độ dài và vị trí của nó (các byte chứa trong nó). Độ dài và dữ liệu của một mẫu là các thông số bên ngoài để định dạng tệp tin; vị trí của byte thì không. Cách cơ bản xác để các dữ liệu được lưu trữ là bên trong của định dạng tệp tintrong tài liệu này. Khi xác định định dạng của bạn thuộc mẫu gì, bạn nên xác định độ dài và dữ liệu của một mẫu.

Tuy nhiên, bạn không nên đề cập đến các hộp sau, vì cách mà chúng được cấu trúc là mở để thay đổi, và các thông tin mà nó lưu trữ có thể được lưu trữ trong những cách khác (ví dụ như thông tin kích thước mẫu có thể ở trong một hộp *stsz*, một hộp *stz2*, hoặc một phân mảnh movie):

Kích thước mẫu (*stsz*), kích thức mẫu rút gọn (*stz2*).

Các mẫu trên thực tế được lưu trữ trong các dải tiếp giáp của mẫu cho một rãnh; những dải này được gọi là khúc dữ liệu, và nó là khúc dữ liệu từ các rãnh khác nhau được xen kẽ. Tuy nhiên, tệp tin có thể được xen kẽ lại hoặc phân đoạn lại; các hộp sau nói về việc phân đoạn được thực hiện như thế nào:

Các độ dịch chuyển đoạn (*stco* hoặc*co64*), ánh xạ mẫu vào đoạn (*stsc*).

Quan trọng nhất, định vị dữ liệu trong một tệp tintrong tài liệu này phải được thực hiện thông qua các hộp này (hoặc tương đương với nó trong phân mảnh movie). Hộp Dữ liệu Đa phương tiện ('*mdat*') chỉ là một vị trí có thể hoặc có thể là cơ bản nó, nó chỉ có thể được coi là một túi không được sắp xếp của các bit không được nhận dạng. Không có gì đảm bảo rằng các tài liệu mong muốn trong một hộp dữ liệu đa phương tiện là dữ liệu duy nhất trong hộp hoặc trong bất kỳ thứ tự cụ thể nào, và đặc biệt là nếu tài liệu tham chiếu dữ liệu được sử dụng, không có gì đảm bảo rằng bất kỳ mẫu đặc biệt nào ngay cả trong một hộp dữ liệu đa phương tiện. Đề cập đến dữ liệu đa phươn tiện ('*mdat*') trong một hộp đặc điểm kỹ thuật dẫn xuất gần như chắc chắn là một sai lầm, và cố gắng xác định (hoặc giả sử) cấu trúc của nó là chiếm đoạt các đặc điểm kỹ thuật trong tài liệu này, và là một lỗi.

Để yêu cầu một kiểu, thời gian, hoặc kích thước nhất định của việc đan xen trong một đặc điểm kĩ thuật kết hợp là hoàn toàn chấp nhận được ("đặc điểm kỹ thuật này yêu cầu các tệp tin được khép kín, và các dữ liệu đa phương tiện được giải mã theo thứ tự thời gian, xen kẽ vào tốc độ cơ bản không lớn hơn một giây “).

**C.7.2 Thời gian**

Tương tự như vậy, các mẫu được biểu hiện bằng tham số trong thời gian trong định dạng tệp tin bởi dấu thời gian giải mã, và tùy chọn bởi dấu thời gian tổng hợp của nó. Bạn nên xác định những cái này có Ngữ nghĩa gì với đa phương tiện của bạn. Tuy nhiên, cách mà các cái này được lưu trữ lại là ở bên trong của định dạng tệp tintrong tài liệu này. Tuy nhiên bạn không nên đề cập đến các hộp sau, vì cách mà chúng được cấu trúc mở để thay đổi, và các thông tin mà chúng lưu trữ có thể được lưu trữ theo những cách khác:

Hộp ánh xạ thời gian và mẫu (*stts*), hộp độ dịch chuyển tổng hợp (*ctts*).

Tương tự như vậy, hiệu ứng time-structure nên được bảo vệ bằng các định dạng tệp tin, nhưng có một người làm đơn giản tệp tintrong tài liệu này có thể hợp nhất hai sửa đổi liền kề rằng trong thực tế có liên kết với nhau (ví dụ như hai chỉnh sửa trống, hoặc một chỉnh sửa mà chọn thời gian A-B theo sau là một lựa chọn B-C).

**C.7.3 Các kiểuđa phương tiện**

Có một số kiểuđa phương tiện trong các đặc điểm kỹ thuậttrong tài liệu này: hình ảnh, âm thanh, siêu dữ liệu,... Những thứ này được đại diện bởi các kiểu xử lý rãnh và các mào đầu đa phương tiện của media-specific. Có thể đăng ký bộ xử lí đa phương tiện mới, nhưng điều này hiếm khi được yêu cầu. Việc đăng ký cũng cần được kiểm tra; bộ xử lý cần thiết có thể đã được định nghĩa trong một đặc điểm kỹ thuật dẫn xuất.

**C.7.4 Các kiểu mã hóa**

Tên của một đầu vào mục mẫu xác định định dạng mã hóa được sử dụng. Đây là một trong những cách cơ bản mà đặc điểm kỹ thuật trong tài liệu này được biểu hiện bằng tham số; ví dụ như AVC (MPEG-4 Part 10) sử dụng 'avc1' như một kiểu đầu vào mục mẫu. Xác định tên này cho một codec, và đăng ký nó, và sau đó xác định những hộp thêm vào là một giá trị tính mẫu cho codec này, là các cách cơ bản mà các định dạng trong tài liệu này được sử dụng. Bạn nên xác định những thứ này cho hệ thống mã hóa của bạn. Lưu ý rằng về mặt kỹ thuật các kiểu mã hóa được 'scoped' bởi các kiểuđa phương tiện (mặc dù chúng tôi không cố gắng để xác định cùng mã bốn ký tự như hai codec khác nhau trong hai kiểuđa phương tiện, chẳng hạn như hình ảnh và âm thanh, để tránh nhầm lẫn).

**C.7.5 Thông tin mẫu con**

Các đặc điểm kỹ thuật trong tài liệu này có thể mang thông tin về ranh giới “mẫu con” cho từng mẫu. Tuy nhiên, định nghĩa mẫu con là gì thì cụ thể cho một hệ thống mã hóa. Bạn có thể muốn định nghĩa nó khi xác định làm thế nào một hệ thống mã hóa được lưu trữ.

**C.7.6 Sự phụ thuộc của mẫu**

Các định dạng trong tài liệu này cho phép bạn xác định một số thông tin giải mã phụ thuộc cho một hệ thống mã hóa. Đặc biệt, bạn nên xác định thế nào là một sync hợp lệ hoặc điểm truy nhập ngẫu nhiên (điểm mà từ đó giải mã có thể được bắt đầu). Chúng có thể được đánh dấu trong các định dạng tệp tin (trong bảng mẫu đồng bộ, hoặc dán cờ trong mảnh movie). Nên ít quan tâm đến việc làm thế nào để đồng bộ mẫu được đánh dấu.

Tương tự như vậy, nó có thể chỉ ra các mẫu:

a) Phụ thuộc vào mẫu khác, hoặc có thể được giải mã một cách độc lập;

b) Bị phụ thuộc vào mẫu khác, hoặc có thể được kiểu bỏ mà không ảnh hưởng đến giải mã;

c) Chứa nhiều mã hóa của các thông tin giống nhau, có thể với sự phụ thuộc khác (là mã hóa dư thừa).

Đối với hầu hết các hệ thống mã hóa, nghĩa của những cái này là hiển nhiên và không cần giải thích rõ ràng; Tuy nhiên, họ có thể cần tuyên bố rõ ràng đối với một số hệ thống mã hóa.

**C.7.7 Các nhóm mẫu**

Các nhóm mẫu cung cấp một cách khác để miêu tả mẫu và đặc điểm của chúng. Để sử dụng các nhóm mẫu, bạn có thể xác định một kiểu nhóm, và sau đó làm thế nào một nhóm được xác định (miêu tả nhóm). Các định dạng tệp tin sau đó có thể sắp đặt một mẫu đã cho với một định nghĩa duy nhất của một nhóm của một kiểu bất kì đã cho. Xác định các kiểu nhóm mới và cách mà chúng được biểu hiện bằng tham số là một cách quan trọng để biểu hiện bằng tham số định dạng tệp tin.

**C.7.8 Mức rãnh**

Các rãnh có thể được liên kết với nhau trong định dạng tệp tin này theo hai cách quan trọng.Các tham chiếu rãnh là một liên kết cho thấy sự tham chiếu hoặc sự phụ thuộc của một rãnhvàomột rãnh khác (ví dụ như rãnh siêu dữ liệu miêu tả một rãnh đa phương tiệnphụ thuộc vào rãnh đa phương tiệnđó, vì không có nó thì không có nghĩa). Kiểu tham chiếu rãnh mới có thể được đăng ký và được sử dụng trong thông số kỹ thuật dẫn xuất. Tương tự các rãnh có thể được nhóm lại thành bộ lựa chọn thay thế, trong đó thiết bị độcđược mong đợi có thể chọn được một rãnh phù hợp với nó (ví dụ trên cơ sở các codec được hỗ trợ, tốc độ bit, kích cỡ màn hình, v.v). 3GPP 26,234 đã đưa ra khái niệm này và bao gồm dữ liệu người dùng (một phần mở rộng cho phép) để đưa ra một gợi ý tại sao một rãnh là bộ phận của một nhóm ("Tôi có một codec khác').

Cuối cùng, các rãnh có thể được kích hoạt hay vô hiệu hóa trong các định dạng tệp tin. Các rãnh bị vô hiệu hóa có thể được sử dụng, ví dụ, đối với tính năng tùy chọn (ví dụ như ghi chú đóng).

**C.7.9 Bảo vệ**

Tương tự như việc tham số hóa của các lược đồ mã hóa bằng cách sử dụng kiểu đầu vào mục mẫu và các hộp thêm vào trong đầu vào mục mẫu, định dạng được nêu trong trong tài liệu này cho phép bảo vệ các rãnh, tham số hóa bởi kiểu lược đồ và nội dung của hộp thông tin lược đồ. Hộp thông tin lược đồ do kiểu lược đồ “sở hữu “, mở rộng để chứa các hộp không cần phải đăng ký do đã nằm trong phạm vi của kiểu lược đồ.

Việc bảo vệ có thể khá tinh tế; ví dụ như có nhiều hệ thống mã hóa được liên kết với nhau. Điều đó dùng đề mã hóa 'các nội dung của hộp*mdat* ', nhưng nó cũng có tác động không tốt đến những thay đổi nhỏ của tệp tin. Nó cũng có thể bảo vệ các đoạn, các đoạn này thể hiện hoạt động liên tục của dữ liệu đa phương tiện của một rãnh. Nhưng nếu thực hiện việc phân đoạn lại tệp tin thì có khả năng mất an toàn.

Thay vào đó, hãy xem xét sửa đổi mẫu, hoặc giới thiệu siêu dữ liệu thời gian song song, hoặc sử dụng các nhóm mẫu, để giới thiệu đủ bối cảnh để cho phép cả giải mã và thủ thuật dựa trên tệp tin. Siêu dữ liệu thời gian song song sẽ có thể ở trong một rãnh, và một rãnh tham chiếu nên được sử dụng để chỉ ra rằng bảo vệ dữ liệu phụ thuộc vào rãnh ngữ cảnh mã hóa song song.

**C.8 Cấu trúc của movie phân mảnh**

Khi xây dựng một tệp tin bị phân mảnh để phát lại, có một số khuyến nghị cho các cơ cấu nội dung mà sẽ tối ưu hóa bộ phận phát lại và truy nhập ngẫu nhiên. Các khuyến nghị như sau:

* Các tệp tin nên bao gồm các hộp được sắp xếp theo thứ tự sau:
  + '*ftyp*';
  + '*moov*';
  + Cặp '*moof*' và '*mdat*' (số lượng tùy ý);
  + '*mfra*'.
* Một hộp '*moof*' bao gồm ít nhất một '*traf*' cho mỗi đa phương tiện. Khi các tệp tin có chứa một rãnh hình ảnhduy nhất và một rãnh âm thanhduy nhất, '*moof*' sẽ có hai '*traf*', một cho hình ảnh và một cho âm thanh.
* Đối với hình ảnh, các mẫu có thể truy nhập ngẫu nhiên được lưu trữ như là mẫu đầu tiên của mỗi '*traf*. Trong trường hợp làm mới bộ giải mã tuần tự, một mẫu có thể truy nhập ngẫu nhiên và các điểm phục hồi tương ứng được lưu trữ trong cùng một phân mảnh movie. Đối với âm thanh, các mẫu có thời gian biểu hiện gần nhất cho mỗi mẫu hình ảnh có thể truy nhập ngẫu nhiên được lưu trữ như các mẫu đầu tiên của mỗi '*traf*. Do đó, các mẫu đầu tiên của mỗi đa phương tiện trong ‘*moof*’ có thời gian trình bày xấp xỉ bằng nhau.
* Mẫu đầu tiên (có thể truy nhập ngẫu nhiên) được ghi lại trong '*mfra*' cho cả hình ảnh và âm thanh.
* Tất cả các mẫu trong '*mdat*' được xen kẽ với độ sâu xen kẽ thích hợp.

Độ dịch chuyển và thời gian trình bày ban đầu của tất cả các ‘*moof*'được đưa ra trong'*mfra*'cho cả âm thanh và hình ảnh.

Thiết bị đọc sẽ tải '*moov*' và '*mfra*' vào lúc đầu, và giữ chúng trong bộ nhớ trong quá trình phát lại. khi truy nhập ngẫu nhiên là cần thiết, thiết bị đọc sẽ tìm kiếm '*mfra*' để tìm điểm truy nhập ngẫu nhiên có thời gian trình bày gần nhất với thời gian chỉ định.

Vì mẫu đầu tiên trong '*moof*' có thể truy nhập ngẫu nhiên truy nhập, thiết bị đọc có thể chuyển đến điểm truy nhập ngẫu nhiên. Thiết bị đọc có thể đọc '*moof*' của các điểm truy nhập ngẫu nhiên từ đầu. Các '*mdat*' tiếp theo bắt đầu từ các mẫu truy nhập ngẫu nhiên. Như vậy, một tìm kiếm hai bước sẽ không cần thiết cho truy nhập ngẫu nhiên.

Lưu ý rằng một hộp '*mfra*' là tùy chọn và không bao giờ có thể xảy ra trong một tệp tin nhất định.

**C.9 Siêu dữ liệu**

Phần lớn những gì được nói ở trên về các rãnh và dữ liệu của chúng được áp dụng cho các mục tin siêu dữ liệu, mục tin siêu dữ liệu không có cấu trúc thời gian. Đặc biệt, Sự phân chia của các mục tin thành các mức độ - cho phép chúng được xen kẽ. Nó sẽ là một sai lầm khi thiết kế một số hỗ trợ mới dựa trên cơ cấu mở rộng.

**C.10 Đăng ký**

Đăng ký! Nếu có gì không rõ thì cần liên hệ với cơ quan đăng ký tại địa chỉ<http://www.mp4ra.org>. Việc đăng ký, hỗ trợ và khuyến nghị đều. Nếu không đăng ký có nghĩa là nhãncủa bạn có thể xung đột với người khác, cũng không thể liên hệ và do đó không được cung cấp tài liệu một cách hiệu quả. RA phát hiện được nhiều nhãn đã được phát minh và sử dụng, nhưng không đăng ký. Những người tạo ra các nhãn đó có thể tạo nên những sản phẩm không an toàn do vậy không nên sử dụng.

**C.11 Hướng dẫn về việc sử dụng các nhóm mẫu, các rãnh siêu dữ liệu theo thời gian, và thông tin mẫu bổ trợ**

Định dạng tiệp tin đa phương tiện cơ sở theo chuẩn ISO có ba cơ chế cho siêu dữ liệu theo thời gian có thể được liên kết với mẫu cụ thể: nhóm mẫu, các rãnh siêu dữ liệu theo thời gian, và các thông tin mẫu phụ trợ. Đặc điểm kỹ thuật dẫn xuất có thể cung cấp chức năng tương tự với một hoặc nhiều trong ba cơ chế này. Mục này cung cấp hướng dẫn cho các chi tiết kỹ thuật dẫn xuất để lựa chọn giữa ba cơ chế.

Các nhóm mẫu và siêu dữ liệu theo thời gian được kết hợp ít chặt chẽ hơn với các dữ liệu đa phương tiện và thường 'miêu tả', trong khi đó thông tin mẫu phụ trợ có thể được yêu cầu để giải mã.

Thông tin mẫu phụ trợ chỉ được dùng để sử dụng khi các thông tin có liên quan trực tiếp đến mẫu trên cơ sở một-một, và là cần thiết cho việc xử lý mẫu đa phương tiện và trình bày. Đối với nội dung tổng quát, các giải pháp hiện có của các tracks thêm vào nên được sử dụng. Cả thông tin mẫu phụ trợ và dữ liệu đa phương tiện mẫu đều được giải quyết bằng cách sử dụng con trỏ byte và thông tin kích thước, và do đó, khi các byte cùng tạo thành dữ liệu cho nhiều hơn một mẫu ta có thể chia sẻ dữ liệu đó bằng cách tái sử dụng con trỏ cùng một byte.

Các nhóm mẫu có thể hữu ích trong trường hợp sau đây.

* Khi có nhiều mẫu chia sẻ cùng các giá trị siêu dữ liệu, nó là không gian hiệu quả để xác định siêu dữ liệu trong một Hộp Mô tả Nhóm Mẫu và sự kết hợp của các mẫu cho siêu dữ liệu trong mẫu của Hộp Nhóm.
* Khi thông tin nhóm mẫu được lưu trữ trongHộp Movie và Hộp Phân mảnh Movie, nó cung cấp một chỉ số cho dữ liệu trong Hộp Dữ liệu Đa phương tiện. Không có dữ liệu từ các hộp truyền thông dữ liệu cần phải được lấy, mà do đó có thể làm giảm lượng truy nhập đĩa khi so sánh với rãnh siêu dữ liệu theo thời gian và thông tin mẫu phụ trợ.

Rãnh siêu dữ liệu theo thời gian có thể hữu ích trong trường hợp sau đây:

* Rãnh siêu dữ liệu theo thời gian giống nhau có thể liên quan đến nhiều hơn một rãnh. Nói cách khác, một rãnh siêu dữ liệu theo thời gian có thể độc lập hơn về nội dung của các rãnh liên quan so với các nhóm mẫu và thông tin mẫu phụ trợ.
* Để nối thêm một tệp tin với một rãnh siêu dữ liệu theo thời gian thì dễ dàng hơn so với thông tin phụ mẫu hoặc nhóm mẫu, bởi vì thông tin mẫu phụ trợ và mẫu của các Hộp Nhóm phải ở tại cùng một hộp phân mảnh rãnh như các mẫu liên quan, trong khi siêu dữ liệu theo thời gian có thể ở tại Hộp Phân mảnh Movie của nó. Ví dụ, để cung cấp một phụ đề bổ sung như siêu dữ liệu theo thời gian dễ dàng hơn so với sử dụng thông tin mẫu phụ trợ.
* Thời hạn của các mẫu siêu dữ liệu theo thời gian không cần phải phù hợp với thời gian liên quan đến đa phương tiện hoặc mẫu gợi ý. Trong trường hợp thời gian của mẫu siêu dữ liệu theo thời gian kéo dài trên nhiều đa phương tiện hoặc gợi ý mẫu liên quan, các rãnh siêu dữ liệu theo thời gian có thể có nhiều không gian hiệu quả hơn so với thông tin mẫu phụ trợ.

Thông tin mẫu phụ trợ có thể hữu ích trong các trường hợp sau đây:

* Các dữ liệu liên quan đến mẫu đang thay đổi đầy đủ thường xuyên. mà các nhóm mẫu cụ thể có thể không được ủng hộ từ quan điểm không gian lưu trữ.
* Số lượng dữ liệu liên quan đến mẫu lớn đén nỗi mà việc vận chuyển nó trong Hộp Moviehoặc Hộp Phân mảnh Movie(theo yêu cầu của mẫu nhóm) có thể gây bất lợi. Vídụ, trong tải về tiến bộ, làm nhỏ kích thước củaHộp Movie để giữ thời gian đệm ban đầu nhỏ.
* Khi mỗi mẫu được kết hợp với siêu dữ liệu, thông tin mẫu phụ trợ cung cấp nhiều kết hợp đơn giản hơn của các thông tin phụ trợ cho mẫu khi so sánh với cùng chức năng của các rãnh siêu dữ liệu theo thời gian, thường đòi hỏi phải giải quyết thời gian giải mã mẫu để thiết lập mối liên hệ giữa các mẫu siêu dữ liệu theo thời gian và các mẫu đa phương tiện hoặc mẫu gợi ý.

# Phụ lục D (Tham khảo) Tổ chức chứng nhận

**D.1. Các điểm mã được đăng kí**

Các mã điểm trong các định dạng tệp tin là các trường 32 bit, thông thường là bốn kí tự in được (được biết đến như mã bốn kí tự hoặc 4CCs). Một định danh ObjectType có kiểu số nguyên 8 bit.

Các điểm mã có thể được đăng kí là:

1. Hộptệp tin định dạng định danh. Lưu ý rằng trong một số chi tiết kĩ thuật, các hộp được gọi là nguyên tử. Lưu ý tiếp theo rằng sự ra đời cả các kiểu nguyên tử mới thì được khuyến khích, nói chung các tính năng mở rộng khác của các định dạng tệp tin này nên được sử dụng nếu có thể.
2. Định dạngtệp tinkiểu theo dõi định danh. Một cặp định danh thường được sử dụng để xác định kiểu theo dõi (âm thanh, hình ảnh,..) và một tiêu đề nguyên tử truyền thông cụ thể (tiêu đề truyền thông hình ảnh,..). Người ta cho rằng nhu cầu đối với các kiểu rãnh mới là hiếm, tuy nhiên hầu hết các đa phương tiện nên được chia thành các kiểu hiện có (ví dụ: bộ giải mã hình ảnh nên sử dụng các rãnh hình ảnh, giao thức gợi ý sử dụng cácrãnh chỉ dẫn,..)
3. Tệp tinmiêu tả mẫu định dạng và định dạng mẫu (còn gọi là codec names). Tệp tin này bao gồm các mật mã âm thanh, hình ảnh và các nhận dạng giao thức cho cácrãnh chỉ dẫn. Bất kì đăng kí của một định dạng mẫu mới nào cũng sẽ tự động được cấp một định danh đối tượng, do đó xác định việc vận chuyển định dạng này trong các hệ thống MPEG-4 có thể chống lại các khuôn khổ kí hiệu.
4. Định dạngtệp tin định danh theo dõi tài liệu tham khảo. Các phụ thuộc giữa các rãnh được gõ vào định dạng tệp tin (ví dụ: cácrãnh chỉ dẫn phụ thuộc vào các rãnh truyền thông được gợi ý, sử dụng một rãnh phụ thuộc kiểu gợi ý).
5. Đặc điểm kĩ thuật này bao gồm một “kiểutệp tin” bao gồm một danh sách các “thương hiệu” mà xác định được các thông số kĩ thuật của các tệp tin là phù hợp. Các cơ quan xác định các thông số tiêu chuẩn dựa trên định nghĩa cấu trúc của các định dạng tệp tin này thường sử dụng một nhãnmới để xác định các tệp tin phù hợp với đặc điểm kĩ thuật của họ. Bất kì đăng kí của một nhãn mới nào cần ghi rõ đặc diểm kĩ thuật cơ bản xác mà các nhãn đã chỉ ra.
6. Trong hệ thống khuôn khổ kí hiệu MPEG-4, giá trị objective được sử dụng để xác định các định dạng của các dòng dữ liệu. Một định dang objectype có thể yêu cầu sự độc lập của các bộ nhận dạng tệp tin trên.
7. Các mẫu nhóm liên kết dạng thông tin với các nhóm mẫu. Kiểu nhóm này có thể được đăng kí.
8. Các đa phương tiện hoặc siêu dữ liệu có thể được bảo vệ và các chương trình bảo vệ sử dụng xác định với một kiểu chương trình bảo vệ đã đăng kí.

Những mã điểm này được nhắc đến trong phần còn lại của phụ lục này là định danh đăng kí, viết tắt là RIDs.

**D.2. Thủ tục yêu cầu của một giá trị đăng kí định danh MPEG-4**

Các yêu cầu của một mã điểm MPEG-4 như chi tiết trên giá trị để xác định một định dạng dữ liệu cá nhân sẽ được áp dụng cho Tổ chức chứng nhận. Các hình thức đăng kí là có sẵn từ Tổ chức chứng nhận. Người yêu cầu phải cung cấp thông tin qui định tại D.4. Các công ti và tổ chức có đủ điều kiện để áp dụng.

**D.3. Trách nhiệm của Tổ chức chứng nhận**

Trách nhiệm cơ bản của Cơ quan đăng ký quản trị việc đăng ký của các bộ nhận dạng dữ liệu cá nhân được nêu trong phụ lục này; một số trách nhiệm khác có thể được tìm thấy trong Chỉ thị JTC 1.

Các cơ quan đăng ký có trách nhiệm:

1. Thực hiện thủ tục đăng kí cho các ứng dụng trong một RID duy nhất phù hợp với chỉ thị JTC 1;
2. Tiếp nhận và xử lí các ứng dụng phân bố địa điểm từ các nhà cung cấp ứng dụng;
3. Xác định các ứng dụng đã nhận được là phù hợp với thủ tục đăng kí và thông báo cho người đăng kí trong vòng 30 ngày kể từ ngày nhận được đơn xác nhận của RID;
4. Thông báo cho các nhà cung cấp ứng dụng có nhu cầu bị từ chối bằng văn bản trong vòng 30 ngày kể từ ngày nhận được ứng dụng và xem xét các đệ trình lại của các ứng dụng này một cách kịp thời;
5. Duy trì cơ bản xác các nhận dạng được phân bố. Sửa đổi qui cách định dạng sẽ được chấp nhận và duy trì bởi Tổ chức chứng nhận;
6. Thực hiện các nội dung của các đăng kí này có sẵn theo yêu cầu của các Cơ quan quốc gia JTC 1-là các thành viên của ISO hoặc IEC, các tổ chức liên lạc của ISO hoặc IEC và với bất kì các bên liên quan;
7. Duy trì một cơ sở dữ liệu của các định dạng yêu cầu RID được cung cấp và bị từ chối. Các bên tìm kiếm thông tin kĩ thuật về định dạng của dữ liệu các nhân có một RID sẽ có quyền truy nhập vào các thông tin thuộc một phần cơ sở dữ liệu được duy trì bởi Tổ chức chứng nhận;
8. Báo cáo hoạt động hàng năm với JTC 1, ITTF và Ban thư kí SC 29 hoặc các bên chỉ định tương ứng;
9. Dàn xếp việc sử dụng các RID hiện có bất cứ khi nào có thể.

**D.4 Thông tin liên hệ của Tổ chức chứng nhận**

Apple Computer Inc.

One Infinite Loop, M/S 301-4B

Cupertino, California 95014

USA

E-mail: mp4reg@group.apple.com

Web: <http://www.mp4ra.org/>

**D.5. Trách nhiệm của các bên đăng kí RID**

Bên yêu cầu định danh định dạng sẽ phải:

1. Sử dụng mẫu đơn và tuân theo các thủ tục do Tổ chức chứng nhận đưa ra;
2. Bao gồm một miêu tả về mục đích của định danh được đăng kí và các chi tiết kĩ thuật cần thiết như qui định trong mẫu đơn;
3. Cung cấp thông tin liên lạc chỉ ra cách một miêu tả hoàn chỉnh có thể đạt được trên cơ sở không phân biện đối xử;
4. Đồng ý tiến hành việc sử dụng có mục đích nhượng lại RID trong một khoảng thời gian hợp lí;
5. Duy trì một hồ sơ vĩnh viễn của mẫu đơn và thông báo nhận được từ Tổ chức chứng nhận nhượng lại RID.

**D.6. Thủ tục khiếu nại cho các ứng dụng bị từ chối**

Nhóm quản lí đăng kí (Registration Management Group) được thành lập để có thẩm quyền kháng cáo việc bị từ chối yêu cầu nhượng lại RID. RMG sẽ có một thành viên được đề cử bởi P- và L- các thành viên của Ủy ban kĩ thuật ISO chịu trách nhiệm về tiêu chuẩn ISO/IEC 14496. RMG sẽ có một cuộc triệu tập và đề cử bí mật từ các thành viên. Tổ chức chứng nhận được cho phép để bầu cử một quan sát viên không bỏ phiếu.

Trách nhiệm của RMG là:

1. Xem xét và hành động trên tất cả các khiếu nại trong một khoảng thời gian hợp lí;
2. Thông báo bằng văn bản, tổ chức thực hiện lời kêu gọi xem xét lại các kiến nghị về việc sắp xếp RMG;
3. Xem xét các báo cáo tóm tắt hoạt động hàng năm của Tổ chức chứng nhận;
4. Cung cấp cho các thành viên hội đồng của ISO và Ủy ban quốc gia về truyền thông với các thông tin liên quan đến phạm vi hoạt động của Tổ chức chứng nhận.

**D.7 Mẫu đơn đăng kí**

**D.7.1 Thông tin liên lạc của tổ chức đăng kí RID**

Tên tổ chức:

Địa chỉ:

Số điện thoại:

Fax:

E-mail:

Telex:

**D.7.2 Yêu cầu cho một RID cụ thể**

LƯU Ý: Nếu hệ thống đã được thực hiên và đang được sử dụng, điền vào mục và mặt hàng này D.7.3 và nhảy đến D.7.5, hoặc bỏ qua bước này và nhảy đến D.7.3

**D.7.3 Mô tả ngắn gọn của RID đang được sử dụng và hệ thống dữ liệu đã được bổ sung**

**D.7.4 Trạng thái của dự định áp dụng việc nhượng lại RID**

**D.7.5 Ngày tiến hành dự định RID**

**D.7.6 Đại diện được ủy quyền**

Tên:

Tiêu đề:

Địa chỉ:

Email:

Chữ kí:

**D.7.7 Mẫu thông báo sử dụng của Tổ chức chứng nhận**

Đăng ký không được chấp nhận \_\_\_\_\_

Lý do:

Đăng ký được chấp nhận \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Mã số đăng ký\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Đính kèm 1: Đính kèm bản chi tiết kỹ thuật của định dạng dữ liệu đăng ký.

Đính kèm 2: Đính kèm thông báo thủ tục khiếu nạicho các ứng dụngbị từ chối.

# Phụ lục E (Quy định) Các nhãn định dạng tệp tin

**E.1 Giới thiệu**

Sự xuất hiện của nhãn trong *compatible\_brandlist* của hộp *ftyp* vừa mang tính xác nhận vừa mang tính cho phép. Nó xác nhận tệp tin tuân thủ tất cả các yêu cầu của nhãn đó, và chỉ cho phép đầu đọc đọc nhãn đó được đọc tệp tin.

Nhìn chung, các đầu đọc phải thực thi tất cả các đặc điểm đã được quy định cho một nhãn trừ khi có một trong số các điều kiện sau đây:

1. Media đang được đọc không yêu cầu đặc điểm đó: ví dụ, hình ảnh I-frame không cần bảng đồng bộ mẫu, và nếu như không có việc tổng hợp và sắp xếp lại, thì sẽ không cần đến bảng độ lệch thời gian; tương tự như vậy, nếu như không cần bảo vệ nội dung, thì cũng không cần đến những hỗ trợ cho các cấu trúc bảo vệ nội dung;
2. Tệp tin tuân theo một đặc tả kỹ thuật khác không cho phép việc sử dụng đặc điểm đó (ví dụ, một số đặc tả kỹ thuật dẫn xuất cấm việc sử dụng phân mảnh movie);
3. Trong các trường hợp vận hành sản phẩm mà việc sử dụng một vài cấu trúc là không phù hợp; lấy ví dụ, các cấu trúc rãnh chỉ dẫn chỉ thích hợp với các sản phẩm dùng chuẩn bị nội dung cho, hoặc thực hiện phân phối tệp tin (chẳng hạn như streaming) cho giao thức trong rãnh chỉ dẫn.

Những phần sau đây sẽ liệt kê các nhãn được định nghĩa trong tiêu chuẩn này; thứ tự các mục không ngầm định tính thừa kế - khi xuất hiện thừa kế, nó sẽ được thông báo một cách tường minh. Các nhãn khác có thể được định nghĩa trong các tiêu chuẩn khác. Lưu ý rằng nếu một nhãn là tập con của một nhãn khác (ví dụ: các yêu cầu ‘*isom’* là tập con của các yêu cầu *‘iso2’*) thì:

1. Các tệp tin tương thích với tập con sẽ luôn luôn tương thích với tập cha; tệp tin tương thích với *‘isom’* sẽ luôn tương thích với *‘iso2’.*
2. Các sản phẩm hỗ trợ tự động cho tập cha có thể hỗ trợ cho tập con; một sản phẩm hỗ trợ cho *‘iso2’* sẽ hỗ trợ được cho *‘isom’.*

Không có nhãn nào được định nghĩa ở đây yêu cầu hỗ trợ cho bất cứ một dạng đa phương tiện (ví dụ hình ảnh, âm thanh, siêu dữ liệu), mã hóa (ví dụ một codec cụ thể nào đó), hoặc các cấu trúc hỗ trợ cho một dạng đa phương tiện cụ thể nào (ví dụ đầu vào mục mẫu hình ảnh hoặc các hộ có chứa một dạng đầu vào mục mẫu đặc thù nào đó).

Các định danh cụ thể hơn có thể được sử dụng để xác định cơ bản xác các phiên bản của các đặc tả kỹ thuật để cung cấp các thông tin cụ thể hơn. Không nên sử dụng các nhãn kiểu này làm nhãn cơ bản; định dạng tệp tin cơ sở này phải được dẫn xuất thành bản đặc tả khác để sử dụng. Bởi thế không có phần mở rộng mặc định của tệp tin, hoặc việc gán dạng MIME cho tập tin, hoặc việc định nghĩa phiên bản phụ khi một trong các nhãn đó là nhãn cơ bản.

**E.2 Nhãn ‘isom’**

Dạng tệp tin *‘isom’* (tệp tin đa phương tiện dựa trên ISO) sẽ được định nghĩa ở phần này của tiêu chuẩn này là các tệp tin tuân theo phiên bản đầu tiên của Định dạng tệp tin đa phương tiện dựa trên ISO.

Yêu cầu phải có hỗ trợ cho các hộp cấu trúc sau đây:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Các kiểu hộp** | | | | | | **Diễn giải** |
| *moov* |  |  |  |  |  | chứa tất cả các siêu dữ liệu |
|  | *mvhd* |  |  |  |  | mào đầu movie, các khai báo tổng quát |
|  | *trak* |  |  |  |  | chứa rãnh hoặc luồng riêng lẻ |
|  |  | *tkhd* |  |  |  | mào đầu rãnh, các thông tin chung về rãnh |
|  |  | *tref* |  |  |  | chứa các tham chiếu của rãnh |
|  |  | *edts* |  |  |  | chứa danh sách biên tập |
|  |  |  | *elst* |  |  | danh sách biên tập |
|  |  | *mdia* |  |  |  | chứa thông tin đa phương tiện của rãnh |
|  |  |  | *mdhd* |  |  | mào đầu đa phương tiện, thông tin chung về đa phương tiện |
|  |  |  | *hdlr* |  |  | điều khiển, ở mức này là dạng (điều khiển) đa phương tiện |
|  |  |  | *minf* |  |  | chứa thông tin đa phương tiện |
|  |  |  |  | *vmhd* |  | mào đầu đa phương tiện hình ảnh, thông tin chung(chỉ dành cho các rãnh hình ảnh) |
|  |  |  |  | *smhd* |  | mào đầu đa phương tiện âm thanh, thông tin chung (chỉ dành cho các rãnh âm thanh) |
|  |  |  |  | *hmhd* |  | mào đầu đa phương tiện chỉ dẫn, thông tin chung (chỉ dành cho các rãnh chỉ dẫn) |
|  |  |  |  | *<mpeg>* |  | các mào đầu luồng mpeg |
|  |  |  |  | *dinf* |  | nguyên tử thông tin dữ liệu, vật chứa |
|  |  |  |  |  | *dref* | nguyên tử tham chiếu dữ liệu, khai báo nguồn của đa phương tiện trong rãnh |
|  |  |  |  | *stbl* |  | nguyên tử bảng mẫu, vật chứa của ánh xạ thời gian/không gian |
|  |  |  |  |  | *stts* | (giải mã) ánh xạ thời gian và mẫu |
|  |  |  |  |  | *ctts* | bảng tổng hợp thời gian và mẫu |
|  |  |  |  |  | *stss* | ánh xạ mẫu đồng bộ (khóa, I-frame) |
|  |  |  |  |  | *stsd* | các mô tả mẫu (các kiểu codec, thiết lập,..) |
|  |  |  |  |  | *stsz* | các kích thước mẫu (đóng khung) |
|  |  |  |  |  | *stsc* | ánh xạ mẫu vào đoạn, thông tin một phần data-offset |
|  |  |  |  |  | *stco* | độ dịch chuyển đoạn, thông tin một phần data-offset |
|  |  |  |  |  | *co64* | độ dịch chuyển đoạn 64-bit |
|  |  |  |  |  | *stsh* | đồng bộ vùng phủ |
|  |  |  |  |  | *stdp* | ưu tiên suy giảm |
| *mdat* |  |  |  |  |  | hộp chứa dữ liệu đa phương tiện |
| *free* |  |  |  |  |  | không gian trống |
| *skip* |  |  |  |  |  | không gian trống |
| *udta* |  |  |  |  |  | dữ liệu người dùng, bản quyền… |
| *ftyp* |  |  |  |  |  | kiểu tệp tin và tính tương thích |
|  |  |  |  |  | *stz2* | kích thước mẫu nhỏ gọn (khung) |
|  |  |  |  |  | *padb* | bit chèn điền đầy mẫu |
|  | *mvex* |  |  |  |  | hộp mở rộng moviw |
|  |  | *mehd* |  |  |  | hộp mào đầu mở rộng movie |
|  |  | *trex* |  |  |  | mặc định mở rộng rãnh |
| *moof* |  |  |  |  |  | phân mảnh movie |
|  | *mfhd* |  |  |  |  | mào đầu phân mảnh movie |
|  | *traf* |  |  |  |  | phân mảnh rãnh |
|  |  | *tfhd* |  |  |  | mào đầu phân mảnh rãnh |
|  |  | *trun* |  |  |  | xử lý phân mảnh rãnh |
| *mfra* |  |  |  |  |  | truy cập ngẫu nhiên phân mảnh movie |
|  | *tfra* |  |  |  |  | truy cập ngẫu nhiên phân mảnh rãnh |
|  | *mfro* |  |  |  |  | độ lệch truy cập ngẫu nhiên phân mảnh movie |

Rãnh chỉ dẫn phải được thừa nhận và nằm trong rãnh chỉ dẫn, rãnh chỉ dẫn giao thức RTP.

Lưu ý rằng một số yêu cầu cho Hộp mào đầu rãnh (Track Header Box) không áp dụng đối với nhãn này; xem 8.3.2.1.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp *‘ctts’*; không yêu cầu hỗ trợ phiên bản 1.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp *‘trun’*; không yêu cầu hỗ trợ phiên bản 1.

LƯU Ý: khi tệp tin được gán nhãn này thì cờ *default-base-is-moof* (xem 8.8.7.1) sẽ không được gán giá trị.

**E.3 Nhãn ‘avc1’**

Nhãn ‘avc1’ được sử dụng để chỉ ra rằng tệp tin tuân thủ với ‘Các mở rộng AVC’ trong các mục con 8.6.4 và 8.9. Nếu được sử dụng mà không kèm theo các nhãn khác, thì ngầm định yêu cầu hỗ trợ cho các mở rộng này. Tiêu chuẩn này cho phép có thể sử dụng ‘avc1’ làm nhãn cơ bản; trong trường hợp đó đặc tả kỹ thuật đó sẽ định nghĩa phần mở rộng của tệp tin và các hành vi yêu cầu.

Nhãn ‘avc1’ yêu cầu hỗ trợ cho nhãn ‘*isom*’. Thêm vào đó, yêu cầu cần có hỗ trợ cho các hộp sau đây:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Các kiểu hộp** | | | | | | **Diễn giải** |
|  |  |  |  |  | *sdtp* | các mẫu độc lập có thể kiểu bỏ được |
|  |  |  |  |  | *sbgp* | ánh xạ mẫu và nhóm |
|  |  |  |  |  | *sgpd* | miêu tả nhóm mẫu |

Trong các nhóm mẫu, yêu cầu phải có hỗ trợ cho các nhóm roll (kiểu nhóm *‘roll’*).

LƯU Ý: khi tệp tin được gán nhãn này thì cờ *default-base-is-moof* (xem 8.8.7.1) sẽ không được gán giá trị.

Lưu ý rằng một số yêu cầu cho Hộp mào đầu rãnh không áp dụng đối với nhãn này; xem mục con 8.3.2.1.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*ctts*’; không yêu cầu hỗ trợ phiên bản 1.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*trun*’; không yêu cầu hỗ trợ phiên bản 1.

Không yêu cầu hỗ trợ cho các hộp SampleDescription trong các phân đoạn movie.

**E.4 Nhãn ‘iso2’**

Nhãn *‘iso2’* được sử dụng để biểu thị tính tương thích với phiên bản thứ hai của Định dạng tệp tin đa phương tiện dựa trên ISO; nó có thể được sử dụng bổ trợ hoặc thay thế cho nhãn *‘isom’* và có cùng các luật sử dụng. Nếu được sử dụng mà không kèm theo nhãn *‘isom’* chỉ thị phiên bản đầu tiên của đặc tả kỹ thuật này, thì nó sẽ chỉ thị yêu cầu hỗ trợ một vài hoặc tất cả các công nghệtrong các mục con 8.6.4, 8.9, 8.11.1 cho tới 8.11.7, 8.11.10, 8.12, hoặc hỗ trợ SRTP trong mục con 9.1.

Nhãn *‘iso2’* yêu cầu hỗ trợ tất cả các đặc điểm của nhãn *‘avc1’.*

Thêm vào đó, yêu cầu hỗ trợ các hộp sau:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Các kiểu hộp** | | | | | | **Diễn giải** |
| *pdin* |  |  |  |  |  | thông tin tải xuống liên tục |
|  |  |  |  |  | *subs* | thông tin trích mẫu |
| *meta* |  |  |  |  |  | siêu dữ liệu |
|  | *iloc* |  |  |  |  | vị trí hạng mục |
|  | *ipro* |  |  |  |  | bảo vệ hạng mục |
|  |  | *sinf* |  |  |  | hộp thông tin kế hoạch bảo vệ |
|  |  |  | *frma* |  |  | hộp định dạng nguyên thủy |
|  |  |  | *schm* |  |  | hộp kiểu kế hoạch |
|  |  |  | *schi* |  |  | hộp thông tin kế hoạch |
|  | *iinf* |  |  |  |  | thông tin hạng mục (trường phiên bản được gán giá trị 0) |
|  | *xml* |  |  |  |  | hộp chứa XML |
|  | *bxml* |  |  |  |  | hộp chứa XML nhị phân |
|  | *pitm* |  |  |  |  | tham chiếu hạng mục cơ bản |

Trong ngữ cảnh của Rãnh chỉ dẫn RTP, cần phải nhận biết được các rãnh chỉ dẫn SRTP. Yêu cầu phải có bảo vệ nội dung và hỗ trợ chung cho các hộp siêu dữ liệu.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp thông tin mục, và phiên bản 0 của hộp vị trí mục.

Lưu ý rằng một số yêu cầu cho Hộp Mào đầu Rãnh không áp dụng đối với nhãn này; xem mục con 8.3.2.1.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*ctts*’; không yêu cầu hỗ trợ phiên bản 1.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*trun*’; không yêu cầu hỗ trợ phiên bản 1.

Không yêu cầu hỗ trợ cho các hộp SampleDescription trong các phân mảnh movie.

LƯU Ý: khi tệp tin được gán nhãn này thì cờ *default-base-is-moof* (xem 8.8.7.1) sẽ không được gán giá trị.

**E.5 Nhãn ‘mp71’**

Nếu một Hộp Siêu dữ liệu với kiểu bộ xử lý MPEG-7 được sử dụng ở mức tệp tin, thì nhãn *‘mp71’* cần được đưa vào danh sách các nhãn tương thích trong hộp kiểu tệp tin.

**E.6 Nhãn ‘iso3’**

Nhãn *‘iso3’* yêu cầu hỗ trợ tất cả các đặc điểm của nhãn *‘iso2’.*

Thêm vào đó, yêu cầu hỗ trợ các hộp sau đây:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Các kiểu hộp** | | | | | | **Diễn giải** |
|  | *fiin* |  |  |  |  | thông tin hạng mục phân phối tệp tin |
|  |  | *paen* |  |  |  | đầu vào mục phân vùng |
|  |  |  | *fpar* |  |  | phân vùng tệp tin |
|  |  |  | *fecr* |  |  | dự trữ FEC |
|  |  | *segr* |  |  |  | nhóm phiên phân phối tệp tin |
|  |  | *gitn* |  |  |  | ánh xạ tên và id nhóm |
| *meco* |  |  |  |  |  | vật chứa siêu dữ liệu bổ sung |
|  | *mere* |  |  |  |  | liên hệ hộp siêu dữ liệu |

Yêu cầu hỗ trợ phiên bản 0 và phiên bản 1 cho hộp thông tin hạng mục.Trong các nhóm mẫu, yêu cầu phải có hỗ trợ cho tốc độ chia sẻ thông tin (kiểu nhóm ‘rash’). Các rãnh chỉ dẫn phân phối tệp tin (đầu vào mục mẫu*‘fdp’*) phải được XXXX.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*ctts*’; không yêu cầu hỗ trợ phiên bản 1.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*trun*’; không yêu cầu hỗ trợ phiên bản 1.

Không yêu cầu hỗ trợ cho các hộp SampleDescription trong các phân mảnh movie.

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp vị trí hạng mục.

LƯU Ý: khi tệp tin được gán nhãn này thì cờ *default-base-is-moof* (xem 8.8.7.1) sẽ không được gán giá trị.

**E.7 Nhãn ‘iso4’**

Nhãn *‘iso4’* yêu cầu hỗ trợ tất cả các đặc điểm của nhãn *‘iso3’.*

Nhãn này yêu cầu hỗ trợ phiên bản 1 của các hộp độ lệch tổng hợp (‘*ctts’* and ‘*iloc’*).

Yêu cầu hỗ trợ phiên bản 1 của hộp vị trí hạng mục, phiên bản 2 của hộp thông tin hạng mục, hộp dữ liệu hạng mục mới và hộp tham chiếu hạng mục.

Thêm vào đó, yêu cầu hỗ trợ các hộp sau:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Các kiểu hộp** | | | | | | **Diễn giải** |
|  | *trgr* |  |  |  |  | chỉ thị nhóm rãnh |
|  |  |  |  | *cslg* |  | ánh xạ giữa tổ hợp và thời gian giải mã |
|  | *idat* |  |  |  |  | dữ liệu hạng mục |
|  | *iref* |  |  |  |  | tham chiếu hạng mục |

Trong tài liệu này chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*trun*’; không yêu cầu hỗ trợ phiên bản 1.

Không yêu cầu hỗ trợ cho các hộp SampleDescription trong các phân mảnh movie.

LƯU Ý: khi tệp tin được gán nhãn này thì cờ *default-base-is-moof* (xem 8.8.7.1) sẽ không được gán giá trị.

**E.8 Nhãn ‘iso5’**

Nhãn *‘iso5’* yêu cầu hỗ trợ tất cả các đặc điểm của nhãn *‘iso4’*.

Nhãn này yêu cầu hỗ trợ cờ *default-base-is-moof*.

Nhãn này yêu cầu xử lý các đầu vào mục mẫu giới hạn (chẳng hạn như ‘*resv’*).

Chỉ yêu cầu hỗ trợ phiên bản 0 của hộp ‘*trun*’; không yêu cầu hỗ trợ phiên bản 1.

Không yêu cầu hỗ trợ cho các hộp SampleDescription trong các phân mảnh movie.

**E.9 Nhãn ‘iso6’**

Nhãn *‘iso6’* yêu cầu hỗ trợ tất cả các đặc điểm của nhãn *‘iso5’*.

Thêm vào đó, yêu cầu hỗ trợ các hộp sau:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Các kiểu hộp** | | | | | **Mục tham chiếu** | **Diễn giải** |
|  |  | *saiz* |  |  | 8.7.8 | các kích thước thông tin bổ trợ mẫu |
|  |  | *saio* |  |  | 8.7.9 | các độ dịch chuyển thông tin bổ trợ mẫu |
|  |  | *tfdt* |  |  | 8.8.12 | thời gian giải mã phân mảnh rãnh |
| *styp* |  |  |  |  | 8.16.2 | kiểu phân đoạn |
| *sidx* |  |  |  |  | 8.16.3 | chỉ số phân đoạn |
| *ssix* |  |  |  |  | 8.16.4 | chỉ số phân đoạn con |
| *prft* |  |  |  |  | 8.16.5 | thời gian tham chiếu của nhà sản xuất |

Nhãn này yêu cầu các hỗ trợ sau đây:

* Các hộp SampleGroupDescription trong các phân mảnh movie;
* Độ lệch tổng hợp (có dấu) trong các hộp chạy rãnh (tức là phiên bản 1 của các hộp chạy rãnh);
* Trong cùng các nhóm mẫu, yêu cầu hỗ trợ cho thông tin điểm truy nhập ngẫu nhiên (kiểu nhóm *‘rap’*).

**E.10 Nhãn ’iso7’**

Nhãn ‘iso7’ yêu cầu phải hỗ trợ tất cả các tính năng của nhãn ‘iso6’.

Nhãn này cũng cần phải hỗ trợ:

* Các giá trị *item\_ID* và item\_value 32 bit;
* Nhận dạng được các rãnh không hoàn chỉnh.

# Phụ lục F (Tham khảo) Các dạng siêu dữ liệu được dán nhãn URI

**F.1 Siêu dữ liệu dán nhãn UUID**

Dạng thức của URI cho siêu dữ liệu dán nhãn UUID được định nghĩa trong IETF RFC 4122: A Universally Unique IDentifier (UUID), URN *Namespace*(tháng 7/2005).

Không có định nghĩa chung nào về dạng của siêu dữ liệu gốc, dữ liệu khởi tạo cho siêu dữ liệu tạm thời, hay siêu dữ liệu tạm thời. Định dạngcủa tất cả những dữ liệu này phụ thuộc vào UUID và định nghĩa của nó.

Tuy nhiên không dễ để các UUID tìm đến điểm gốc của nó, vì vậy sẽ gây ra khó khăn cho bên nhận siêu dữ liệu tìm kiếm thông tin về nó.

Trong trường hợp muốn truy xuất tìm đến điểm gốc thì nên sử dụng một khung siêu dữ liệu được tiêu chuẩn hóa như là MPEG-7, hay một khung đã được đăng kí như SMPTE, hoặc là một de-referencable URL.

**F.2Siêu dữ liệu dán nhãn ISO OID**

Dạng thức của URI cho siêu dữ liệu dán nhãn OID được định nghĩa trong RFC 3061 : A URN *Namespace* of Object Identifiers (2/ 2001).

Không có định nghĩa chung nào về dạng của siêu dữ liệu gốc, dữ liệu khởi tạo cho siêu dữ liệu tạm thời, hay siêu dữ liệu tạm thời. Dạng của tất cả những dữ liệu này phụ thuộc vào UUID và định nghĩa của nó.

Nhiều hệ thống gắn nhãn cụ thể có thể mô tả các định danh đối tượng. Nên sử dụng dạng UUID thật cụ thể.

Không nên dùng bộ nhận dạng đối tượng bắt đầu bằng {joint-iso- itu(2) uuid(25)}(ví dụ: starting urn:oid:2.25). Nên sử dụng trực tiếp UUID URI.

Không nên sử dụng bộ nhận dạng dữ liệu bắt đầu với {iso (1) identified-organization(3) SMPTE(52) metadata-dictionary(1)}(ví dụ urn:oid:1.3.52.1), và cũng không nên sử dụng bất kì OID nào khác được dán nhãn như SMPTE 298M hoặc 336M; nên dùng dạng SMPTE URI đặc trưng hơn.

Các bộ nhận dạng đối tượng được đăng kí với các tổ chức riêng biệt do đó có thể xác định được tổ chức đang sở hữu một bộ nhận dạng này. Tuy nhiên một số phần của bộ nhận dạng đối tượng lại được ủy quyền cho người sử dụng chưa đăng kí (ví dụ như UUIDs như lưu ý ở trên) nên sau đó khả năng truy xuất nguồn tìm đến điểm gốc bị mất.

Nếu cần truy xuất tìm đến điểm gốc thì nên sử dụng một khung siêu dữ liệu được tiêu chuẩn hóa như là MPEG-7, hay một khung đã được đăng kí như là SMPTE, hoặc de-referencable URL.

**F.3 Siêu dữ liệu gán nhãn SMPTE**

Dạng thức của URI cho siêu dữ liệu dán nhãn SMPTE được xác định trong RFC 5119; A Uniform Resource Name (URN) *namespace* for the Society of Motion Picture and Television Engineers (SMPTE).

Siêu dữ liệu gốc ở đây là phần giá trị(V) của bộ ba KLV(key- chìa khóa, length- độ dài, value-giá trị) được xác định ở SMPTE 336M, với chìa khóa(K) là nhãn đưa ra ở URN và độ dài (L) bắt nguồn từ độ dài của điểm.

Tương tự mỗi mẫu siêu dữ liệu thời gian là một phần giá trị của một KLV, trong đó chìa khóa(K) là nhãn URN, và độ dài(L) bắt nguồn từ kích thước của mẫu (tạo ra trong kích thước của mẫu hoặc bảng kích thước mẫu).

Dữ liệu khởi tạo chứa chìa khóa(K) và giá trị(V) của một KLV cung cấp bối cảnh khởi tạo cho KLVs hình thành từ các mẫu, với độ dài (L) bắt nguồn từ kích thước Hộp Dữ liệu. 16 bytes đầu tiên là 1 nhãn SMPTE của dữ liệu khởi tạo, được lưu trữxác định trong SMPTE 336M.

Giá trị của những bytes này được định nghĩa trong SMPTE 377M là “primer pack” (hệ cơ số 16): 06 0E 2B 3402 05 01 01 0D 01 02 01 01 05 01 00. Tuy nhiên, nếu nhãn của dữ liệu khởi tạo không xác định một cấu trúc bối cảnh cụ thể (ví dụ như 1 primer pack), thì một local set sẽ được hình thành. Khái niệm về Local Set được định nghĩa rõ trong SMPTE 377M.

SMPTE 377M sử các định vị để xác định vị trí các tài nguyên khác bên ngoài siêu dữ liệu. Đối với các siêu dữ liệu tĩnh nên sử dụng hộp định vị trong Hộp Siêu dữ liêu. Đối với siêu dữ liệu tạm thời, có thể sử dụng trực tiếp các vị trí từ bên ngoài.

Dữ liệu khởi tạo có thể không có,và nhãn này sẽ xác định được một điểm siêu dữ liệu riêng(ví dụ như định vị) mà không cần bối cảnh.

# Phụ lục G (Tham khảo) Quá trình xử lý các luồng RTP và rãnh chỉ dẫn tiếp nhận

**G.1 Giới thiệu**

**G.1.1 Khái quát**

Phụ lục này khuyến nghị việc ghi cácluồng RTP và việc sử dụng các luồng RTP đã ghi được này để phát và gửi lại.

**G.1.2 Cấu trúc**

Phụ lục này được bố cục như sau:

* G.2 giới thiệu các nguồn tiềm năng tại sao việc phát lại các luồng RTP có thể không đồng bộ và cung cấp một cái nhìn tổng quan về việc làm thế nào để việc đồng bộ hóa chuẩn xác trong quá trình ghi và phát. Điều này đã được đề cập trong các mục trước đó bởi vì cả bộ phận ghi và phát phải hoạt động để đồng bộ hóa chuẩn xác;
* G.3 đưa ra các khuyến nghị về việc lưu trữ các luồng RTP;
* G.4 đưa ra các khuyến nghị về cách phát các tệp tin có chứa các luồng RTP đã ghi;
* G.5 đưa ra các khuyến nghị cho việc gửi lại các luồng RTP đã được lưu trữ trong các tệp tinnhư miêu tả ở G.3.

**G.1.3 Các thuật ngữ và định nghĩa**

Các thuật ngữ và định nghĩa sau đây được áp dụng trong phụ lục này:

**G.1.3.1 Thiết bị đọc (player)**

Thực thể phân tách các tệp tin, giải mã ít nhất một tập hợp con của các rãnh ghi trong tệp tin và kết xuất (render) các rãnh ghi đã giải mã.

**G.1.3.2 Bộ phận ghi (recoding unit)**

Thực thể nhận một hay nhiều luồngdữ liệu được đóng gói và lưu trữ dữ liệu nhận được trong mộttệp tin.

**G.1.3.3 Bộ phận gửi lại (re-sending unit)**

Thực thể này sẽ phân tách mộttệp tin hình thành từ một hay nhiều các luồngdữ liệu được đóng gói và truyền ít nhất một tập con của dữ liệu được lưu trữ trong tệp tin.

**G.2 Quá trình đồng bộ hóa của luồng RTP**

Trong một số tài liệu đã chỉ ra có một số nguồn của phần phát lại không đồng bộ hóa đối với các luồng RTP đã nhận. Chẳng hạn như, khi các luồng RTP được thu nhưrãnh chỉ dẫn tiếp nhận RTP(khái niệm đc nêu trong mục 9.1), thì các thông tin cần thiết để đảm bảo việc đồng bộ hóa bộ phận phát lại cũng sẽ được ghi lại. Khi các luồng RTP được thu như Rãnh đa phương tiện, thì quá trình đồng bộ các bộ phận phát lại của rãnh đa phương tiện phải được đảm bảo bằng cách tạo ra thời gian tổng hợp của môi trường mẫu phù hợp. Danh sách sau đây miêu tả các nguồn của bộ phận phát lại không đồng bộ để nhận các luồngRTP, tóm tắt các phương thức đồng bộđã được giới thiệu và cho biết các điều khoản liên quan để có thêm thông tin.

1. Nhãn thời gian RTP của gói đầu tiên của luồng có một độ dịch chuyển ngẫu nhiên. Ngoài ra, nhãn thời gian RTP của 2 luồng dịch chuyển dựa vào sự khác nhau giữa các độ dịch chuyển ngẫu nhiên ban đầu ngay cả khi tốc độ xung nhịp của nhãn thời gian RTP đã được cân bằng. Giá trị độ dịch chuyển ngẫu nhiên nên được nêu ra ở trường giá trị của độ dịch chuyển tronghộp `tsro´ như miêu tả ở G.3.5
2. Gói thu và nhận đầu tiên của các luồngkhác nhau có thể không có thời gian phát lại xác định như đã nói ở G.3.2. Thời gian bắt đầu không bằng nhau của các luồngthu khác nhau được bù bằng cách phân tách 1 hoặc nhiều báo cáo của bên gửi RTCP để có thời gian phát lại đúng như thời gian thực bên phía bên gửivà có thể tạo ra độ dịch chuyển ban đầu cho bộ phận phát lại sử dụng Hộp Danh sách Biên tậpnhư miêu tả ở G.3.2. Hộp Danh sách Biên tậpđược dịch bằng thiết bị đọc như miêu tả ở G.4.4.
3. Không có gì đảm bảo rằng đồng hồ tạo ra nhãn thời gian RTP của 1 luồng RTP nào đó chạy cùng tốc độ đồng hồ thời gian thực của bên gửi, được sử dụng để tạo ra báo cáo của bên gửi RTCP. Ví dụ nhãn thời gian RTP có thể tạo ra tần số âm thanh 44.1kHz dựa trên cơ sở 1 tần số mẫu liên tục, và do đó nó chịu ảnh hưởng bởi tốc độ xung nhịp. Tuy nhiên báo cáo của bên gửi RTP có thể được tạo ra dựa vào đồng hồ hệ thống chạy ở các tốc độ khác so với tốc độ xung nhịp. Ngoài ra, đồng hồ sử dụng để tạo ra nhãn thời gian RTP cho âm thanh có thể chạy ở tốc độ khác so với đồng hồ sử dụng để tạo ra nhãn thời gian RTP cho hình ảnh.

Một vấn đề tương tự phát sinh trong các thiết bị trình diễn là nếu đồng hồ theo kịp đầu ra của luồngđược giải mã chạy ở tốc độ khác với đồng hồ thời gian thực của thiết bị đọc, hoặc là đồng hồ theo kịp sự dịch chuyển của các luồng được giải mã khác sẽ không được đồng bộ hóa.

Vì vây, đề xuất cho tất cả các vấn đề có thể xảy ra khi đồng hồ chạy ở tốc độ khác nhau là sử dụng báo cáo của bên gửi RTCP để sắp xếp nhãn thời gian RTP của các luồngkhác nhau vào cùng 1 luồngthời gian của đồng hồ thời gian thực (sử dụng để đồng bộ hóa inter-stream). Có thể sắp xếp trong khi thu các luồngbằng cách thay đổi đại diện của nhãn thời gian RTP đã thu, hoặc là trong khi chạy các luồngđược thu bằng cách sử dụng báo cáo của bên gửi RTCP như miêu tả ở G.3.6. Hơn thế, cũng có thể dẫn tốc độ bộ phận phát lại theo tỉ lệ âm thanh như miêu tả ở G.4.4.

1. Đồng hồ thời gian thực của bên gửi có thể chạy ở tốc độ khác với đồng hồ thời gian thực của thiết bị đọc. Ở đây khi ta sử dụng 1 chương trình thu âm ở tốc độ của đồng hồ thời gian thực của thiết bị đọc và sử dụng đồng hồ phát liên quan đến âm thanh như đồng hồ thời gian thực của thiết bị đọc thì kết quả thang thời gian của âm thanh không thay đổi ngay cả khi đồng hồ thời gian thực của thiết bị đọc chạy ở tốc độ khác với tốc độ đồng hồ của bên gửi.

**G.3 Việc ghi các luồng RTP**

**G.3.1 Giới thiệu**

Việc ghi các luồng RTP có thể chia làm 3 cấu trúc tệp tin cơ bản như sau:

1. Một tệp tin chỉ chứa rãnh chỉ dẫn tiếp nhận RTP, không bao gồm rãnh đa phương tiện. Cấu trúc tệp tin này cho phép đánh giá quá trình hiệu quả của việc mất gói, nhưng chỉ thiết bị đọc có khả năng phân tách rãnh chỉ dẫn tiếp nhận RTP mới có thể chạy tệp tin này.
2. Tệp tin chỉ chứa các rãnh đa phương tiệnkhông bao gồm rãnh chỉ dẫn tiếp nhận RTP. Cấu trúc tệp tin này cho phép các thiết bị trình diễn tuân theo tiêu chuẩn của định dạng tệp tin dựa trên tiêu chuẩn ISO để thu các tệp tin cũng như các dạng đa phương tiện được hỗ trợ.
3. Một tệp tin chứa cả rãnh chỉ dẫn tiếp nhận RTPvà các rãnh đa phương tiện. Cấu trúc tệp tin này có lợi ích riêng và được sử dụng tương hỗ với các dạng tệp tin khác được hình thành từ định dạng tệp tinđa phương tiện dựa trên tiêu chuẩn ISO.

Nếu 1 luồng RTP được bảo vệ, một rãnh chỉ dẫn tiếp nhận RTPđược bảo vệ sẽ thay thế bằng 1 rãnh chỉ dẫn tiếp nhận RTP. Vào thời điểm của quá trình phát lại, dữ liệu bên trong rãnh chỉ dẫn tiếp nhận RTPkhông được bảo vệ và sau đó quá trình tương tự xảy ra đối với luồng RTP không được bảo vệ. Thay vào đó, luồng RTP có thể không được bảo vệ trước khi lưu trữ nó như rãnh chỉ dẫn tiếp nhận RTP, nhưng sau đó phải tuân theo quyền sử dụng một số nội dung trong luồng RTP được bảo vệ.

Một số thao của phần thu phổ biến cho cả 3 cấu trúc tệp tin. Bảng G1 chỉ ra các thao tác thu nào là cần thiết cho cấu trúc tệp tin cơ bản.

**Bảng G1.**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tệp tin chỉ chứa rãnh chỉ dẫn tiếp nhận RTP** | **Tệp tin chứa rãnh đa phương tiện** | **Tệp tin chứa cả rãnh chỉ dẫn tiếp nhận RTP và rãnh đa phương tiện** |
| **Bù cho vị trí bắt đầu không giống nhau của các luồngnhận RTP** | Không | Có | Không |
| **Ghi SDP** | Có | Không | Có, chỉ đối với rãnh chỉ dẫn tiếp nhận RTP |
| **Tạo một mẫu với một rãnh chỉ dẫn tiếp nhận RTP** | Có | Không | Có, chỉ đối với rãnh chỉ dẫn tiếp nhận RTP |
| **Thể hiện nhãn thời gian RTP** | Có | Không | Có, chỉ đối với rãnh chỉ dẫn tiếp nhận RTP |
| **Phần thu để tạo điều kiện đồng bộ luồngtrong việc phát lại** | Có | Có | Có |
| **Thể hiện các khoảng thời gian tiếp nhận** | Có | Không | Có, chỉ đối với rãnh chỉ dẫn tiếp nhận RTP |
| **Tạo các mẫu đa phương tiện** | Không | Có | Có, chỉ đối với rãnh đa phương tiện |
| **Tạo các mẫu chỉ tham chiếu tới các mẫu đa phương tiện** | Không | Không | Có |

**G.3.2 Bù vị trí bắt đầu không giống nhau của các luồng RTP**

Khi bắt đầu thu các luồngRTP, thời gian ban đầu của mẫu đa phương tiện đầu tiên trong 1 luồng RTP có thể không bằng thời gian mẫu đa phương tiện đầu tiên ở 1 luồng RTP khác, bởi 1 số lý do sau đây:

* Tần số mẫu của âm thanh và hình ảnh khác nhau;
* Luồngâm thanh và hình ảnh có thể không hoàn toàn xen kẽ nhau bằng việc biểu hiện thời gian trong quá trình truyền.

Nếu rãnh chỉ dẫn tiếp nhận RTCP được lưu trữ, thì ta nên sử dụng bù cho vị trí ban đầu không giống nhau của luồng RTP nhận ở thời gian phát lại, và không nên tạo ra Hộp Danh sách Biên tập liên quan đến rãnh chỉ dẫn tiếp nhận RTP.Nếu rãnh chỉ dẫn tiếp nhận RTCP không được lưu trữ hoặc nếu các các rãnh đa phương tiệnđược lưu trữ, thì các thành phần phía thu cần phải chỉ rõ độ trễ của các luồng khởi tạo, để đồng bộ hóa hình ảnh và âm thanh vào lúc phát lại của các luồng được miêu tả ở phần sau. Ở phía thu nên thực hiện các thao tác dưới đây:

1. Một báo cáo bên gửi RTCP cho biết nhãn thời gian RTP nào đúng với thời gian đồng hồ thời gian thực của thời gian báo cáo được gửi đi. Ít nhất báo cáo của bên gửi RTCP đầu tiên của mỗi luồng RTP nên được phân tách để tạo 1 nhãn thời gian tương ứng của mỗi luồng RTP và thời gian từ đồng hồ thời gian thực của bên gửi. Nhãn thời gian đồng hồ thời gian thực của gói RTP nhận được sớm nhất theo thứ tự biểu diễn được lấy ra từ mỗi luồng RTP bằng phép ngoại suy tuyến tính đơn giản;
2. Nhãn thời gian đồng hồ thời gian thực nhỏ nhất lấy ra ở trên, giữa tất cả các luồng RTP nhận sẽ được ánh xạ đến nhãn thời gian 0 trong luồng trục thời gian movie, ví dụ như việc trình diễn ngay vào lúc đầu tiên phát lại của tệp tin được thu. Trục thời gian đóng vai trò là điều khiển bộ phận phát lại của tệp tin;
3. Trục thời gian đa phương tiện của mỗi rãnh bắt đầu từ 0. Để chuyển trục thời gian đa phương tiện sang đúng vị trí bắt đầu trong trục thời gian movie, người ta tạo ra một Hộp Biên tập và một Hộp Danh sách Biên tập cho mỗi đường RTP khác như sau (không chứa gói có nhãn thời gian đồng hồ thời gian thực sớm nhất).

Hộp Danh sách Biên tậpcó 2 giá trị đầu vào:

1. Giá trị đầu tiên là một chỉnh sửa trống (biểu thị bằng *media\_time* bằng -1). Độ dài của nó (*segment\_duration*) tính bằng độ trênh lệch giữa thời gian biểu diễn của mẫu đa phương tiện sớm nhất trong toàn bộ các luồng RTPvà mẫu đa phương tiện sớm nhất của rãnh. HìnhG.1 trình bày ví dụ làm thế nào để nhận được 1 *segment\_duration* của giá trị vào đầu tiên trong Hộp Danh sách Biên tập;
2. Giá trị của *media\_rate\_integer*của đầu vào mục thứ 2 bằng thời gian tổng hợp của mẫu được trình diễn sớm nhất, và giá trị *segment\_duration* của đầu vào mục thứ 2 trải rộng trên toàn bộ rãnh. Vì khoảng thời gian thực của rãnh có thể chưa biết tại thời điểm tạo Hộp Danh sách Biên tập nên khuyến nghị thiết lập *segment\_duration* với giá trị dương lớn nhất (đối với cả số nguyên 32 bit hoặc 64 bit tùy theo từng phiên bản của hộp).

Giá trị *media\_rate\_integer* bằng 1 đối với các đầu vào mục của Hộp Danh sách Biên tập.



**Hình G.1 – Ví dụ về Hộp Danh sách Biên tập thực hiện việc bù thời gian khởi bắt đầu không đều nhau của các luồng RTP nhận được, *segment\_duration* được sao chép vào đầu vào mục đầu tiên của Hộp Danh sách Biên tập.**

Một số bộ phận ghi có thể phát hiện các gói bắt đầu giải mã, ví dụ luồng hình ảnh IDR của H. 264/AVC, ở đây đề cập đến điểm truy nhập ngẫu nhiên. Nếu 1 luồng chứa gói có nhãn thời gian đồng hồ thời gian thực sớm nhất trong số tất cả các luồng đã nhận và cùng luồng chứa các gói có trước theo thứ tự giải mã, thì gói có trước điểm truy nhập ngẫu nhiên đầu tiên của luồng không nên lưu trữ và cũng không cần xem xét chúng khi quyết định nhãn thời gian đồng hồ thời gian thực sớm nhất trong tất cả các luồng đã nhận.

**G.3.3 Việc ghi SDP**

SDP nên được lưu trữ như sau:với SDP mức phiên (session-level), ví dụ tất cả các dòng trước dòng đa phương tiện cụ thể đầu tiên ("m=" dòng),nên được lưu như là thông tin SDP movie trong Hộp Dữ liệu Người dùng(cụ thể trong 9.1.4.1). Mỗi phần mức đa phương tiện (media-level) trong SDP bắt đầu bằng “m=" dòng và tiếp tục đến phần mức đa phương tiện tiếp theo hoặc đên cuối của miêu tả toàn bộ phiên. Mỗi phần mức đa phương tiện nên được lưu trữ như thông tinSDP rãnh trong Hộp Dữ liệu Người dùng của rãnh chỉ dẫn tiếp nhận RTP tương ứng.

**G.3.4 Tạo mẫu trong 1 rãnh chỉ dẫn tiếp nhận RTP**

Người ta nói rằng mỗi mẫu đại diện cho tất cả các gói RTP nhận có cùng nhãn thời gian RTP, ví dụ các gói liên tiếp trong dãy số thứ tự RTP với 1 nhãn thời gian RTP thông thường. Cấu trúc RTP mẫu được cài đặt có chứa 1 cấu trúc gói RTP với mỗi gói RTP nhận có cùng nhãn thời gian RTP. Mỗi gói RTP được đề xuất chứa 1 RTPampleconstructor. Một RTPampleconstructor sẽ sao chép 1 dãy byte riêng, chỉ ra bởi *sampleoffset* và *length*,đối với một mẫu cụ thể sẽ cho biết bởi *samplenumber* của trong các chỉ dẫn tham khảo của các dữ liệu gói được xây dựng. Dữ liệu của mỗi gói RTP nhận có cùng nhãn thời gian RTP được sao chép đến phần *extradata* của mẫu. rãnh tham khảo được đặt bằng điểm trong các rãnh chỉ dẫn, ví dụ được đặt bằng -1, sampleoffset và length được đặt cho phù hợp với vị trí và kích thước của dữ liệu gói trong mẫu.

Hình G.2 ví dụ về 1 mã giả của mộtmẫu chỉ dẫn tiếp nhận RTP chứa 2 gói tin RTP.



**Hình G.2 – Ví dụ về một mẫu chỉ dẫn tiếp nhận RTP có 2 gói tin (mào đầu và nội dung).**

Việc sử dụng sự kiện chỉ thịxuất hiện lỗi để biểu thị việc mất gói dữ liệu không được khuyến nghị bởi vì trường RTP sequenceseed có thể được sử dụng để phát hiện việc mất mát gói dữ liệu mà không làm tăng không gian lưu trữ. Hơn thế nữa, đơn vị nhỏ nhất của sự kiện xuất hiện lỗi có thể liên quan chính là mẫu(trong 1 rãnh chỉ dẫn tiếp nhận RTP). Vì 1 mẫu có thể chứa nhiều gói nên rất khó để biết gói nào trong những gói này sự kiện chỉ thị lỗi đã đề cập đến.

**G.3.5 Sự biểu diễn của nhãn thời gian RTP**

Nhãn thời gian RTP được tính trong rãnh chỉ dẫn tiếp nhận RTP bằngtổng của 3 giá trị, 1 trong 3 giá trị đó là thời gian giải mã DT trong trục thời gian đa phương tiện của rãnh. Thời gian giải mã là độ dài được viết bằng mã thành thời gian giải mã đến Hộp Mẫu và thêm vào 1 hoặc nhiều cácHộp Xử lý Phân mảnh Rãnh. Thời gian giải mã đến Hộp Mẫu bao gồm 1 số các *sample\_count* và cặp *sample\_delta* nơi mà *sample\_delta*là thời gian được giải mã của mỗi mẫu trong 1 bộ mẫu liên tục, số lượng bằng với *sample\_count*(ví dụ khoảng thời gian mẫu của thời gian giải mã).Hộp Xử lý Phân mảnh Rãnh cho biết 1 cặp *sample\_count* và *sample\_duration* trong đó *sample\_duration* là thời gian giải mã cho mỗi mẫu trong 1 bộ mẫu liên tục, số lượng bằng *sample\_count*(ví dụ khoảng thời gian mẫu). MỗiHộp Phân mảnh Rãnh có thể chứa một số cácHộp Xử lý Phân mảnh Rãnh. Thời gian giải mã DT(i) cho mẫu số i được suy ra từkhoảng thời gian mẫu của tất cả các mẫu trước mẫu i của thời gian giải mã và Hộp Mẫu hoặc từ cácHộp Xử lý Phân mảnh Rãnh đề cập đến bất kì mẫu nào trước mẫu i.

Nhãn thời gian của mẫu i, RTPTS(i) được biểu thị bằng tổng của 3 giá trị đặc thù sau đây:

RTPTS(i)=(DT(i)+tsro.offset+offset) mod 223 (G.1)

Trong đó tsro.offset là giá trị của độ dịch chuyển trong hộp ‘tsro’ và độ dịch chuyển là giá trị trong p hộp rtpoffsetTLV của cấu trúc gói RTP, mod là phép toán module.

Một hộp tsro nên được biểu thị trong biểu trị vào mẫu chỉ dẫn tiếp nhận RTP. Giá trị của độ dịch chuyển trong bất kì hộp tsronào của 1 rãnh nào cũng nên được tính theo nhãn thời gian gói đầu tiên của luồngtương ứng trong dãy thứ tự RTP.

Nếukhông có lặp lạithì giá trị nhãn thời gian RTP sẽ vượt qua tối đa 32 bit số nguyên giữa mẫu i-1 và i, sự khác nhau giữa nhãn thời gian RTP không bằng nhau liên tục trong dãy số thứ tự RTP là:

RTPTS\_DIFF(i)= RTPTS(i)- RTPTS(i-1) với mọi i>1 (G.2)

RTPTS\_DIFF(i) ở đây giữ nguyên không đổi, số lượng khung trong bất kì gói nào không thay đổi, thứ tự truyền giống như thứ tự biểu diễn. Sự ràng buộc này thường gặp ở các luồngâm thanh và hình ảnh không có khả năng thay đổi tạm thời. Nếu RTPTS\_DIFF(i) là 1 hằng số được biểu thị như RTPTS\_DIFF, người ta khuyến nghị những điều sau đây. Giá trị *sample\_delta* trong Decoding Time đến Hộp Mẫu và nếu các phân mảnh movie được sử dụng, giá trị của *sample\_duration* trongHộp Xử lý Phân mảnh Rãnh và các hộp sẽ được gán cho RTPTS\_DIFF, dẫn đến kết quả là nén Decoding time bằng mẫu. CácHộp Xử lý Phân mảnh Rãnh và hộp rtpoffsetTLV không nên sử sụng trong vòng chỉ dẫn tiếp nhận RTP nếu rãnh chỉ dẫn tiếp nhận RTCPđược sử dụng như trong G.3.6. Mặt khác (nếu rãnh chỉ dẫn tiếp nhận RTPkhông được sử dụng) độ dịch chuyển trong hộp rtpoffsetTLV nên đặt về 0.

Khi khả năng thay đổi tạm thời được sử dụng trong luồnghình ảnh, thứ tự truyền và thứ tự phát lại của các gói không được xác định, nhãn thời gian RTP không tăng lên như chức năng của số thứ tự RTP, và RTPTS\_DIFF(i) không liên tục. Tuy nhiên nhãn thời gian RTP điển hình có hành vi liên tục trong thời kì quyết định bởi kích thước GOP(1 cộng với số lượng hình ảnh giữa 2 hình ảnh liên tiếp trong dãy số thứ tự RTP). Ví dụ nếu 2 hình ảnh không tham chiếu được viết bằng mã cho mỗi cặp hình ảnh của tham chiếu như minh họa trong hình G.3 thì kích thước GOP bằng 3. HìnhG.4 minh họa 1 ví dụ của luồng bit có thể thay đổi tạm thời với kích thước GOP bằng 4 (số thứ tự RTP(SN) được bắt đầu từ 0 và gói mỗ*khung hình I* được giả định).



**Hình G.3 – Ví dụ về luồng bit có thời gian linh hoạt với GOP\_size bằng 3.**

(Số chuỗi RTP (SN) bắt đầu bằng 0 và giả thiết có một gói trong mỗi khung hình I. Nhãn thời gian RTP (TS) cũng bắt đầu bằng 0 và thể hiện các đánh dấu thời gian kéo dài trong một khoảng khung hình. Các mũi tên dự đoán nội tại chỉ GOP đầu tiên, các hình trong các GOP khác cũng tương tự).



**Hình G.4 – Ví dụ về một luồng bit linh hoạt có thời gian phân cấp với GOP\_size bằng 4.**

(Số chuỗi RTP (SN) bắt đầu bằng 0 và giả thiết có một gói trong mỗi khung hình I. Nhãn thời gian RTP (TS) cũng bắt đầu bằng 0 và thể hiện các đánh dấu thời gian kéo dài trong một khoảng khung hình).

Sự gia tăng nhãn thời gian RTP gây ra bởi một GOP có nguồn gốc như sau, khi không có vòng lặp ngoài các giá trị nhãn thời gian RTP vượt quá số nguyên 32-bit không dấu giữa mẫu i và i + GOP\_size, Vật chứa:

RTPTS\_GOP\_DIFF(i) = RTPTS(i + GOP\_size) – RTPTS(i) (G.3)

Nếu RTPTS\_GOP\_DIFF (i)là một hằng số bằng RTPTS\_GOP\_DIFF, thì khi không có mẫu i, i + 1,..., i + GOP\_size là một hình ảnh bắt đầu từ một nhóm khép kín của hình ảnh, chẳng hạn như một bức tranh IDR của các luồng H.264/AV. Giá trị của *sample\_delta* trong Decoding Time đến Hộp Mẫu và nếu các phân mảnh movie được sử dụng thì giá trị của *sample\_duration* trong hộp hoặc cácHộp Xử lý Phân mảnh Rãnh được thiết lập vào RTPTS\_GOP\_DIFF / GOP\_size. Hộp rtpoffsetTLV không nên sử dụng cho các hình ảnh ở mức thời gian thấp nhất, nếu RTCP tiếp nhậnrãnh chỉ dẫn được sử dụng (xem G.3.6). Mặt khác (nếu rãnh chỉ dẫn tiếp nhận RTCP không được sử dụng), độ dịch chuyển trong hộp rtpoffsetTLV nên được thiết lập là 0. độ dịch chuyển trong hộp rtpoffsetTLV nên được thiết lập cho hình ảnh ở mức độ thời gian khác để công thức (G.1) được hoàn thành.

Hình G.5 chỉ ra cách đặt thời gian giải mã và độ dịch chuyển được thiết lập cho một luồng bit phân cấp tạm thời có khả năng mở rộng được trình bày trong hình G.4.



**Hình G.5 - Một ví dụ về thiết lập thời gian giải mã (DT) và giá trị của độ dịch chuyển trong hộp rtpoffsetTLV của một luồng bit phân cấp tạm thời khả năng mở rộng với GOP\_size bằng 4.**

(Trong ví dụ này, việc tăng thời gian giải mã giữa các mẫu được thiết lập bằng RTPTS\_GOP\_DIFF / GOP\_size để có một mã hóa nhỏ gọn giải mã thời gian giải mã. Các giá trị độ dịch chuyển trong hộp rtpoffsetTLV được điều chỉnh cho mỗi mẫu để lưu trữ một đại diện của các nhãn thời gian RTP. Để minh họa điều này, các nhãn thời gian RTP và thời gian giải mã được chuẩn hóa được nêu ra.)

Các nhãn thời gian RTP không tuyến tính và định kì được phát hiện từ các gói tin nhận được, và gói tin nhận được các mẫu khác nhau không có cùng thời gian tiếp nhận. Người ta khuyến khích thiết lập giá trị của *sample\_delta* trong Hộp Ánh xạ Thời gian giải mã và Mẫu và nếu phân mảnh movie được sử dụng thì giá trị của *sample\_duration* trong hộp hoặc cácHộp Xử lý Phân mảnh Rãnh sẽ thể hiện thời gian giải mã các gói tin đầu tiên của mẫu. Thời gian giải mã nguồn gốc DT (i) nên bằng với thời gian tiếp nhận các gói tin đầu tiên của mẫu trừ đi thời gian tiếp nhận các gói tin đầu tiên của mẫu nhận được đầu tiên của luồng.

Cần lưu ý rằng thành phần nhãn thời gian không được chỉ định một cách rõ ràng trong mẫu của bất kỳ rãnh chỉ dẫnnào. Do đó, đối vớirãnh chỉ dẫn tiếp nhận RTP, các thành phần nhãn thời gian được suy ra từ các thông tin liên quan đến các nhãn thời gian RTP được chỉ ra trong luồng được lưu trữ. Đối với một rãnh chỉ dẫn tiếp nhận RTPkhông phải là sự kết hợp với rãnh chỉ dẫn tiếp nhận RTP, thời gian thành phần của một gói RTP nhận được suy ra được từ tổng các mẫu thời gian DT(i) và giá trị của các độ dịch chuyển trong hộp rtpoffsetTLV bao gồm cả mẫu này. Đối với một rãnh chỉ dẫn tiếp nhận RTPđược kết hợp với rãnh chỉ dẫn tiếp nhận RTP, thời gian thành phần được suy ra như sau.Để cho các gói RTP nhận được có nhãn thời gian RTP sớm nhất thì thời gian thành phần bằng 0. Bất kì gói RTP còn lại nào cũng có một thành phần thời gian tương đương với nhãn thời gian RTP khác biệt của gói RTP hiện tại và các gói RTP sớm nhấttrong thứ tự trình bày trong G.3.6.3.

**G.3.6 Phần ghi các hoạt động để tạo điều kiện cho việc đồng bộ nội luồng trong việc phát lại**

**G.3.6.1 Giới thiệu chung**

Đồng bộ hình và tiếng, ví dụ: đồng bộ cơ bản xác giữa việc ghi âm các luồngRTP, trong khi phát lại có thể được tối ưu với hai phương pháp sau:

1. RCTP rãnh chỉ dẫn tiếp nhậnđược tạo ra cho mỗi rãnh chỉ dẫn tiếp nhận RTP. Thời gian giữa các đồng hồ nhãn thời gian RTP của các luồngkhác nhau được điều chỉnh vào thời điểm khi màcác tệp tin được phân tích và các luồng đa phương tiện bao gồm trong các tệp tin được giải mã và sử dụng. Việc điều chỉnh lại thời gian được thực hiện tương tự như sẽ được thực hiện cho các luồng RTP được nhận và sử dụng cùng một lúc. Chế độ này hoạt động rất đơn giản cho các đơn vị ghi. Tuy nhiên, truy nhập vào một tệp tin từ một vị trí phát lạicơ bản xác có thể là gây khó khăn hơn, vì nó đòi hỏi tất cả các luồng ghi lại tạithời điểm truy nhập;
2. Các clock drift giữa các luồng RTP đã ghi được thay thế bằng cách thay đổi các nhãn thời gian RTP của một hoặc nhiều luồng được ghi lại. Chế độ hoạt động này yêu cầu xử lý các báo cáo của bên gửi RTCP tại thời điểm được ghi. Tuy nhiên, đối với các thiết bị đọc thì đơn giản hơn.

Đơn vị ghi nên sử dụng hộp đồng bộ nhãn thời gian [9.4.1.2] để chỉ ra phương pháp tiếp cận đồng bộ hình và tiếng đã được sử dụng. Hộp Đồng bộ Nhãn thời gian bao gồm *timestamp\_sync*. Ở đây, *timestamp\_sync* bằng 1 chỉ ra rằng thiết bị đọc nên sử dụng rãnh chỉ dẫn tiếp nhận RTCPcho đồng bộ hình và tiếng. Còn *timestamp\_sync* bằng 2 chỉ ra rằng thiết bị đọcnên sử dụng thời gian thành phần cho quá trình đồng bộ hình và tiếng.

Một số quá trình thực hiện có thể tạo các rãnh chỉ dẫn tiếp nhận RTCP trước tiên trong khi ghi thời gian thực và sau đó bù việc trôi thời gian băng cách thay đổi các nhãn thời gian RTP như một bước tiền xử lý off-line.

**G.3.6.2 Tạo điều kiện cho đồng bộ hình và tiếng dựa trên báo cáo của bên gửi RTCP**

Một đơn vị ghi lưu trữ tất cả các báo cáo của bên gửi RTCP cho một luồng RTP đặc biệt như mẫu tương ứng trong rãnh chỉ dẫn tiếp nhận RTP.

**G.3.6.3 Bù trôi thời gian của đồng hồ trong nhãn thời gian**

Người ta không khuyến khích thay đổi nhãn thời gian RTP của các luồng âm thanh được ghi lại. Mỗi sự thay đổi như vậy sẽ gây ra một khoảng thời gian thay đổi âm thanh trong thiết bị đọc,

Đại diện ghi lại của các nhãn thời gian RTP trong hình ảnh và các luồngâm thanh khác nên được sửa đổi bằng các sử dụng các thủ tục sau đây.

1. Đầu tiên, nhãn thời gian đồng hồ của một khung hình hình ảnh được gửi từ các nhãn thời gian RTP tương ứng với khung hình Ihình ảnh như tổng của các nhãn thời gian đồng hồ của khung hình hình ảnh trước và sự khác biệt của các nhãn thời gian của khung hình hình ảnh hiện tại trong các đơn vị của nhãn thời gian đồng hồ.
2. Thứ hai, thời gian xem lại b cho khung hình hình ảnh vào thời gian đồng hồ được gửi dựa trên báo cáo của bên gửi RTCP. Nếu không có báo cáo của bên gửi RTCPcơ bản xác cho biết thời gian đồng hộ cho khung hình hình ảnh có sẵn, thì thời gian đồng hồ có thể được ngoại suy giả định rằng tốc độ đồng hồ nhãn thời gian RTP và thời gian gửi trong báo cáo của bên gửi RTCP vẫn không thay đổi.
3. Thứ 3, dựa trên các báo cáo của bên gửi RTCP cho âm thanh, các nhãn thời gian RTP âm thanh được chạy cùng một lúc với các khung hình hình ảnh tại thời điểm b của thời gian đồng hồ được lấy. Đây không phải là khung âm thanh có cơ bản xác nhãn thời gian RTP xuất phát. Các nhãn thời gian đồng hồ c của một mẫu âm thanh được tính từ nguồn gốc nhãn thời gian âm thanh RTP là tổng của các nhãn thời gian đồng hồ của khung âm thanh trước và sự khác biệt của các nhãn thời gian RTP của nhãn thời gian âm thanh RTP có nguồn gốc và nhãn thời gian RTP của khung âm thanh trước.

Sự khác biệt giữa a và c, nếu có, nên bù cho hình ảnh nhãn thời gian RTP trong tệp tin. Trong thực tế, cách dễ nhất có thể để thêm sự khác biệt cho các trường độ dịch chuyểnlà trong hộp rtpoffsetTLV được minh họa ở hình G.6.Trong các trường hợp khác, việc viết lại các Hộp Ánh xạ Mẫu và Thời gian Giải mã vàHộp Xử lý Phân mảnh Rãnh (nếu có) có thể sẽ khó khăn hơn để thực hiện, bởi vì mã hóa mẫu thời gian được thực hiện bằng sự kết hợp của cách đếm mẫu, thời gian mẫu và có thể yêu cầu nhiều không gian lưu trữ hơn.



**Hình G.6 – Ví dụ về việc điều chỉnh đúng đồng bộ hình và tiếng trong biểu diễn nhãn thời gian RTP.**

**G.3.7 Biểu diễn thời gian tiếp nhận**

Theo qui định tại 9.4.1.4, thời gian tiếp nhận một gói tin được chỉ định bằng tổng thời gian giải mã của mẫu có chứa các gói tin và giá trị của relative\_time của cấu trúc RTPpacket của gói tin.

Thời gian tiếp nhận các gói tin RTP đã nhận sớm nhất nên là 0, và thời gian tiếp nhận tất cả các gói tiếp theo nên liên quan đến thời gian tiếp nhận của các gói tin RTP nhận được sớm nhất.

Nguồn đồng hồ cho thời gian tiếp nhận là không xác định và có thể xác định, ví dụ, đồng hồ của bên nhận. Nếu khoảng thời gian tiếp nhận của một rãnh chỉ dẫn tiếp nhậntrùng lặp hoàn toàn hoặc 1 phần với khoảng thời gian tiếp nhận của các rãnh chỉ dẫn tiếp nhậnkhác, thì các clock cho các rãnh chỉ dẫnnày sẽ trở nên giống nhau.

Thời gian tiếp nhận của một gói tin phải tương ứng với thời gian của giao thức ngăn lớp RTP, UDP thường, và kết quả đầu ra của gói tin bên dưới.

**G.3.8 Tạo các mẫu đa phương tiện**

Các mẫu đa phương tiện được tạo ra từ các gói RTP đã nhận theo dữ liệu RTP và cơ bản RTP đó. Tuy nhiên, hầu hết các đa phương tiện tiêu chuẩn mã hóa chỉ ghi chú sự giải mã các luồnglỗi và do đó cần đảm bảo rằng nội dung trong các đa phương tiện có thể được giải mã một cách cơ bản xác bởi bất kì bộ giải mã tiêu chuẩn nào. Do đó,việc xử lí lỗi đa phương tiện đòi hỏi 2 bước sau: phát hiện các lỗi đa phương tiện và suy luận các mẫu có thể được giải mã 1 cách cơ bản xác. Các bước này được miêu tả ở đoạn tiếp theo.

Các gói tin bị mất RTP có thể được phát hiện từ 1 trong các giá trị số RTP. Các gói tin RTP chứa đựng một số lỗi thì thường không được chuyển tiếp đến các ứng dụng như kiểm tra lỗi UDP và các gói tin bị kiểu bỏ trong ngăn xếp giao thức của bên nhận. Do đó, các gói tin chứa một số lỗi thì thường được coi như mất gói tin của bên nhận.

Các mẫu đa phương tiện có thể được giải mã một cách cơ bản xác phụ thuộc vào định dạng mã hóa đa phương tiện và do đó không được miêu tả ở đây trong chi tiết. Nói chung, việc dự đoán inter-sample là yếu hoặc không tồn tại trong các định dạng mã hóa âm thanh, trong khi hầu hết các định dạng hình ảnh mã hóa sử dụng dự đoán này.Do đó, một mẫu bị mất trong nhiều định dạng âm thanh thường có thể được thay thế bằng một mẫu âm thanh im lặng hoặc mẫu che dấu lỗi âm thanh. Nó cần được phân tích liệu có sự mất mát của một movie liên quan đến một hình ảnh không tham chiếu nào không hay tổng quát hơn là hình ảnh tham khảo, trong đó mức độ của hệ thống phân cấp cókhả năng mở rộng tổn thất thời gian xảy ra. Sau đó, nên đưa ra kết luận hình ảnh nào có thể không được giải mã cơ bản xác. Ví dụ, một sự mất mát của một hình ảnh không tham chiếu không ảnh hưởng đến việc giải mã của bất kỳ hình ảnh nào khác, trong khi một sự mất mát của một hình ảnh tham chiếu trong mức thời gian cơ sở thường ảnh hưởng đến tất cả các hình ảnh cho đến khi hình ảnh tiếp theo cho truy nhập ngẫu nhiên, chẳng hạn như một bức ảnh IDR trong H.264/AVC.

**G.3.9 Tạo các mẫu chỉ dẫntham chiếu tới các mẫu đa phương tiện**

Các mẫu đa phương tiện được tạo ra từ các gói RTP nhận như được giải thích trong G.3.8. rãnh chỉ dẫn tiếp nhận RTP được tạo ra như được giải thích trong G.3.4, nhưng nội dung của cấu trúc RTPpacket phụ thuộc vào sự tồn tại của mẫu đa phương tiện tương ứng như sau.

Nếu phần tải dữ liệu của gói RTP nhận được biểu diễn trong một rãnh đa phương tiện, các tham chiếu theo thiết lập để chỉ ra các đa phương tiện và bao gồm các gói tải dữ liệu. Nó không được khuyến khích sử dụng như là 1 bản sao của gói tải dữ liệu trong phần extradata của mẫu RTP nhận được để tiết kiệm không gian lưu trữ và khiến cho các hoạt động chỉnh sửa tệp tin trở nên dễ dàng hơn.

Nếu gói tải dữ liệu của gói RTP nhận được không được đại diện cho rãnh đa phương tiện, thì sự thể hiện của cấu trúc gói RTP được tạo ra như đã giải thích ở G.3.4

**G.4 Phát các luồng RTPđã ghi**

**G.4.1 Giới thiệu**

Trong mục này sẽ miêu tả hoạt động cần thiết đối với phát lại của các tệp tin có chứa luồng RTP được ghi. Nó được tổ chức như sau:

* Trước khi các luồng RTP có thể được sử dụng, nội dung của các tệp tin cần được phân tích. Đặc biệt, rãnh thay thế đại diện cho các luồngđa phương tiện giống nhau nên được xác định và một trong những rãnh đó được chọn để giải mã và phát lại. Định dạngmã hóa phải được phát hiện để nó có thể được giải mã bởi thiết bị đọc hay không. Các hoạt động được miêu tả chi tiết trong G.4.2.
* Nếu một rãnh chỉ dẫn tiếp nhận RTPđang được xử lý, cần Lưu ý một số điều như trong mục G.4.3. Ví dụ, các gói bị mất cần được phát hiện và xử lý một cách thích hợp.
* Việc đồng bộ hóa của các mẫu đa phương tiện giải mã nên được xử lý đúng như miêu tả trong G.4.4.
* Nếu các luồng RTP lưu trữ trong một tệp tin được truy nhập từ một vị trí khác so với nơi bắt đầu của các luồng này, thì các inter-stream đồng bộ hóa và giải mã khởi tạo là cần thiết và được miêu tả trong G.4.5.

**G.4.2 Sự chuẩn bị cho quá trình phát lại**

Trong giai đoạnchuẩn bị của quá trình phát lại, thiết bị đọc chọn rãnh được sử dụng. Đầu tiên, việc theo dõi cơ bản của tệp tin được phân tích. Các rãnh được phân kiểu theo các nhóm mà chúng thuộc về. rãnh thuộc cùng 1 nhóm được chỉ định bởi cùng 1 giá trị của *alternate\_group* trong hộp mào đầu. Một rãnh từ mỗi nhóm thay thế được chọn để phát lại như sau.

Nếu có một rãnh chỉ dẫn tiếp nhận RTP trong nhóm thay thế, nó sẽ được dùng để thực hiện việc phát lại, bởi vì nó chứa một đại diện toàn bộ của luồng RTP nhận được, không giống như các rãnh đa phương tiệnđược bắt nguồn từ các luồng RTP nhận được, cái mà có thể sử dụng tập hợp con của các gói RTP nhận được để giải mã bằng bất kỳ bộ giải mã tiêu chuẩnnào.

Sự tương thích giữa thiết bị đọc và trackđược chọn phải được đảm bảo. Ví dụ, cần kiểm tra xem liệu các mật mã, hồ sơ, và mức độ sử dụng trong rãnh có cân bằng với khả năng hỗ trợ của thiết bị đọc hay không.

Mật mã, hồ sơ, mức độ sử dụng của luồng bit được mật mã hóa trong một rãnh chỉ dẫn tiếp nhận RTPcó thể suy ra từ miêu tả SDP của luồng RTP. Các miêu tả SDP được lưu trữ trong các chỉ số theo dõi movie-level. Nếu SDP không thay đổi trong suốt các tệp tin, nó có thể được lưu trữ bổ sung như thông tin SDP movie và rãnh thông tin SDP trong Hộp Dữ liệu Người dùng. Nếu rãnh thông tin SDP được biểu thị, nó có thể được phân tích để tìm ra mật mã, hồ sơ và mức độ được sử dụng cho luồng bit có chứa trong rãnh chỉ dẫn tiếp nhận RTP. Nếu thông tin SDP movie hoặcrãnh thông tin SDP không được biểu thị, chỉ số theo dõi movie-level bị bỏ qua để tìm và phân tích mỗi chỉ số SDP, do đó các mật mã, hồ sơ, và mức độ sẽ được sử dụng cho luồng bit có chứa trong rãnh chỉ dẫn tiếp nhận RTP.

Nếu rãnh chỉ dẫn tiếp nhận RTPkhông tồn tại ở một nhóm khác, một hoặc nhiều mục mẫu của rãnh đa phương tiệntrong nhóm thay thế thì nó nên được kiểm tra để tìm ra thiết bị đọc có khả năng hỗ trợ.

**G.4.3 Giải mã một mẫu trong một rãnh chỉ dẫn tiếp nhận RTP**

Các gói tin RTP ban đầu có thể được dựng lại từ một mẫu rãnh chỉ dẫn tiếp nhận RTPbằng cách tạo ra các tiêu đề gói tin RTP từ các cấu trúc RTPpacket và bằng cách giải quyết các các cấu trúc RTPpacket. Vì vậy, một cách để tệp tinthiết bị đọc xử lý rãnh chỉ dẫn tiếp nhận RTPlà tái tạo các luồng, gói tin đã nhận được và xử lý các luồng, gói tin đã được tái tạo lại đó như thể là chúng mới được nhận.

*Trường* relative\_time bao gồm trong cấu trúc RTPpacket có thể được sử dụng để sắp xếp việc chèn các gói tin vào bộ đệm cho bên nhận RTP. Tuy nhiên, việc được khuyến khích nhiều hơn là thay đổi quá trình giải mã các luồng RTP, bằng cách như vậy các bộ đệm đầu ra bộ giải mã được lưu trữ một cách đầy đủ nhất có thể và để tránh gián đoạn hoặc việc giật phát lại gây ra bởi các gói tin sau hoặc các vấn đề thường xuyên ở thời gian giải mã trong các hệ thống chạy các quá trình khác bên cạnh thiết bị đọc.

Tổn thất gói được phát hiện từ những khoảng trống trong dãy số RTP. Phản ứng với tổn thất gói phụ thuộc vào bộ giải mã cụ thể và cũng có thể phụ thuộc vào người dùng.

**G.4.4 Đồng bộ hình và tiếng**

Các bước sau đây được yêu cầu để đạt được cơ bản xác đồng bộ hóa giữa các luồng:

1. Đồng bộ hóa inter-stream vào thời điểm bắt đầu của quá trình phát lại.

Vị trí bắt đầu của trục thời gian đa phương tiện của một rãnh có thể được dịch chuyển trong trục thời gian movie của tệp tin như miêu tả trong hai dưới đây

Đối với một rãnh đa phương tiệnvà một rãnh chỉ dẫn tiếp nhận RTPkhông được liên kết với một rãnh chỉ dẫn tiếp nhận RTCP, một Hộp Danh sách Biên tập được sử dụng để thay đổi vị trí bắt đầu của trục thời gian đa phương tiện như miêu tả trong G.3.2. Các trục thời gian đa phương tiện của các rãnh đã lựa chọn để phát lại được sắp xếp tới trục thời gian movie bằng cách phân tích các Hộp Danh sách Biên tập của các rãnh, nếu có. Việc phát lại từng rãnh đa phương tiệnvà từng rãnh chỉ dẫn tiếp nhận RTPkhông được liên kết với một rãnh chỉ dẫn tiếp nhận RTPbắt đầu ở vị trí trục thời gian movie được chỉ ra trong Hộp Danh sách Biên tập của rãnh hoặc từ lúc bắt đầu trục thời gian movie.

Đối với rãnh chỉ dẫn tiếp nhận RTPcó liên quan với rãnh chỉ dẫn tiếp nhận RTCPtương ứng, thì việc chuyển đổi vị trí bắt đầu trục thời gian đa phương tiện trong trục thời gian movie được suy ra như sau. Các trục thời gian đa phương tiện của các rãnh chỉ dẫn tiếp nhận RTPchứa các gói tin RTP sớm nhất (thời gian bên gửiđồng hồ) trong số tất cả các rãnh rãnh chỉ dẫn tiếp nhận RTPthì không được thay đổi trong trục thời gian movie (ví dụ, bắt đầu tại thời điểm 0 trên trục thời gian movie). Thời gian bắt đầu của trục thời gian đa phương tiện của bất kỳ rãnh chỉ dẫn tiếp nhận RTPtheo dõi khác bằng sự khác biệt về nhãn thời gian của các gói tin RTP sớm nhất trong các rãnh hiện tại và theo dõi có chứa các gói RTP sớm nhất trong số tất cả các rãnh chỉ dẫn tiếp nhận RTP.

1. Xây dựng lại nhãn thời gian RTP và thành phần thời gian trên trục thời gian đa phương tiện(G.3.5).
2. Sửa chữa nhãn thời gianvà thành phần thời gian dựa trên báo cáo của bên gửi RTP, nếu như rãnh chỉ dẫn tiếp nhận RTCPđược sử dụng.

Việc điều chỉnh này được thực hiện tương tự như miêu tả ở G.3.6.3. Tuy nhiên, thay vì thêm sự khác biệt giữa lần a và c vào các đại diện của nhãn thời gian RTP trong các tệp tin, thì sự khác biệt này được thêm vào trong quá trình phát lại đối với thời gian hiển thị của các khung hình hình ảnh trên trục thời gian movie.

1. Tốc độ đầu ra của các mẫu đa phương tiện giải mã.

Người ta khuyến khích nên sử dụng một chương trình ghi lại tốc độ đồng hồ của thiết bị đọc và dùng đồng hồ phát lại như đồng hồ của thiết bị đọc. Việc phát lại âm thanh này được bố trí liên tục ở tần số mẫu tự nhiên của các tín hiệu âm thanh. Một đồng hồ hiển thị của các thiết bị đọc chạy ở tốc độ của âm thanh phát lại, tức là giá trị của nó luôn bằng với (số lượng các mẫu âm thanh không nén thường gặp nhất đã được sử dụng) ×(tần số tự nhiên của tín hiệu âm thanh). Sự phát lại các được đồng bộ hóa với đồng hồ hiển thị của thiết bị đọc. Nói cách khác, khi đồng hồ hiển thị của thiết bị đọc giao với thời gian thành phần của một mẫu hình ảnh trên trục thời gian movie, các mẫuhình ảnh này sẽ được phát. Chỉ khi một tệp tin được đồng thời ghi lại và phát lại nếu đồng hồ của bên nhận chạy nhanh hơn so với đồng hồ bên gửi, tốc độ phát lại theo tỉ lệ của đồng hồ bên gửi có thể không đượckhuyến khích và việc đồng bộ hóa tốc độ đồng hồ bên gửi có thể được thực hiện như sau.

Tốc độ của thời gian bên gửi được phục hồi lại bằng cách tạo ra mối quan hệ giữa các lầntiếp nhận (theo đồng hồ máy thu) và các nhãn thời gian đồng hồ tương ứng của bên gửi được dựng lại từ báo cáo của bên gửi RTCP. Người ta khuyên nên sử dụng đồng hồ phát lại âm thanh như đồng hồ máy thu. Mối quan hệ giữa thời gian tiếp nhận và nhãn thời gian tương ứng của bên gửi nên được chia trung bình trên một số lượng lớn các gói tin nhận được. Một yếu tố nữa là kết quả của các trung bình nhân giữa thời gian tiếp nhận và nhãn thời gian tương ứng của bên gửi.

Thời gian hiển thị trên trục thời gian của đồng hồ thu được lấy từ từng mẫu. Nếu rãnh chỉ dẫn tiếp nhận RTCPđang được sử dụng, thời gian hiển thị là thành phần thời gian của mẫu trên trục thời gian movie, bao gồm việc sửa chữa clock drift như miêu tả ở 3 bước trên. Nếu rãnh chỉ dẫn tiếp nhận RTCPkhông được sử dụng, thời gian hiển thị là thời gian thành phần trực tiếp của mẫu trên trục thời gian movie. Vì vậy, chỉ dùng cho mục đích phát lại, thời gian hiển thị của các mẫu trong tất cả các rãnh đã phát nên được nhân bởi hệ số nhân khoảng thời gian.

Thời gian kéo dài của tín hiệu nên được thực hiện cho phù hợp. Các mẫu được phát tại cơ bản các thời điểm trình bày. Trong thực tế, các yếu tố nhân của khoảng thời gian và việc ánh xạ từ trục thời gian RTP lênđồng hồ của bên gửi (bước 3 ở trên) có thể được thực hiện như một hoạt động đơn lẻ.

**G.4.5 Truy nhập ngẫu nhiên**

Truy nhập ngẫu nhiên đề cập đến một truy nhập phi tuyến tính với luồng đa phương tiện đại diện trong tệp tin. Nói cách khác, trong một hoạt động truy nhập ngẫu nhiên, các tệp tin được truy nhập từ mẫu khác hơn so với các mẫu đã được phát trước đây hoặc các tệp tin ban đầu được truy nhập từ một vị trí nào đó không phải là điểm khởi đầu của trục thời gian movie.

Người ta khuyến khích rằng nên cung cấp cho người sử dụng các chức năng truy nhập ngẫu nhiên liên quan đến trục thời gian movie của tệp tin chứ không phải là bất kì mốc thời gian nào khác, ví dụ như thời gian đồng hồbên gửi. Bằng việc sử dụng trục thời gian movie làm cơ sở, số lượng các bước cho một hoạt động ngẫu nhiên được giữ ở mức thấp.

Đầu tiên, các khung đa phương tiện được bắt nguồn từ một vị trí mong muốn (hoặc gần nhất với khung hình đó, nếu không có vị trí mong muốn truy nhập ngẫu nhiên cơ bản xác). Trong trường hợp của rãnh đa phương tiện, rãnh chỉ dẫn tiếp nhận RTPcho âm thanh, và bất cứ rãnh chỉ dẫn tiếp nhận RTPcó *timestamp\_sync* bằng 2, khung đa phương tiện gần nhất với vị trí mong muốn truy nhập ngẫu nhiên có thể được gửi đi trực tiếp dựa trên nhãn thời gian thành phần (trên trục thời gian đa phương tiện) chuyển đi bởi vị trí đầu tiên khi xuất phát cho thấy trong Hộp Danh sách Biên tập. Trong trường hợp rãnh chỉ dẫn tiếp nhận RTPkhông âm thanh có *timestamp\_sync* bằng 1 (chỉ sử dụng cho rãnh chỉ dẫn tiếp nhận RTCP),các mẫu phải được gửi đi như trong G.4.4, cho đến khi thời gian hiển thị gần nhất với vị trí mong muốn truy nhập ngẫu nhiên được tìm thấy.

Thứ hai, việc giải mã nhiều luồng bit đa phương tiện chỉ có thể bắt đầu từ một kiểu hình cụ thể, chẳng hạn như một hình ảnh IDR của H.264/AVC. Do đó, việc triển khai thiết bị đọc có thể có nhiều cách tiếp cận khác nhau, bao gồm các cách sau:

1. Khám phá các khung hình gần nhất tại hoặc trước vị trí mong muốn truy nhập ngẫu nhiên mà từ đó việc giải mã có thể bắt đầu, bắt đầu giải mã từ khung đó, và bắt đầu gửi đi từ các điểm truy nhập ngẫu nhiên mong muốn. Cách tiếp cận này có thể bao gồm một số xử lý chậm trễ trước khi việc gửi đi được bắt đầu;
2. Bắt đầu giải mã và gửi đi tại hoặc sau các điểm truy nhập mong muốn ngẫu nhiên bằng cách sử dụng khung hình đầu tiên mà từ đó việc giải mã được bắt đầu. Thông thường, việc phát lại âm thanh sẽ bắt đầu sớm hơn việc phát lạihình ảnh, nhưng sự xử lý chậm trễ trước khi gửi đi thì nhỏ hơn so với các tùy chọn trước đó.

**G.5 Gửi lại các luồng RTP đẫ ghi**

**G.5.1 Giới thiệu**

Việc gửi lại các luồng RTP đã được ghi trước đó vào trong một tệp tin là một hoạt động đáng giá. Ví dụ, nếu các luồng RTP được nhận thông qua dịch vụ truyền hình hoặc trực tiếp và được ghi vào một tệp tin, nó có thể ghi lại khi gửi lại cho họ từ một thiết bị đến thiết bị khác trong một không gian gia đình sử dụng mạng kết nối WLAN. Khoản này cung cấp các khuyến nghị về việc gửi lại các luồng RTP được ghi lại.

Một hệ thống thông tin liên lạc dựa trên RTP bao gồm một thiết bị đầu nguồn (hay còn gọi là một bên gửi) và một điểm đến thiết bị cuối nguồn (còn gọi là một bộ thu) và có thể chứa một hoặc nhiều bộ trộn và phiên dịch. Bên gửivà bên nhận là những thiết bị đầu cuối của phiên RTP và RTCP. Cách hoạt động của phiên dịch RTP và máy trộn được qui định tại RFC 3550 và làm rõ trong RFC 5117. Nhìn chung, các đơn vị ghi nhận các luồng RTP và lưu trữ chúng vào một tệp tin hoạt động như một thiết bị đầu cuối, và một đơn vị gửi-đọc lại lưu trữ các luồng RTP từ một tệp tin và gửi chúng đi hoạt động như một nguồn. Thông thường, payload của luồng RTP gửi lại không được sửa đổi, điều này khiến cho sự kết hợp của đơn vị ghi và đơn vị gửi lại hoạt động tương tự như bộ chuyển đổi truyền tải như miêu tả trong RFC 5117. Tuy nhiên, đặc tính thiết yếu của một bộ chuyển đổi là cho bên nhận không thể phát hiện sự hiện diện của nó. Do đó, sự kết hợp của một đơn vị ghi âm và một đơn vị gửi lại không thể hoạt động như một bộ chuyển đổi truyền tải, trừ khi việc gửi lại xảy ra đồng thời với việc ghi của các luồng gốc. Trường hợp này được coi là hiếm hoi, các cuộc thảo luận tại khoản này liên quan đến một đơn vị ghi âm như một điểm đến chấm dứt RTP và RTCP phiên bản gốc và một đơn vị tái gửi như một nguồn RTP và RTCP phiên bản mới.

Phần này được tổ chức như sau:

* G.5.2 bao gồm các khuyến nghị về việc làm thế nào để soạn các gói tin RTP từ rãnh chỉ dẫn tiếp nhận RTPvà làm thế nào để sắp xếp việc truyền tải các gói RTP;
* G.5.3 Thảo luận gói RTCP nên được phát ra như thế nào và các gói RTCP đã nhận được sản xuất như thế nào.

**G.5.2 Gửi lại các gói RTP**

Các gói nên được thiết lập và truyền đi như sau:

Payload của các gói được khuyến nghị xây dựng như trong rãnh chỉ dẫn tiếp nhận. Ví dụ, dữ liệu của gói nên xác định với bên nhận, nếu không cần 1 kích thước gói khác cho mạng lưới mà các gói được gửi lại.

* Giá trị mào đầu của các gói RTP nên giữ giống như trong cấu trúcRTPpacket tương ứng, trừ các trường hợp sau:
* Độ dịch chuyển nhãn thời gian RTP ban đầu và độ dịch chuyển chuỗi số RTP nên được chọn ngẫu nhiênbất chấp giá trị lưu trữ trong *trường*dịch chuyển của hộp ‘tsro’ trongrãnh chỉ dẫn tiếp nhậnmẫu được đề cập, hay là giá trị của miền RTPequenceseed của cấu trúc RTPpacket của bất kì gói nào của RTPrãnh chỉ dẫn tiếp nhậntương đương.
* Giá trị của miền nhãn thời giantrong RTP là tổng của độ dịch chuyển ban đầu ngẫu nhiên, giá trị của độ dịch chuyển trong cấu trúc RTPpacket, và thời gian giải mã của mẫu RTP tương ứng. Nếu tổng vượt quá tối đa số nguyên không dấu 32 bit, nên được đóng lại.
* Giá trị liên quan của chuỗi số RTP bằng những cái được thu trong giá trị của miền RTPequenceseed. Do đó nếu có 1 gói trong luồng đã thu bị mất, luồng đã gửi lại cũng có một khoảng cách tương tương trong chuỗi số RTP, và bên nhận có thể suy luận ra 1 gói đã bị mất.
* Giá trị của miền đếm CSRC luôn luôn bằng 0, bởi vì không có nguồn đóng góp nào của phần đã được thu RTP trước chủ động thay đổi các luồng để phần RTP của các luồng được gửi lại. Khoảng xác định nguồn (cho cả SSRC và CSRC) là phần đặc thù. Do đó danh sách CSRC của bản ghi nhãn RTP nên để trống bất chấp giá trị CSRC có thể được lưu trữ với các luồng đã nhận (bao gồm hộp ReceivedCSRC TLV trong cấu trúc RTPpacket).
* Giá trị của miền payload có thể được lựa chọn phụ thuộc vào sơ đồ dùng làm dấu hiệu đang được sử dụng.
* Giá trị của miền SSRC nên được chọn ngẫu nhiên nhưng xung đột có thể xảy ra nên được giải quyết như đặc thù trong RFC 3550. Giá trị SSRC của 1 luồng nhận có thể được lưu trữ trong hộp receivedSsrccủa rãnh chỉ dẫn tiếp nhậnmẫu đã đề cập, nhưng nên bỏ qua khi luồng được gửi lại.
* Phần mở rộng mào đầu RTP đã được ghi được lưu trong rtphdrext TLV trong cấu trúc RTPpacket, nếu có,chỉ nên gửi lại nếu đơn vị gửi lại có thể xác minh rằng nó có giá trị đối với luồng gửi lại. Nếu đơn vị gửi lại không thể phân tách ngữ nghĩa của việc mở rộng bản ghi nhãn RTP đã được thu thì không nên gửi lại.

Thời gian nhận của 1 gói tin được tính bằng tổng thời gian giải mã của mẫu chỉ dẫn tiếp nhận RTP chứa gói tin đó và giá trị của relative\_time của cấu trúc RTPpacket, bằng thời gian truyền gói tin với sai lệchdo trễ truyền dẫn và trễ xử lý trong chồng giao thức của bên nhận. Sự sai lệch của các gói liền kề có thể không bằng nhau do jitter trễ truyền dẫn và độ trễ xử lý không đồng đều. Thêm vào đó, chồng giao thức sử dụng khi nhận luồng thông tin đó có thể khác với chồng giao thức được dùng cho việc gửi lại luồng này. Do những lý do đó, thời gian nhận thường không áp dụng được cho các gói tin phải truyền lại. Trong mọi trường hợp, gói tin truyền lại nên được xác nhận rằng luồng tin truyền lại đó tương tích với mô hình bộ đệm được sử dụng, nếu việc nhận luồng và gửi lại luồng tương tự nhau thì thời gian nhận có thể được dùng làm cơ sở cho việc lập lịch truyền gói tin. Bộ phận gửi lại cần kiểu bỏ jitter trễ truyền lan trong luồng. Nếu bộ phận gửi lại không thể đưa ra được kết luận rằng môi trường mạng và các chồng giao thức được dùng khi nhận luồng và khi gửi lại luồng giống nhau hoặc không chắc chắn kiểu lập lịch gói tin phù hợp thì nó sẽ sử dụng thời gian giải mã làm cơ sở cho việc lập lịch.

**G.5.3 Xử lý RTCP**

Các báo cáo của bên gửi RTCP và các bản tin RTCP khác được phục hồi theo sự ràng buộc đặc thù trong RFC 3550hơn là trực tiếp sử dụng bản tin RTCP được ghitrong các rãnh chỉ dẫn tiếp nhận RTCP.

Một báo cáo bên gửi RTCP chứa thời gian đồng hồ khi báo cáo được gửi và nhãn thời gian RTP tương ứng cùng lúc với thời gian đồng hồ hiển thị. Nhãn thời gian RTP cho một báo cáo bên gửi RTCP được tạo ra như sau:thời gian biểu diễn trên 1 trục thời gian của đồng hồ tham chiếu được lấy ra từ đồng hồ mẫu của đơn vị gửi lại, bắt đầu từ 0 vào lúc bắt đầu của phiên làm việc. Mẫu tương đương thời gian đồng hồ hiển thị có thể không tồn tại trong rãnh chỉ dẫn tiếp nhận RTPphản hồi, bởi vì khilấy mẫu của các mẫu trong rãnh chỉ dẫn tiếp nhận RTCPcó thể không phù hợp với lúctruyền đi của báo cáo bên gửi RTCP. Tuy nhiên như đã hướng dẫn ở RFC 3550, nhãn thời gian RTP được lấy ra nếu có 1 mẫu trong luồng RTP tương ứng với thời gian đồng hồ hiển thị thì nhãn thời gianRTP của 1 báo cáo của bên gửi nên được nội suy tuyến tính từ nhãn thời gian RTP của mẫu ngay trướcvà sau khi đồng hồ hiển thị trong báo cáo bên gửi RTCP. Để kết luận là mẫu đi ngay trước hoặc sau khi đồng hồ hiển thị trong báo cáo bên gửi RTCP, thời gian biểu diễn trên trục thời gian của đồng hồ tham chiếu nên lấy cho đến khi tìm ra mẫu gần nhất. Nếu rãnh chỉ dẫn tiếp nhận RTPđang được gửi lại, thời gian biểu diễn là thời gian tổng hợp của mẫu trên luồngtrục thời gian movie, cũng bao gồm đồng hồ theo chiều hướng đúng như miêu tả ở bước 3 của G.4.4. Nếu rãnh chỉ dẫn tiếp nhận RTCPkhông được biểu thị, thời gian biểu diễn trực tiếp bằng thời gian tổng hợp của mẫu trên luồngtrục thời gian movie.

Khi xử lý các báo cáo nhận RTCP, nên để ý là số lũy tích chồng chất đã báo cáo của gói bị mất cũng bao gồm các gói chưa được gửi mà không bao giờ nhận được tương ứng với khoảng trống trong chuỗi số RTP trong rãnh chỉ dẫn tiếp nhận RTP. Nên thực hiện mọi việc quản lí quá tải, truyền lại hoặc các phương pháp phục hồi các gói đã bị mất.

# Phụ lục H (Quy định) Các điểm truy nhập luồng

**H.1 Giới thiệu**

Phụ lục này định nghĩa điểm truy nhậpluồng (SAP) và định rõ 6 kiểu SAP.

Một điểm truy nhậpluồng cho phép truy nhập ngẫu nhiên mộtthực thể chứamột hay nhiều luồng đa phương tiện. Một thực thể chứa có thể chứa nhiều hơn mộtluồng đa phương tiện, mỗi luồng là một phiên bản mã hóa của đa phương tiện liên tục của kiểuđa phương tiện nào đó. Mỗi SAP là một vị trí trong một thực thể chứa cho phép bộ phận phát lại của mộtluồng đa phương tiện xác định được bắt đầu sử dụng a) chỉ thông tin chứa trong thực thể chứa bắt đầu từ vị trí đó trở đi và b) dữ liệu khởi tạo có thể từ các phần khác của thực thể chứa hoặc là có sẵn bên ngoài. Các đặc điểm kĩ thuật nhận được nên được chi tiết hóa nếu cần một dữ liệu khởi tạo để truy nhập vào thực thể chứa ở một điểm truy nhập theo luồng, và làm thế nào dữ liệu khởi tạo có thể được truy nhập.

**H.2 Các đặc tính của điểm truy nhập theo luồng (SAP)**

Với mỗi SAP, các đặc tính ISAP, TSAP, ISAU, TDEC, TEPT và TPTF được xác định và định nghĩa như sau:

* TSAP là thời gian biểu diễn sớm nhất của bất kì đơn vị truy nhập nào của luồng đa phương tiện như là tất cả các đơn vị truy nhập của luồng đa phương tiện với thời gian biểu diễn lớn hơn hoặc bằng TSAP, mà có thể giải mã cơ bản xácsử dụng luồng bit bắt đầu ở ISAP và không có dữ liệu nào trước ISAP.
* ISAP là vị trí lớn nhất trong luồng bit như là tất cả các đơn vị truy nhập của luồng đa phương tiệnvới thời gian biểu diễn lớn hơn hoặc bằng TSAP, mà có thể giải mã sử dụng dữ liệu luồng bit bắt đầu ở ISAP và không dữ liệu nào trước ISAP.
* ISAU là vị trí bắt đầu trong luồng bit của đơn vị truy nhập cuối cùng trong thứ tự giải mã trong luồng đa phương tiện như là tất cả các đơn vị truy nhập của luồng đa phương tiện, với thời gian biểu diễn lớn hơn hoặc bằng TSAP, có thể giải mã cơ bản xác sử dụng đơn vị truy nhập cuối cùng và các đơn vị truy nhập theo sau trong thứ tự giải mã, không có đơn vị truy nhập nào sớm hơn trong thứ tự giải mã.

LƯU Ý:ISAU luôn luôn lớn hơn hoặc bằng ISAP.

* TDEC là thời gian biểu diễn sớm nhất của bất kì đơn vị truy nhập nào của luồng đa phương tiện mà có thể được giải mã cơ bản xác sử dụng dữ liệu trong luồng bit bắt đầu ở ISAU và không có dữ liệu nào trước ISAU.
* TEPT là thời gian biểu diễn sớm nhất của bất kì đơn vị truy nhập nào của luồng đa phương tiện bắt đầu ISAU trong luồng bit.
* TPTF là thời gian biểu diễn của đơn vị truy nhập đầu tiên của luồng đa phương tiện trong thứ tự giải mã trong luồng bit bắt đầu ở ISAU.

**H.3 Các kiểu SAP**

6 kiểu SAP được định nghĩa với các tính chất sau:

* Kiểu 1: TEPT= TDEC= TSAP= TPTF
* Kiểu 2: TEPT= TDEC= TSAP< TPTF
* Kiểu 3: TEPT< TDEC= TSAP<= TPTF
* Kiểu 4: TEPT<= TDEC< TSAP= TPTF
* Kiểu 5: TEPT= TDEC< TSAP
* Kiểu 6: TEPT< TDEC< TSAP

LƯU Ý: các kiểu của SAP chỉ phụ thuộc vào đơn vị truy nhập nào có thể được giải mã cơ bản xác và sự sắp xếp của chúng trong thứ tự biểu diễn. Các kiểu tương ứng với vài thuật ngữ phổ biến:

* Kiểu 1 tương ứng với cái được biết đến trongmột số sơ đồ mật mã gọi là “điểm truy nhập ngẫu nhiên GOP đóng” (ở đó tấy cả các đơn vị truy nhập theo thứ tự giải mã, bắt đầu từ ISAP có thể được giải mã cơ bản xác, kết quả là 1 chuỗi thời gian liên tục của các đơn vị truy nhập giải mã cơ bản xác mà không có khoảng trống nào). Thêm vào đó đơn vị truy nhập trong thứ tự giải mã cũng là đơn vị truy nhập đầu tiên trong thứ tự biểu diễn.
* Kiểu 2 tương ứng với cái được biết đến trongmột số sơ đồ mật mã gọi là “điểm truy nhập ngẫu nhiên GOP đóng “, trong đó có một đơn vị truy nhập đầu tiên trong thứ tự giải mã trong luồng đa phương tiện bắt đầu từ ISAU không phải là đơn vị truy nhập đầu tiên trong thứ tự biểu diễn.
* Kiểu 3 tương ứng với cái được biết đến trongmột số sơ đồ mật mã gọi là “điểm truy nhập ngẫu nhiên GOP mở “, trong đó có một số đơn vị truy nhập trong thứ tự giải mã theo ISAU mà không thể được giải mã một cách cơ bản xác và có thời gian biểu diễn nhỏ hơn TSAP.
* Kiểu 4 tương ứng với cái được biết đến trongmột số sơ đồ mật mã gọi là “điểm truy nhập ngẫu nhiên Gradual Decoding Refresh “, trong đó có một số đơn vị truy nhập trong thứ tự giải mã bắt đầu từ và theo ISAU mà không thể được giải mã cơ bản xác và có thời gian biểu diễn nhỏ hơn TSAP.
* Kiểu 5 tương ứng với trường hợp mà có ít nhất 1 đơn vị truy nhập trong thứ tự giải mã bắt đầu từ ISAP, không thể được giải mã cơ bản xác và có thời gian biểu diễn lớn hơn TDEC trong khi TDEC là thời gian biểu diễn sớm nhất của bất kì đơn vị truy nhập nào bắt đầu từ ISAU.
* Kiểu 6 tương ứng với trường hợp mà có ít nhất 1 đơn vị truy nhập trong thứ tự giải mã bắt đầu từ ISAP, không thể được giải mã cơ bản xác và có thời gian biểu diễn lớn hơn TDEC trong khi TDEC không phải là thời gian biểu diễn sớm nhất của bất kì đơn vị truy nhập nào bắt đầu từ ISAU.

# Phụ lục I (Quy định) Đăng ký dạng MIME của các phân đoạn

**I.1 Giới thiệu**

Phụ lục này cung cấp bản đăng kí MIME của các phân đoạn đa phương tiện được định dạng theo 8.16.

**I.2 Bản đăng kí**

Tên kiểuđa phương tiện MIME: video.

Tên MIME phụ: iso\_segment.

Các tham số yêu cầu: không.

Các tham số tùy chọn: chỉ rõ trong RFC 6381 và các phiên bản sau của RFC 6381.

Các vấn đề về giải mã: như đối với video/mp4.

Các vấn đề về bảo mật: xem phần 5 của RFC 4337.

Các vấn đề về khả năng tương thích: trong họ tiêu chuẩn ISO/IEC 14496 tồn tại nhiều phần tương thích lẫn nhau, họ tiêu chuẩn này cũng có phần mềm tham chiếu cho việc đọc và ghi định dạng tệp tin này.

Thông số kĩ thuật ra công bố: ISO/IEC 14496-12:2012 (được mong đợi).

Trình ứng dụng: đa phương tiện.

Thông tin thêm:

Số kỳ diệu: không có.

Phần mở rộng của tệp tin: m4s.

Mã kiểutệp tin Macintosh: không có.

Thông tin liên hệ: David Singer, singer@apple.com.

Hình thức sử dụng: phổ thông.

Tác giả/ Người quản lí thay đổi: David Singer, chịu trách nhiệm cơ bản về định dạng tệp tin ISO/IEC 14496.

# Danh mục tài liệu tham khảo

[1] [ISO/IEC 14496-12:2012](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=61988) (Information technology -- Coding of audio-visual objects -- Part 12: ISO base media file format).

[2] [ISO/IEC 14496-12:2012](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=61988)/Cor 1: 2013(Information technology -- Coding of audio-visual objects -- Part 12: ISO base media file format - Technical Corrigendum 1.

[3] [ISO/IEC 14496-12:2012](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=61988)/Amd 1: 2013 (Information technology -- Coding of audio-visual objects -- Part 12: ISO base media file format - Amendment 1: Various enhancements including support for large metada.

[4] [ISO/IEC 14496-12:2012](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=61988)/Amd 2: 2014 (Information technology -- Coding of audio-visual objects -- Part 12: ISO base media file format - Amendment 2: Carriage of timed text and other visual overlays.