

INSO

13285-4-4

1st. Edition  
2015



جمهوری اسلامی ایران  
Islamic Republic of Iran

سازمان ملی استاندارد ایران

Iran National Standardization Organization



استاندارد ملی ایران

۱۳۲۸۵-۴-۴

چاپ اول

۱۳۹۳

فناوری اطلاعات - معماری افزاره جامع  
اتصال و اجرا (UPnP) -

قسمت ۴-۴: پروتکل واپایش (کنترل)  
افزاره صوتی تصویری - سطح ۲ -  
ساختارهای داده صوتی تصویری

**Information technology –UPnP device  
architecture – Part 4-4: Audio Video  
Device Control Protocol – Level 2 –  
Audio Video Data Structures**

ICS: 35.200

## به نام خدا

### آشنایی با سازمان ملی استاندارد ایران

مؤسسه استاندارد و تحقیقات صنعتی ایران به موجب بند یک ماده ۳ قانون اصلاح قوانین و مقررات مؤسسه استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱ تنها مرجع رسمی کشور است که وظیفه تعیین، تدوین و نشر استانداردهای ملی (رسمی) ایران را به عهده دارد.

نام موسسه استاندارد و تحقیقات صنعتی ایران به موجب یکصد و پنجاه و دومین جلسه شورای عالی اداری مورخ ۹۰/۶/۲۹ به سازمان ملی استاندارد ایران تغییر و طی نامه شماره ۲۰۶/۳۵۸۳۸ مورخ ۹۰/۷/۲۴ جهت اجرا ابلاغ شده است.

تدوین استاندارد در حوزه های مختلف در کمیسیون های فنی مرکب از کارشناسان سازمان، صاحب نظران مراکز و مؤسسات علمی، پژوهشی، تولیدی و اقتصادی آگاه و مرتبط انجام می شود و کوششی همگام با مصالح ملی و با توجه به شرایط تولیدی، فناوری و تجاری است که از مشارکت آگاهانه و منصفانه صاحبان حق و نفع، شامل تولیدکنندگان، مصرف-کنندگان، صادرکنندگان و واردکنندگان، مراکز علمی و تخصصی، نهادها، سازمان های دولتی و غیر دولتی حاصل می شود. پیش نویس استانداردهای ملی ایران برای نظرخواهی به مراجع ذی نفع و اعضای کمیسیون های فنی مربوط ارسال می شود و پس از دریافت نظرها و پیشنهادهای در کمیته ملی مرتبط با آن رشته طرح و در صورت تصویب به عنوان استاندارد ملی (رسمی) ایران چاپ و منتشر می شود.

پیش نویس استانداردهایی که مؤسسات و سازمان های علاقه مند و ذی صلاح نیز با رعایت ضوابط تعیین شده تهیه می کنند در کمیته ملی طرح و بررسی و در صورت تصویب، به عنوان استاندارد ملی ایران چاپ و منتشر می شود. بدین ترتیب، استانداردهایی ملی تلقی می شوند که بر اساس مفاد نوشته شده در استاندارد ملی ایران شماره ۵ تدوین و در کمیته ملی استاندارد مربوط که سازمان ملی استاندارد ایران تشکیل می دهد به تصویب رسیده باشد.

سازمان ملی استاندارد ایران از اعضای اصلی سازمان ملی استاندارد (ISO)<sup>۱</sup>، کمیسیون ملی الکتروتکنیک (IEC)<sup>۲</sup> و سازمان ملی اندازه شناسی قانونی (OIML)<sup>۳</sup> است و به عنوان تنها رابط<sup>۴</sup> کمیسیون کدکس غذایی (CAC)<sup>۵</sup> در کشور فعالیت می کند. در تدوین استانداردهای ملی ایران ضمن توجه به شرایط کلی و نیازمندی های خاص کشور، از آخرین پیشرفت های علمی، فنی و صنعتی جهان و استانداردهای بین المللی بهره گیری می شود.

سازمان ملی استاندارد ایران می تواند با رعایت موازین پیش بینی شده در قانون، به منظور پشتیبانی از مصرف کنندگان، حفظ سلامت و ایمنی فردی و عمومی، حصول اطمینان از کیفیت محصولات و ملاحظات زیست محیطی و اقتصادی، اجرای بعضی از استانداردهای ملی ایران را برای محصولات تولیدی داخل کشور و/یا اقلام وارداتی، با تصویب شورای عالی استاندارد، اجباری نماید. سازمان می تواند به منظور حفظ بازارهای ملی برای محصولات کشور، اجرای استانداردهای کالاهای صادراتی و درجه بندی آن را اجباری نماید. همچنین برای اطمینان بخشیدن به استفاده کنندگان از خدمات سازمان ها و مؤسسات فعال در زمینه مشاوره، آموزش، بازرسی، ممیزی و صدور گواهی سامانه های مدیریت کیفیت و مدیریت زیست-محیطی، آزمایشگاه ها و مراکز کالیبراسیون (واسنجی) و وسایل سنجش، سازمان ملی استاندارد ایران این گونه سازمان ها و مؤسسات را بر اساس ضوابط نظام تأیید صلاحیت ایران ارزیابی می کند و در صورت احراز شرایط لازم، گواهی نامه تأیید صلاحیت به آن ها اعطا و بر عملکرد آن ها نظارت می کند. ترویج افزاره ملی یکاها، کالیبراسیون (واسنجی) و وسایل سنجش، تعیین عیار فلزات گرانبها و انجام تحقیقات کاربردی برای ارتقای سطح استانداردهای ملی ایران از دیگر وظایف این سازمان است

1 - International Organization for Standardization

2 - International Electrotechnical Commission

3 - International Organization of Legal Metrology (Organisation Internationale de Metrologie Legale)

4 - Contact point

5 - Codex Alimentarius Commission

## کمیسیون فنی تدوین استاندارد

«فناوری اطلاعات - معماری افزاره جامع اتصال و اجرا (UPnP) - قسمت ۴-۴: پروتکل واپایش (کنترل) افزاره صوتی تصویری - سطح ۲ - ساختارهای داده صوتی تصویری»

### رئیس

قسمتی، سیمین  
(فوق لیسانس مهندسی فناوری اطلاعات)

### دبیر:

معروف، سینا  
(لیسانس مهندسی کامپیوتر، سخت افزار)

### اعضاء: (اسامی به ترتیب حروف الفبا)

اسدی پویا، سمیرا  
(فوق لیسانس، مهندسی فناوری اطلاعات)

سعیدی، عذرا  
(فوق لیسانس مهندسی مخابرات)

شیرازی میگون، مریم  
(لیسانس مهندسی فناوری اطلاعات)

فرهاد شیخ احمد، لیلا  
(فوق لیسانس مهندسی کامپیوتر، نرم افزار)

کماسی، مهدی  
(لیسانس مهندسی کامپیوتر، نرم افزار)

مهدوی، مهدی  
(فوق لیسانس، مهندسی فناوری اطلاعات)

وحدت جعفری، محسن  
(فوق لیسانس، هوش مصنوعی)

رئیس اداره فناوری اطلاعات، شرکت نفت پاسارگاد

عضو هیات علمی دانشگاه تربیت مدرس

یزدیان ورجانی، علی  
(دکتری، برق)

## فهرست مندرجات

صفحه	عنوان
ب	آشنایی با سازمان ملی استاندارد ایران
ج	کمیسیون فنی تدوین استاندارد
و	پیش‌گفتار
۱	۱ هدف و دامنه کاربرد
۱	۱-۱ مقدمه
۳	۲-۱ نشانه‌گذاری
۶	۳-۱ انواع داده‌های مشتق شده
۷	۴-۱ مدیریت فضای نام XML در DCP های استاندارد شده
۱۵	۵-۱ دنباله‌های تعریف شده توسط فروشنده
۱۶	۶-۱ منابع
۱۹	۲ الگوی ساختار داده صوتی تصویری
	۳ طرح ساختار داده صوتی تصویری ۲۹

## پیش‌گفتار

استاندارد «فناوری اطلاعات - معماری افزاره جامع اتصال و اجرا (UPnP) - قسمت ۴-۴: پروتکل واپایش (کنترل) افزاره صوتی تصویری - سطح ۲ - ساختارهای داده صوتی تصویری» که پیش‌نویس آن در کمیسیون‌های مربوط توسط سازمان ملی استاندارد ایران تهیه و تدوین شده و در سیصد و شصت و چهارمین اجلاس کمیته ملی استاندارد فناوری اطلاعات مورخ ۱۳۹۳/۱۲/۱۱ مورد تصویب قرار گرفته است، اینک به استناد بند یک ماده ۳ قانون اصلاح قوانین و مقررات موسسه استاندارد و تحقیقات صنعتی ایران، مصوب بهمن‌ماه ۱۳۷۱، به‌عنوان استاندارد ملی ایران منتشر می‌شود.

برای حفظ همگامی و هماهنگی با تحولات و پیشرفت‌های ملی و جهانی در زمینه صنایع، علوم و خدمات، استانداردهای ملی ایران در مواقع لزوم تجدیدنظر خواهد شد و هر پیشنهادی که برای اصلاح و تکمیل این استانداردها ارائه شود، هنگام تجدیدنظر در کمیسیون فنی مربوط مورد توجه قرار خواهد گرفت. بنابراین، باید همواره از آخرین تجدیدنظر استانداردهای ملی استفاده کرد.

منبع و مأخذی که برای تهیه این استاندارد مورد استفاده قرار گرفته به شرح زیر است:

ISO/IEC 29341-4-4, 2011: Information technology – UPnP device architecture –  
Part 4-4: Audio Video Device Control Protocol – Level 2 – Audio Video Data Structures

# فناوری اطلاعات - معماری افزاره جامع اتصال و اجرا (UPnP)<sup>۱</sup> - قسمت ۴-۴: پروتکل واپایش (کنترل) افزاره صوتی تصویری - سطح ۲ - ساختارهای داده صوتی تصویری

## ۱ هدف و دامنه کاربرد

### ۱-۱ مقدمه

هدف از تدوین این استاندارد تعریف جانمایی<sup>۲</sup> سند زبان نشانه‌گذاری امتدادپذیر (XML)<sup>۳</sup> الگوی ساختار داده صوتی تصویری (AVDT)<sup>۴</sup> است. سند AVDT، الزامات قالب و محدودیت‌های ساختارهای داده مختلف مورد استفاده در مشخصات AV UPnP را شرح می‌دهد. اگرچه این ساختارهای داده به صورت بسیار دقیق در مشخصات خدمت، تعریف شده‌اند اما در اکثر موارد، هر مشخصه ساختار داده، درجه مشخصی از تغییر را به منظور تطبیق تفاوت‌ها بین افزاره‌های مجزا، مجاز می‌داند.

هدف سند AVDT، این است تا هر افزاره را قادر سازد تا تغییر خاص خود را (در زمان عملیات) از این ساختارهای داده AV شرح دهد. اسناد AVDT به کاربران ساختارهای داده AV (مثلاً نقاط واپایش<sup>۵</sup> UPnP) اجازه می‌دهند تا تعداد نمونه‌های آن ساختارهای داده را که با مشخصات خدمت تطابق دارند اما با قابلیت‌های خاص افزاره، سازگار نیستند، کاهش دهد. هدف نهایی سند AVDT، کاهش شرایط بروز خطاهایی است که توسط نقاط واپایش سازنده نمونه‌های ساختارهای داده‌ای ایجاد می‌شوند که از قابلیت‌های ایستای<sup>۶</sup> (شناخته شده) افزاره، فراتر می‌روند. متأسفانه، سازوکار AVDT هرگز همه شرایط قابل اجتناب خطا را حذف نمی‌کند اما با ارائه اطلاعات بیشتر درباره قابلیت‌های خاص افزاره به مشتری، به کاهش آن‌ها کمک می‌کند.

همان‌طور که در بالا شرح داده شد، سند AVDT، نوع خاص و قابل خواندن توسط ماشین برای ساختار داده AV است که توسط یکی از مشخصه‌های UPnP AV تعریف شده است. برای یک افزاره معین، هر نمونه از ساختار داده هم باید با تعریف مشخصات و هم با تعریف افزاره AVDT آن ساختار داده، مطابقت داشته باشد. از قضا، سند AVDT، نوع محدودتر و مجازتری از تعریف مشخصات است. مستندات AVDT محدودترند چون جنبه‌های خاصی از ساختار داده را محدود می‌کنند (از قبیل مقادیر مجاز هر فیلد) که در غیر این-

---

1 - Universal Plug and Play

2 - Layout

3 - Extensible Markup Language

4 - Audio Video Datastructure Template

5 - Control points

6 - Static

صورت توسط تعریف مشخصات، مجاز هستند اما به علت محدودیت‌های ساختارهای AVDT، بیان الزامات پیچیده‌تری که توسط مشخصه‌ها تعیین شده‌اند امکان‌پذیر نیست (به‌طور مثال وابستگی‌های ظریف متقابل بین حوزه‌های ساختار داده). در نتیجه، نمونه‌هایی از ساختار داده که با شرح AVDT معین، مطابقت دارند، ممکن است به طور کامل با تمامی الزامات مشخص شده در مشخصات، تطابق نداشته باشند.

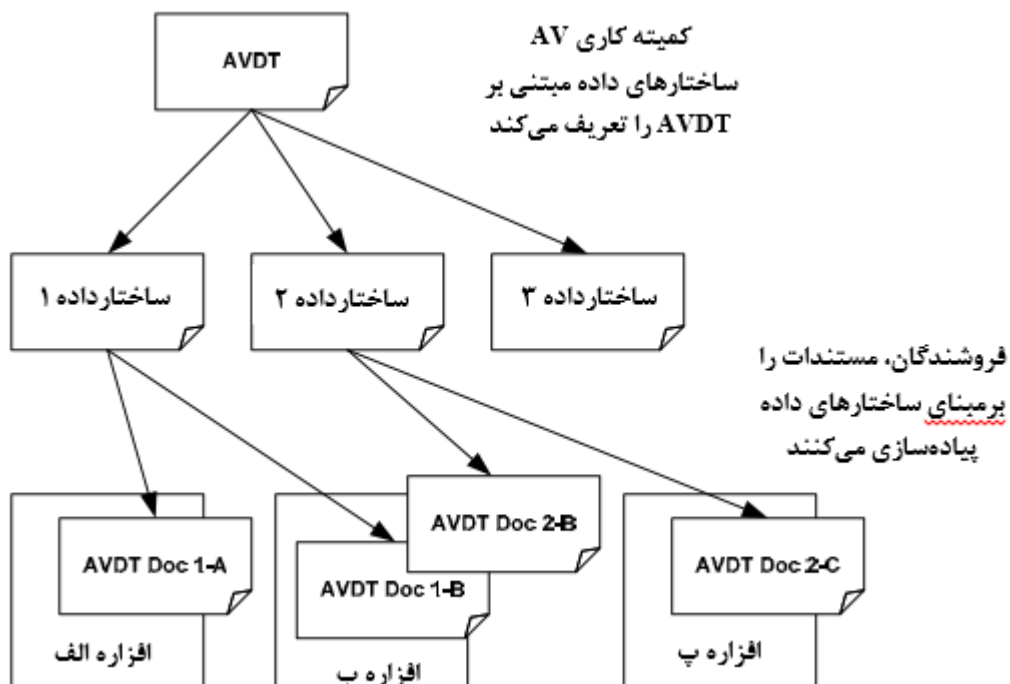
انواعی از ساختارهای داده که می‌توانند توسط سند AVDT شرح داده شوند مجموعه‌ای (غیر سلسله مراتبی) از مقادیر ویژگی‌های نام‌گذاری شده را نشان می‌دهند. مجموعه‌ای از اسامی ویژگی‌های مجاز و مقادیر مجاز آن‌ها برای یک ساختار داده معین، توسط یکی از مشخصات UPnP AV مشخص می‌شوند. نمونه‌های مجزای این ساختارهای داده، از طریق سند XML که ویژگی‌ها و اجزای آن با مجموعه ویژگی‌های نام‌گذاری شده تطابق دارند نشان داده می‌شوند. به عبارت دیگر، در سند XML که با نمونه معینی از ساختار داده خاص، تطابق دارد، هر ویژگی و جزء XML شامل مقداری از ویژگی نام‌گذاری شده خاص است.

سند AVDT، از نظر مفهومی مشابه طرح XML است، از این نظر که هر دو هستار، ویژگی‌ها و اجزایی از XML را مشخص می‌کنند که در هر نمونه سند معینی، ظاهر می‌شوند. علاوه بر این، هم طرح‌های XML و هم اسناد AVDT مقادیر مجزای را مشخص می‌کنند که برای هر جزء و ویژگی که با ویژگی خاصی متناظرند مجاز هستند، اما متفاوت با طرح XML، سند AVDT می‌تواند وابستگی‌های خاصی را بین دو یا چند ویژگی، مشخص کند. برای مثال، مجموعه‌ای از مقادیر مجاز یک ویژگی، ممکن است به مقدار واقعی ویژگی دیگر، وابسته باشد. نمایش این نوع روابط متقابل با استفاده از طرح XML دشوار است. بنابراین، ساختار سند AVDT مورد نیاز است.

در سناریوهای مختلف ساختار AV، گاهی به تبادل قابلیت‌های افزاره نیاز است تا سطح بالای قابلیت همکاری متقابل تضمین شود. به منظور بیان قابلیت به صورت پارامتری، مشخصه AV الگوهای مختلفی را برای هر منظور، تعریف می‌کند. افزاره از این الگو استفاده می‌کند و به آن مقادیری می‌دهد تا قابلیت‌هایش را در زمان اجرا منعکس کند.

الگوی ساختار داده AV (AVDT)، ساختار متداولی برای تعیین الگوهای مختلفی است که «ساختار داده» نامیده می‌شوند و در XML نوشته می‌شوند و هر ساختار داده از زیرمجموعه‌ای از AVDT استفاده می‌کند تا موارد ضروری را تأمین کند.





شکل ۱- نمونه کاربرد AVDT

## ۲-۱ نشانه‌گذاری

در این استاندارد ملی، ویژگی‌ها به‌عنوان الزامی، توصیه‌شده یا اختیاری مانند زیر شرح داده‌شده‌اند: کلمات کلیدی «باید»، «نباید»، «الزامی»، «توصیه می‌شود»، «توصیه نمی‌شود»، «توصیه شده»، «مجاز است» و «اختیاری» در این ویژگی باید آن‌گونه که در RFC 2119 شرح داده‌شده است، تفسیر شوند. علاوه بر این، کلمات کلیدی زیر در این مشخصات استفاده‌شده‌اند:

ممنوع - تعریف یا رفتاری که توسط این ویژگی، ممنوع شده است و متضاد الزامی است.  
الزامی مشروط - تعریف یا رفتار، به شرطی بستگی دارد. اگر شرط مشخص شده برآورد شود، تعریف یا رفتار، الزامی است و در غیر این صورت ممنوع است.

اختیاری مشروط - تعریف یا رفتار به شرطی بستگی دارد. اگر شرط مشخص شده برآورده شود، تعریف یا رفتار، اختیاری است و در غیر این صورت ممنوع است.

این کلمات کلیدی هرگاه الزاماتی را برای پروتکل و ویژگی‌های برنامه کاربردی به‌طور واضح مشخص می‌کنند که قابلیت تعامل دارند و امنیت پیاده‌سازی را تحت تأثیر قرار می‌دهند، با حرف بزرگ نوشته می‌شوند. هنگامی که این کلمات با حروف بزرگ نوشته نمی‌شوند، به این معناست که با مفهوم طبیعی خود در زبان، به‌کاررفته‌اند.

- رشته‌هایی که به معنای واقعی کلمه، مورد نظر هستند در «نقل قول دوتایی» قرار می‌گیرند.
- کلماتی که مورد تأکید هستند به‌صورت کج چاپ می‌شوند.

- کلمات کلیدی که توسط کمیته کاری UPnP AV تعریف شده‌اند با استفاده از سبک نویسه‌ی [forum](#) چاپ می‌شوند.
- کلمات کلیدی که توسط ویژگی‌های معماری افزاره‌ی UPnP تعریف شده‌اند با استفاده از سبک نویسه‌ی [arch](#) چاپ می‌شوند.
- جداکننده‌ی دونقطه دوتایی «::»، رابطه والد - فرزندی سلسله مراتبی را بین دو شیء که توسط دونقطه دوتایی از هم جدا شده‌اند مشخص می‌کند. این جداکننده در زمینه‌های متعددی مورد استفاده است، برای مثال:

Service::Action(), Action()::Argument, parentProperty::childProperty

### ۱-۲-۱ انواع داده

این ویژگی از تعاریف نوع داده از دو منبع مختلف استفاده می‌کند. انواع داده تعریف شده معماری افزاره UPnP برای تعریف انواع داده‌ی متغیر حالت و شناسه‌ی اقدام [DEVICE] استفاده شده‌اند. فضای نام طرح XML<sup>۱</sup> برای تعریف خواص انواع داده، استفاده شده است. [XML SCHEMA- 2]

برای معماری افزاره UPnP که انواع داده‌های [boolean](#) را تعریف کرده است، استفاده از مقدار «[0](#)» برای مقادیر نادرست و مقدار «[1](#)» برای مقادیر درست، قویاً توصیه می‌شود؛ اما هنگامی که به‌عنوان شناسه‌های ورودی به کار می‌روند، مقادیر «[false](#)»، «[no](#)»، «[true](#)»، «[yes](#)» نیز مجاز است مورد توجه قرار گیرند و باید مورد قبول واقع شوند؛ اما قویاً توصیه می‌شود که تمامی متغیرهای حالت [boolean](#) و شناسه‌های خروجی، به‌صورت «[0](#)» یا «[1](#)» نشان داده شوند.

برای انواع داده‌های [boolean](#) مشخص شده طرحی XML، استفاده از مقدار «[0](#)» برای مقادیر نادرست و مقدار «[1](#)» برای مقادیر درست، قویاً توصیه می‌شود؛ اما هنگامی که به‌عنوان ویژگی‌های ورودی استفاده می‌شوند، مقادیر «[false](#)» و «[true](#)» نیز مجاز است مورد توجه قرار گیرند و باید مورد قبول واقع شوند؛ اما قویاً توصیه می‌شود که تمامی ویژگی‌های بولی به‌صورت «[0](#)» یا «[1](#)» نشان داده شوند.

### ۲-۲-۱ رشته‌های جایگذاری شده در رشته‌های دیگر

بعضی شناسه‌ها و متغیرهای رشته‌ای شرح داده شده در این استاندارد، شامل زیررشته‌هایی هستند که باید به‌طور مستقل برای پردازش‌های دیگر، قابل تشخیص و استخراج باشند. این امر، مستلزم مشخص کردن جداکننده‌های زیررشته‌ای مناسب و سازوکار گریز<sup>۲</sup> است، به‌طوری که این جداکننده‌ها بتوانند به‌صورت نویسه‌های معمولی در رشته و زیررشته‌های مستقل آن ظاهر شوند. این استاندارد از رشته‌های جایگذاری شده در دو زمینه استفاده می‌کند - فهرست‌های مقادیر جداشده با ویرگول (CSV)<sup>۳</sup> (به بند ۱-۳-۱،

1 - Schema  
2 - Escaping mechanism  
3 - Comma separated values

«فهرست‌های مقادیر جداشده با ویرگول (CSV)» رجوع شود) و مقدار مشخصات در رشته‌های معیارهای جستجو. قراردادهای گریز، از نویسه واکچ خط<sup>۱</sup> «\» به صورت زیر استفاده می‌کنند (کد نویسه u+005C).  
الف - واکچ خط («\») در هر دو زمینه به صورت «\\» نشان داده می‌شود.

ب - ویرگول (»,»)»

- در ورودی‌های زیر رشته‌ای تکی در فهرست‌های CSV، به صورت «\» نشان داده می‌شود.

- در رشته‌های جستجو، گریز ندارد.

پ - نقل قول دوتایی («"»)

- در فهرست‌های CSV گریز ندارد.

- در رشته‌های جستجو هنگامی که به صورت حائل شروع یا پایان مقدار مشخصات، ظاهر می‌شود، گریز ندارد.

- در رشته‌های جستجو هنگامی که به صورت نویسه‌ای ظاهر می‌شود که بخشی از مقدار مشخصات است به صورت «\» نشان داده می‌شود.

### ۳-۲-۱ شکل Backus-Naur بسط یافته (EBNF)<sup>۲</sup>

شکل بسط یافته Backus-Naur در این استاندارد برای توصیف ترکیب رسمی ساختارهای خاصی مورد استفاده است. در اینجا کاربرد آن، طبق منبع [EBNF] است.

### ۱-۳-۲-۱ اصول چابی برای EBNF

نمادهای Non-terminal، توالی‌های غیر مشخص نویسه‌ها از مجموعه حروف بزرگ و کوچک انگلیسی، ارقام «0» تا «9» و خط پیوند<sup>۳</sup> («\_») هستند. توالی‌های نویسه بین 'single quotes'، رشته‌های پایانه‌ای هستند و باید با معنی واقعی، در رشته‌های معتبر، ظاهر شوند. توالی نویسه‌ها بین (\*comment delimiters\*)، تعاریف زبان انگلیسی یا توضیحات تکمیلی نمادهای مربوط هستند. فاصله خالی در EBNF برای جدا کردن جزءهای EBNF استفاده می‌شود نه برای نشان دادن فاصله خالی در رشته‌های صحیح. کاربرد فاصله خالی در رشته‌های معتبر، در EBNF به وضوح شرح داده شده است و در نهایت، EBNF از عملگرهای زیر استفاده می‌کند.

---

1 - Backslash

2 - Extended Backus-Naur form

3 - Hyphen

## جدول ۱-۱ عملگرهای EBNF

معناشناسی	عملگر
تعریف- نماد non-terminal در سمت چپ، با یک یا چند توالی متناوب از نمادهای terminal و/یا non-terminal در سمت راست آن مشخص می‌شود.	::=
جداکننده جایگزین- توالی‌های سمت راستی را که به‌طور مستقل، مجاز به تعاریف نمادهای non-terminal سمت چپ هستند، جدا می‌کند.	
تکرار پوچ <sup>۱</sup> - به این معناست که عبارت سمت چپ مجاز است صفر یا چند بار، روی دهد.	*
تکرار غیر پوچ - به این معناست که عبارت سمت راست آن باید دست کم یک‌بار و مجاز است چندبار روی دهد.	+
اختیاری- عبارت بین کروشه‌ها اختیاری است	[ ]
گروه‌بندی- عبارات بین پرانتزها را گروه‌بندی می‌کند.	( )
گستره نویسه- به طور کامل همه نویسه‌های بین عملوندهای چپ و راست را نشان می‌دهد.	-

### ۱-۳ انواع داده مشتق شده

این بند نوع داده‌ی مشتق شده‌ای را تعریف می‌کند که به‌صورت نوع داده‌ی رشته‌ای با ترکیب خاصی نشان داده می‌شود. این مشخصه، از تعاریف نوع داده‌ی **string** استفاده می‌کند که از دو منبع مختلف نشأت می‌گیرد. نوع داده‌ی رشته‌ای مشخص‌شده‌ی ساختار افزاره UPnP، برای مشخص کردن متغیر حالت و انواع داده‌ی رشته‌ای شناسه اقدام، استفاده می‌شود. فضای نام طرح XML برای تعریف خواص انواع داده‌ی xsd:string، استفاده می‌شود. تعریف زیر، برای هر دو نوع داده‌ی رشته‌ای به کار می‌رود.

#### ۱-۳-۱ فهرست‌های مقادیر جداشده با ویرگول (CSV)

خدمات صوتی تصویری UPnP از متغیرهای حالت، شناسه‌های اقدام و ویژگی‌هایی استفاده می‌کنند که فهرست‌ها یا آرایه‌های تک‌بعدی مقادیر را نشان می‌دهند. معماری افزاره UPnP، نسخه ۱.۰، [DEVICE]، هم نوع آرایه و هم نوع فهرست را فراهم نمی‌کند و فقط نوع فهرست در اینجا تعریف می‌شود. مجاز است فهرست‌ها همگن (همه‌ی ارقام از یک نوع) یا ناهمگن (مقادیری با انواع مختلف) باشند. هم‌چنین ممکن است فهرست‌ها از تکرار توالی‌های همگن یا ناهمگن تشکیل شده باشند که همه آن‌ها معناشناسی و نحو مشابهی دارند. (تعداد یکسان ارقام، انواع مشابه ارقام و به ترتیب مشابه). نوع داده‌ی فهرست همگن، **string** یا xsd:string است و توسط CSV(X) مشخص شده که در آن X، نوع مقادیر تکی است. نوع داده‌ی فهرست ناهمگن نیز **string** یا xsd:string است و توسط CSV(X,Y,Z) مشخص می‌شود که در آن X، Y و Z نوع دیگری از مقادیر تکی هستند. اگر تعداد مقادیر در فهرست ناهمگن، برای نشان دادن تکی هر نوع، بسیار زیاد

- باشد آن نوع متغیر به صورت CSV (ناهمگن) نشان داده می‌شود و شرح متغیر، اطلاعات اضافی را در بر می‌گیرد به طوری که توالی مورد انتظار مقادیر در فهرست و انواع مربوط به آن‌ها ظاهر می‌شوند. نوع داده‌ی فهرست توالی تکراری، **string** یا `xsd:string` است و توسط `CSV({X,Y,Z})` مشخص می‌شود که در آن X، Y و Z انواع مقادیر تکی در توالی هستند و این توالی مجاز است صفر یا چند بار تکرار شود.
- فهرستی به‌عنوان نوع **string** (برای متغیرهای حالت و شناسه‌های اقدام) یا نوع `xsd:string` (برای خواص) ارائه شده است.
  - ویرگول‌ها، مقادیر فهرست را از هم جدا می‌کنند.
  - مقادیر صحیح در CSV با همان ترکیب نوع داده‌های صحیح که در [DEVICE] مشخص شده‌اند نشان داده می‌شوند (علامت‌های راهنمای اختیاری، صفرهای راهنمای اختیاری، کدهای ASCII)
  - مقادیر بولی، در CSV شناسه اقدام و متغیر حالت به صورت «**0**» برای نادرست و «**1**» برای درست، نشان داده می‌شوند. این مقادیر، زیرمجموعه‌ای از مقادیر نوع داده‌ی بولی هستند که در [DEVICE] مشخص شده‌اند: **0, false, 1, no, true, yes**.
  - مقادیر بولی برای خواص CSV به صورت «**0**» برای نادرست و «**1**» برای درست، نشان داده شده‌اند. این مقادیر، زیرمجموعه‌ای از مقادیر نوع داده‌ی بولی مشخص شده در [XML SCHEMA- 2] هستند: `0, false, 1, true`.
  - قراردادهای گریز برای نویسه‌های واکنج خط و ویرگول، در بند ۲-۲-۱ «رشته‌های جایگذاری شده در رشته‌های دیگر» مشخص شده‌اند.
  - فاصله خالی، قبل، بعد یا داخل هر نوع داده عددی، مجاز نیست.
  - فاصله خالی، قبل، بعد یا داخل هر نوع داده دیگر، بخشی از رقم است.

جدول ۱-۲ مثال‌های CSV

توضیحات	مقدار	پالایش نوع رشته
فهرستی از دو معیار مرتب‌سازی ویژگی	"+artist,-date"	CSV ( <b>string</b> ) or CSV (xsd:string)
فهرستی از ۵ عدد صحیح	"1,-5,006,0,+7"	CSV ( <b>int</b> ) or CSV (xsd:integer)
فهرستی از ۴ بولی	"0,1,1,0"	CSV ( <b>Boolean</b> ) or CSV (xsd:Boolean)
فهرستی از ۲ نام "Fred".Smith", "jones" و "Davey"	"smith\fred,jones\davey"	CSV ( <b>string</b> ) or CSV (xsd:string)
توجه کنید که مقدار دوم رشته‌ای با جاهای خالی است	"-29837, string with leading blanks,0"	CSV ( <b>i4,string,ui2</b> ) or CSV (xsd:int, Xsd:string, Xsd:usingnedshort)
CSV غیرمجاز. فاصله خالی به‌عنوان بخشی از مقدار صحیح مجاز نیست	"3, 4"	CSV ( <b>i4</b> ) or CSV (xsd:int)
فهرستی از ۳ مقدار رشته‌ای خالی	",""	CSV ( <b>string</b> ) or CSV (xsd:int)
فهرستی از تعداد غیر مشخصی از افراد و ویژگی‌های مربوطه. هر فرد با ۳ جزء شرح داده می‌شود: نام <b>string</b> واحد <b>string</b> و سال‌های خدمت <b>ui2</b> یا نام xsd:string, واحد xsd:string و سال‌های خدمت XSD:Unsignedshort	Alice,marketing,5,sue,R&D,21,Dave,Finance,7"	CSV (heterogeneous)

#### ۱-۴ مدیریت فضای نام XML در DCP استاندارد شده

مشخصات UPnP استفاده گسترده‌ای از فضای نام XML به عمل می‌آورد. این امر اجازه تفکیک DCP را می‌دهد و حتی اجزای DCP تکی را جدا می‌کند تا به‌طور مستقل طراحی شوند و هنگامی که اسناد XML را به اشتراک می‌گذارند از تداخل نام جلوگیری کنند. هر نامی در اسناد XML دقیقاً متعلق به یک فضای نام است. در مستندات، نام‌های XML به یکی از این دو شکل ظاهر می‌شوند: مناسب و نامناسب. یک نام نامناسب (یک نام بدون دونقطه) هیچ نویسه دونقطه‌ای «(:)» را در بر نمی‌گیرد. نام نامناسب به فضای نام اختیاری مستندات تعلق دارد. نام مناسب، دو نام بدون «:» است که با یک نویسه دونقطه، از هم جدا شده‌اند. نام بدون دونقطه قبل از دونقطه، یک پیشوند فضای نام برای نامی مناسب است و نام بدون دونقطه پس از دونقطه، نام محلی نام مناسب است. (معنی محلی فضای نام توسط پیشوند فضای نام شناسایی می‌شود) به همین شکل، نام نامناسب یک نام محلی در فضای نام پیش‌فرض است.

نام رسمی فضای نام، نشانی وب (URI)<sup>۱</sup> است. پیشوند فضای نام مورداستفاده در یک سند XML، نام فضای نام نیست. نام فضای نام باید کلاً منحصر به فرد باشد که تعریف منفردی دارد و برای هر کسی که از این فضای نام استفاده می‌کند قابل دسترسی است. هر جا که استفاده شود چه داخل و چه خارج مستندات XML یک معنا دارد اما پیشوند فضای نام در کاربرد رسمی XML، فقط در یک سند XML مشخص می‌شود و باید

1 - Uniform Resource Identifier

به‌طور محلی، در آن سند، منحصر به فرد باشد. هر نام معتبر بدون دونقطه XML مجاز است استفاده شود و در استفاده رسمی XML، هیچ‌گاه نیازی به دو سند XML برای استفاده از پیشوند فضای نام مشابه برای اشاره به یک فضای نام مشابه نیست. ایجاد و استفاده از پیشوند فضای نام توسط کمیته W3C XML در [XML-NMSP] منحصراً به عنوان جایگزین خلاصه محلی مناسب برای نام URI کامل یک فضای نام در مستندات، استانداردسازی شد. تمامی ویژگی‌های شی AV توسط نام ویژگی‌ها و عناصر در XML نشان داده شده‌اند بنابراین نام تمامی ویژگی‌ها به یک فضای نام XML تعلق دارد.

به همان دلیل که پیشوندهای فضای نام در مستندات XML، راحت هستند، در متن مشخصات نیز برای اشاره به فضاهای نام، استفاده از پیشوند فضای نام مناسب است؛ بنابراین، این مشخصه، یک پیشوند استاندارد را برای همه فضاهای نام XML مورد استفاده در اینجا، اظهار می‌دارد. به علاوه، این مشخصات فراتر از یک سند XML منفرد، هر جا که بین پیشوندها، متن‌ها، مثال‌های XML و ویژگی‌های رشته مقدردهی شده معنا دارند، دامنه را بسط می‌دهند. این بسط دامنه، از قواعد XML برای کاربرد در مستندات، فراتر نمی‌رود بلکه فقط آن‌ها را در زمینه‌های مهمی که برای مشخصات XML، خارج از محدوده هستند تکمیل می‌کند. برای مثال، شناسه‌های اقدامی که به ویژگی‌های CDS اشاره می‌کنند از قبیل شناسه *SearchCriteria* از اقدام *Search()* یا شناسه *Filter* از اقدام *Browse()*، باید از پیشوندهای فضای نام از پیش تعریف شده در هنگام اشاره به ویژگی‌های CDS استفاده کند («upnp:»، «dc:» و غیره)

تمامی فضاهای نام مورد استفاده در این مشخصه، در جداول «تعاریف فضای نام» و «اطلاعات مرتبط با طرح» فهرست شده‌اند. برای هر کدام از این فضای نام‌ها، جدول ۱-۳، «تعاریف فضای نام»، شرح مختصری از آن، نام (URI) و نام پیشوند استاندارد تعریف شده‌اش را ارائه می‌دهد. برخی فضاهای نام در برگرفته شده در این جداول به‌طور مستقیم مورد استفاده یا مورد اشاره در این استاندارد نیست. آن‌ها برای تکمیل تطبیق موقعیت‌هایی که در آن این مشخصه به همراه دیگر مشخصات UPnP برای ایجاد سامانه کاملی از خدمات و افزارها استفاده می‌شود، در برگرفته می‌شوند. برای مثال، از آنجاکه خدمت ثبت زمان بندی به خدمت فهرست راهنمای محتوا بستگی و اشاره دارد، پیشوند فضای نام از پیش تعریف شده «sis»، گنجانده می‌شود. مشخصات منفرد در چنین مجموعه‌هایی همه از پیشوند استاندارد مشابه استفاده می‌کنند، همچنین پیشوندهای استاندارد جدول ۱-۴، «اطلاعات مرتبط با طرح»، برای اطلاعات فضای نام اضافی مرجع متقابل استفاده شده‌اند. این جدول شامل عنصر (عناصر) ریشه‌ی سند XML معتبر هر فضای نام، نام پرونده طرح، اطلاعات نسخه (در بخش زیر با جزئیات مورد بحث قرار می‌گیرند) و پیوندی به بند ۱-۴-۳، «مثال‌های کاربرد فضای نام» برای طرح مرتبط، است.

تعاریف الزامی برای این فضای نامی، اسناد مورد اشاره در جدول ۱-۳ هستند. طرح‌ها برای پشتیبانی از این تعاریف هم برای درک انسان هم ابزارهای آزمون طراحی شده‌اند. به هر حال محدودیت‌های زبان طرح XML مشکلاتی را برای طرح‌های تعریف شده‌ی UPnP در نشان دادن صحیح تمامی جزئیات تعاریف فضای نام ایجاد می‌کند، در نتیجه طرح‌ها بسیاری از مستندات XML را که طبق مشخصات، معتبر نیستند، معتبرسازی می‌کنند.

کمیته کاری انتظار دارد تا پالایش این طرح‌ها را پس از انتشار مشخصات برای کاهش تعداد مستندات که توسط طرح‌ها، دارای اعتبار می‌شوند ادامه دهد اما طرح‌ها همچنان مستندات اطلاعاتی و پشتیبانی کننده‌ای هستند. ممکن است برخی طرح‌ها در نسخه‌های آتی مشخصات، الزامی شوند.

### جدول ۱-۳ تعاریف فضای نام

پیشوند فضای نام استاندارد	نام فضای نام	شرح فضای نام	مرجع مستند تعریف الزامی
<b>فضای نام تعریف شده توسط کمیته کاری AV</b>			
av	Urn:schemas-upnp-org:av:av	انواع داده مشترک برای استفاده در طرح‌های AV	[AV-XSD]
avs	Urn:schemas-upnp-org:av:avs	ساختارهای مشترک برای استفاده در طرح AV	[AVS-XSD]
avdt	Urn:schemas-upnp-org:av:avdt	الگوی ساختار داده	[AVDT]
Avt-event	Urn: schemas-upnp-org:metadata-1-0/AVT/	متغیر حالت روی داده <i>LastChange</i> برای AVTransport	[AVT]
Cds-event	Urn:schemas-upnp-org:av:cds-event	متغیر حالت روی داده <i>LastChange</i> برای ContentDirectory	[CDS]
Didl-lite	Urn: schemas-upnp-org:metadata-1-0/Didl-lite/	ساختار و فراداده برای ContentDirectory	[CDS]
Rcs-event	Urn: schemas-upnp-org:metadata-1-0/RCS/	متغیر حالت روی داده <i>LastChange</i> برای RenderingControl	[RCS]
srs	Urn:schemas-upnp-org:av:srs	ساختار و فراداده برای ScheduledRecording	[SRS]
Srs-event	Urn:schemas-upnp-org:av:srs-event	متغیر حالت روی داده <i>LastChange</i> برای ScheduledRecording	[SRS]
upnp	Urn: schemas-upnp-org:metadata-1-0/upnp/	فراداده برای ContentDirectory	[CDS]
<b>فضاهای نام تعریف شده بیرونی</b>			
dc	http://purl.org/dc/elements/1.1/	هسته دوبلین	[DC-TERMS]
xsd	http://www.w3.org/2001/XMLSchema	زبان طرح XML نسخه ۱.۰	[XML SCHEMA-1] [XML SCHEMA-2]
xsi	http://www.w3.org/2001/xmlschema-instance	طرح نمونه XML طرح سند	بند ۲-۶ و ۳-۲-۷ از [XML SCHEMA-1]
xml	http://www.w3.org/XML/1998/namespace	فضای نام «xml:»	[XML-NS]



جدول ۱-۴ اطلاعات مرتبط با طرح

پیشوند فضای نام استاندارد	URI وابسته و نام پرونده • برگه ۱، برگه ۲، برگه ۳	عنصر(عناصر) ریشه‌ی معتبر	مرجع طرح
فضاهای نام تعریف‌شده توسط کمیته AV			
av	Av-vn-yyymmdd.xsd Av-vn.xsd Av.xsd	n/a	[AV-XSD]
avs	Avs-vn-yyymmdd.xsd Avs-vn.xsd Avs.xsd	<capabilities> <features> <statevariablevaluepairs>	[AVS-XSD]
avdt	Avdt-vn-yyymmdd.xsd Avdt-vn.xsd Avdt.xsd	<avdt>	[AVDT]
Avt-event	Avt-event-vn-yyymmdd.xsd Avt-event-vn.xsd Avt-event.xsd	<event>	[AVT-EVENT-XSD]
Cds-event	cds-event-vn-yyymmdd.xsd cds-event-vn.xsd cds-event.xsd	<stateevent>	[CDS-EVENT-XSD]
Didl-lite	Didl-lite-vn-yyymmdd.xsd Didl-lite-vn.xsd Didl-lite.xsd	<DIDL-Lite>	[DIDL-LITE-XSD]
Rcs-event	Rcs-event-vn-yyymmdd.xsd Rcs-event-vn.xsd Rcs-event.xsd	<event>	[RCS-EVENT-XSD]
Srs	srs-vn-yyymmdd.xsd srs-vn.xsd srs.xsd	<srs>	[SRS-XSD]
Srs-event	Srs-event-vn-yyymmdd.xsd Srs-event-vn.xsd Srs-event.xsd	<stateevent>	[SRS-EVENT-XSD]
upnp	upnp-vn-yyymmdd.xsd upnp-vn.xsd upnp.xsd	n/a	[UPNP-XSD]
فضاهای نام تعریف‌شده بیرونی			
dc	Absolute URL: <a href="http://dublincore.org/schemas/xmls/simpledc20021212.xsd">http://dublincore.org/schemas/xmls/simpledc20021212.xsd</a>		[DC-XSD]
xsd		n/a	<schema>
xsi		n/a	n/a
xml		n/a	[XML-XSD]

#### ۱-۴-۱ الزامات پیشوند فضای نام

رخدادهای بسیاری در مشخصات انواع داده رشته‌ای که شامل نام‌های XML است، وجود دارد. این نام‌های XML در رشته‌ها، در شرایط آگاهی دادن فضای نام، پردازش نخواهند شد؛ بنابراین، همه رخدادهای در

مستندات نمونه نام‌های XML در رشته‌ها باید از پیشوندهای فضای نام استاندارد که در جدول ۳ بیان شد استفاده کنند.

به‌منظور پردازش مناسب اسناد XML شرح داده‌شده، افزارها و نقاط واپایش باید از پردازشگرهای XML آگاه از فضای نام، [XML-NMSP] برای خواندن و نوشتن استفاده کنند. همان‌طور که توسط [XML-NMSP] اجازه داده شده است، تنها پیشوندهای فضای نام مورداستفاده در یک سند نمونه، در اختیار ایجادکننده سند هستند؛ بنابراین، پیشوند اظهارشده برای فضای نام مجاز است از پیشوند استاندارد، متفاوت باشد. همه افزارها باید قادر باشند تا به‌طور صحیح هر سند نمونه XML معتبر را پردازش کنند، حتی وقتی از یک پیشوند غیراستاندارد برای اسامی XML معمولی استفاده می‌کنند؛ اما توصیه اکید می‌شود که تمامی افزارها از این پیشوندهای استاندارد برای تمامی مستندات استفاده کنند تا از اشتباه گرفتن توسط خواننده‌های انسانی و ماشینی جلوگیری شود. این پیشوندهای استاندارد در تمامی متون توصیفی و همه مثال‌های XML در مشخصات بالا و مشخصات UPnP استفاده شده‌اند. هم‌چنین هر مشخصه منفردی مجاز است برای متن توصیفی خودش، فضای نام پیش‌فرضی را قرار دهد. در این حالت ممکن است نام‌های آن فضای نام، بدون پیشوند ظاهر شوند.

برای هر مشخصه UPnP AV اگر وجود داشته باشد، فضای نام پیش‌فرض در نظر گرفته‌شده در جدول ۱-۵، «فضای نام پیش‌فرض برای مشخصات AV» ارائه شده است.

یادآوری - تمامی طرح‌های UPnP AV ویژگی‌های «نامناسب» را اعلام می‌کنند، بنابراین پیشوندهای فضای نام هرگز با نام‌های ویژگی تعریف‌شده توسط کمیته AV استفاده نمی‌شوند.

جدول ۱-۵ فضای نام پیش‌فرض برای مشخصات AV

نام مشخصه AV	پیشوند فضای نام پیش‌فرض
AVTransport	Avt - event
ConnectionManager	n/a
ContentDirectory	Didi - lite
MediaRenderer	n/a
MediaServer	n/a
RenderingControl	rsc - event
ScheduledRecording	srs

#### ۱-۴-۲ نام‌های فضای نام، نگارش فضای نام و نگارش طرح

مشخصات خدمت UPnP AV، چندین ساختار داده را تعریف می‌کند (از قبیل متغیرهای حالت و شناسه‌های اقدام) که فرق آن‌ها در یک سند XML است که باید با یک یا چند فضای نام XML خاص، تطابق داشته باشد. هر فضای نام، منحصرأً توسط نام فضای نام اختصاص داده شده شناسایی می‌شود. فضای نام‌هایی که توسط کمیته کاری AV مشخص می‌شوند باید توسط URN نام‌گذاری شوند. به جدول ۳، «تعاریف فضای نام» برای فهرست فعلی نام‌های فضای نام رجوع شود. علاوه بر این هر فضای نام مطابق با سند طرح XML

است که باز نمود فضای نام مرتبط قابل خواندن توسط ماشین را فراهم می‌کند تا معتبر سازی خودکار مستندات نمونه XML را فراهم آورد (متغیرهای حالت و شناسه‌های اقدام).

در یک طرح XML و سند نمونه XML، نام هر فضای نام متناسب به صورت مقداری از ویژگی `XmInS` در عنصر ریشه ظاهر می‌شود. هر ویژگی `XmInS` نیز شامل پیشوند فضای نامی است که با آن فضای نام مرتبط است تا عنصر و نام‌های ویژگی را که در فضاهای نام مختلف تعریف شده‌اند، مناسب‌سازی کند. طرح‌هایی که با فضاهای نام فهرست شده تطابق دارند، توسط مقادیر URI که در ویژگی `SchemaLocation` نیز در عنصر اصلی، فهرست شده‌اند مشخص می‌شوند. (به بند ۱-۴-۳، «مثال‌های کاربردی فضای نام» رجوع شود) به‌منظور فعال کردن قابلیت پیش‌سو و پس‌سو، اسامی فضای نام، به‌طور همیشگی مشخص می‌شوند و نباید حتی وقتی که نسخه‌ی جدیدی از یک ویژگی تغییر می‌کند، تعریف فضای نام، تغییر کند؛ اما همه‌ی تغییرات در مشخصات فضای نام باید به‌صورت پس‌سو، تطابق داشته باشد. به‌عبارت‌دیگر، مشخصات به‌روز فضای نام نباید هیچ سند XML را که با مشخصات اولیه همان فضای نام تطابق دارند غیر معتبر سازد. برای مثال، این بدان معناست که فضای نام نباید طوری تغییر کند که به ویژگی یا عنصر جدیدی نیاز باشد. اگرچه اسامی فضاهای نام نباید تغییر کند اما فضاهای نام هنوز نسخه‌هایی دارد که مجموعه خاصی از تغییرات تعریف‌شده را برمی‌گرداند. هر وقت تعریف یک فضای نام، تغییر می‌کند، نسخه فضای نام، یکی افزایش می‌یابد. هر زمان که نسخه جدید فضای نام ایجاد شود، یک سند طرح XML جدید (`.xsd`) ایجاد و منتشر می‌شود به‌طوری‌که مشخصات جدید فضای نام به شیوه‌ای که قابل خواندن با افزاره باشد نشان داده شود. از آنجاکه سند طرح XML، فقط نمایشی از مشخصات فضای نام است، خطاهای انتقال می‌توانند روی دهند بنابراین گاهی اوقات طرحی منتشر شده لازم است تا خطاهای نمایش فضای نام یا خطاهای دیگر را دوباره اصلاح کند. به‌منظور شناسایی آسان تعدد بالقوه انتشار طرح‌ها برای فضاهای نام مشابه، URI هر طرح منتشر شده باید با قالب زیر، مطابقت داشته باشد.

- برگه ۱:

"`http://www.upnp.org/schemas/av/" schema-root-name "-v" ver "-" yyyyymmdd`

که در آن:

- `schema-root-name` نام عنصر اصلی فضای نام است که این طرح نشان می‌دهد.
  - `Ver`، شماره نسخه فضای نامی است که توسط طرح نشان داده می‌شود.
  - `yyyyymmdd`، سال، ماه و روزی است (در تقویم میلادی) که این طرح، منتشر شده است.
- جدول ۴، «اطلاعات مرتبط با طرح»، قالب‌های URI را برای هر یک از فضاهای نام که توسط کمیته UPnP AV تعریف شده‌اند شناسایی می‌کند.

به‌عنوان مثال، طرح اصلی URI برای فضاهای نام «`rscs-event`» (که توسط انتشارات اصلی UPnP AV در سال ۲۰۰۲ منتشر شد) این‌طور بود:

«`http://www.upnp.org/schemas/av/rscs-event-v1-20020625.xsd`»

هنگامی که مشخصات خدمت UPnP AV، در سال ۲۰۰۶ به روز شد، URI برای نسخه به روز شده‌ی فضای نام «rcs-event» به صورت زیر بود:

«<http://www.upnp.org/schemas/av/rcs-event-v2-20060531.xsd>»

اما در سال ۲۰۰۶، در طرح URI برای فضای نام تازه ایجادشده‌ی «srs-event» به این شکل درآمد:

«<http://www.upnp.org/schemas/av/srs-event-v1-20060531.xsd>»

باید توجه داشت که فیلد نسخه برای طرح «srs-event»، «v2» است چون مربوط به نسخه دوم فضای نام است.

علاوه بر این، برای URI طرح‌ی تاریخ‌داری که مرتبط با هر فضای نام هست، هر فضای نام نیز مجموعه‌ای از URI طرح‌ی بی‌تاریخ دارد. این URI بی‌تاریخ، دو قالب متمایز را با معانی که به‌طور جزئی متفاوت هستند، دارد:

- برگه ۲:

“<http://www.upnp.org/schemas/av/>” *schema-root-name* “-v” *ver*

که در آن *ver* در بالا شرح داده شده است.

- برگه ۳:

“<http://www.upnp.org/schemas/av/>” *schema-root-name*

فرم ۲ مربوط به URI طرح بی‌تاریخ، همیشه با جدیدترین انتشار طرح‌ای که نسخه فضای نام نشان داده شده توسط *ver* را نمایش می‌دهد پیوندی دارد. برای مثال URI بی‌تاریخ، «[.../av/rcs-event-v2.xsd](http://www.upnp.org/schemas/av/rcs-event-v2.xsd)» با جدیدترین انتشار طرح نسخه دو فضای نام «rcs-event» مرتبط است؛ بنابراین در ۳۱ ماه می سال ۲۰۰۶ (20060531)، URI طرح بی‌تاریخ، با طرح‌ای مرتبط شد که به «[.../av/rcs-event-v2.xsd](http://www.upnp.org/schemas/av/rcs-event-v2.xsd)» شناخته می‌شود. علاوه بر این اگر طرح نسخه‌ی دو فضای نام «rcs-event» دوباره منتشر شود، برای مثال برای تصحیح خطا در طرح 20060531، پس از URI طرح بی‌تاریخ مشابه («[.../av/rcs-event-v2.xsd](http://www.upnp.org/schemas/av/rcs-event-v2.xsd)») به‌طور خودکار برای مرتبط کردن طرح نسخه‌ی دو به‌روز شده برای فضای نام «rcs-event» به‌روز می‌شود. فرم سه از URI طرح بی‌تاریخ، همیشه با جدیدترین انتشار طرح‌ای که بالاترین نسخه فضای نام منتشر شده را نشان می‌دهد، مرتبط است. برای مثال، در ۲۵ ژوئن ۲۰۰۲ (20020625)، URI طرح بی‌تاریخ «[.../av/rcs-event.xsd](http://www.upnp.org/schemas/av/rcs-event.xsd)» با طرح‌ای مرتبط شد که به «[.../av/rcs-event-v1-20020625.xsd](http://www.upnp.org/schemas/av/rcs-event-v1-20020625.xsd)» شناخته می‌شود؛ اما در ۳۱ می ۲۰۰۶ (20060531) URI طرح بی‌تاریخ مشابهی با طرح‌ای مرتبط شد که به «[.../av/rcs-event-v2-20060531.xsd](http://www.upnp.org/schemas/av/rcs-event-v2-20060531.xsd)» شناخته می‌شود.

هنگام ارجاع دادن URI طرح در یک سند نمونه XML یا سند ارجاع طرح XML، قواعد کاربردی زیر اعمال می‌شوند:

- تمامی اسناد نمونه چه با خدمت و چه با نقطه واپایش ایجاد شده باشند باید از فرم سه استفاده کنند.
- همه‌ی طرح‌های منتشرشده UPnP AV که به دیگر طرح‌های UPnP ارجاع می‌دهند نیز باید از فرم سه استفاده کنند.

در یک سند نمونه XML، تعریف ویژگی `schemaLocation` از فضای نام طرح XML «<http://www.w3.org/2002/XMLSchema-instance>» می‌آید و رخداد تکی این ویژگی می‌تواند محل یک یا چند طرح را نشان دهد. مقدار ویژگی `schemaLocation` از فهرست مقادیر جداشده با فاصله خالی تشکیل شده است که به‌عنوان فضای نامی که URI محل طرحی خود را به دنبال دارد تفسیر می‌شود. این زوج توالی، برای طرح‌هایی که نیاز است تا برای این سند الگو باشند، به‌صورت الزامی تکرار می‌شود. علاوه بر نام‌گذاری طرح URI و قوانین کاربردی شرح داده‌شده در بالا، هر طرح منتشرشده باید شامل یک ویژگی `version` در عنصر ریشه `<schema>` باشد؛ و مقدارش باید با قالب زیر، متناظر باشد:

ver "yyymmdd"

که در آن ver و yyymmdd در بالا شرح داده‌شده‌اند.

خود ویژگی `version`، خودشناسایی نسخه فضای نام و تاریخ منتشرشده طرح را ارائه می‌دهد. برای مثال، در طرح اصلی منتشرشده، برای فضای نام «`rsc-event`» (`.../rsc-event-v2-20020625.xsd`)، عنصر ریشه `<schema>` شامل ویژگی `version="2-20020625"` است.

#### ۳-۴-۱ مثال‌های کاربرد فضای نام

ویژگی `SchemaLocation` برای اسناد نمونه XML از فضای نام نمونه طرح XML «<http://www.w3.org/2002/xml-schema-instance>» می‌آید. رخداد تکی یک ویژگی می‌تواند محل یک یا چند طرح را اظهار کند. مقدار ویژگی `schemaLocation` از فهرست مقادیر جداشده با فاصله خالی تشکیل شده است: نام فضای نام، محل طرح URL خود را به دنبال دارد. این زوج توالی برای طرح‌هایی که باید برای این سند نمونه باشند به‌صورت الزامی تکرار می‌شود.

مثال ۱- سند نمونه XML `DIDL-Lite` نمونه‌گیری شود. باید توجه داشت که ارجاع‌ها به طرح‌های UPnP AV هیچ اطلاعاتی از نسخه یا تاریخ انتشار را ندارد. به‌عبارت‌دیگر، ارجاعات از فرم ۳ در بالا پیروی می‌کنند. در نتیجه، این مثال، برای همه انتشارات مشخصات خدمت UPnP AV معتبر است.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<DIDL-Lite
```

```
xmlns:dc="http://purl.org/dc/elements/1.1/"
```

```

xmlns="urn:schemas-upnp-org:metadata-1-0/DIDL-Lite/"
xmlns:upnp="urn:schemas-upnp-org:metadata-1-0/upnp/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
urn:schemas-upnp-org:metadata-1-0/DIDL-Lite/
http://www.upnp.org/schemas/av/didl-lite.xsd
urn:schemas-upnp-org:metadata-1-0/upnp/
http://www.upnp.org/schemas/av/upnp.xsd">
<item id="18" parentID="13" restricted="0">
...
</item>
</DIDL-Lite>

```

#### ۵-۱ دنباله‌های تعریف‌شده توسط فروشنده

هرگاه فروشندگان، متغیرهای حالت، اقدامات یا ویژگی اضافی تعریف‌شده توسط فروشنده ایجاد کنند، نام‌های اختصاص‌یافته و بازنمود XML آن‌ها باید از قراردادهای نام‌گذاری و قواعد XML زیر پیروی کند.

#### ۱-۵-۱ نام‌های اقدام تعریف‌شده توسط فروشنده

نام‌های اقدام تعریف‌شده توسط فروشنده باید با «X» شروع شوند. علاوه بر این، باید با یک نام دامنه اختصاص‌یافته ICANN متعلق به فروشنده که با نویسه («-») همراه است، دنبال شود. سپس به دنبال آن باید نام اقدام اختصاص‌یافته به فروشنده بیاید. نام اقدام واگذارشده فروشنده نه باید شامل نویسه‌ی خط پیوند («-»، «2D هگزادسیمال در کد UTF-8) باشد و نه نویسه‌ی علامت عدد<sup>۱</sup> («#»، «23 در مبنای ۱۶ از کدهای UTF-8). نام‌های اقدام اختصاص‌یافته به فروشنده، نسبت به بزرگ و کوچک بودن حرف، حساس هستند. اولین نویسه نام باید یک حرف از کدهای US-ASCII («a» - «z»، «A» - «Z»، رقم US-ASCII («1» - «9»، خط زیر («\_») یا رقم یا حرف یونیکد بزرگ‌تر از U+077F باشد. نویسه‌های بعدی باید یک حرف از کدهای US-ASCII («a» - «z»، «A» - «Z»، رقم US-ASCII («1» - «9»، خط زیر («\_»)، نقطه («.»)، نویسه‌ی ترکیبی یونیکد، گسترده‌ساز<sup>۲</sup> یا رقم یا حرف یونیکد بزرگ‌تر از U+077F باشد. سه حرف اول، در هیچ ترکیبی، نباید «XML» باشد.

1 - Number sign  
2 - Extender

#### ۲-۵-۱ نام‌های متغیر حالت تعریف‌شده توسط فروشنده

نام‌های متغیر حالت تعریف‌شده توسط فروشنده باید با «X» شروع شوند. علاوه بر این، باید با یک نام دامنه اختصاص‌یافته ICANN متعلق به فروشنده که با نویسه («-») همراه است، دنبال شود. سپس به دنبال آن باید نام اقدام اختصاص‌یافته به فروشنده بیاید. نام اقدام واگذارشده فروشنده نه باید شامل نویسه‌ی خط پیوند («-»، «2D هگزادسیمال در کد UTF-8) باشد و نه نویسه‌ی علامت عدد («#»، «2D در مبنای ۱۶ از کدهای UTF-8). نام‌های اقدام اختصاص‌یافته به فروشنده، نسبت به بزرگ و کوچک بودن حرف، حساس هستند. اولین نویسه نام باید یک حرف از کدهای US-ASCII («a» - «z»، «A» - «Z»، «a» - «z»، «A» - «Z») رقم US-ASCII («1» - «9») خط زیر («\_») یا رقم یا حرف یونیکد بزرگ‌تر از U+077F باشد. نویسه‌های بعدی باید یک حرف از کدهای US-ASCII («a» - «z»، «A» - «Z»، «a» - «z»، «A» - «Z») رقم US-ASCII («1» - «9») خط زیر («\_») نقطه («.») نویسه‌ی ترکیبی یونیکد، گسترده‌ساز یا رقم یا حرف یونیکد بزرگ‌تر از U+077F باشد. سه حرف اول، در هیچ ترکیبی، نباید «XML» باشد.

#### ۳-۵-۱ ویژگی‌ها و عناصر XML تعریف‌شده توسط فروشنده

فروشنده‌های UPnP مجاز هستند ویژگی‌ها و عناصر فاقد استاندارد را به سند XML استاندارد UPnP اضافه کنند. هر اضافه کردن ویژگی باید در حیطه فضای نام XML متعلق به فروشنده باشد. XML دلخواه باید در عنصری که با «X» شروع شود، محصور باشد و این عنصر باید زیر عنصر ترکیبی استاندارد باشد. ویژگی‌های فاقد استاندارد مجاز است به عناصر استاندارد فراهم شده توسط این ویژگی‌ها که در قلمروی فضای نام فروشنده است و با «X» شروع می‌شود، اضافه شوند.

#### ۴-۵-۱ نام‌های خاصیت تعریف‌شده توسط فروشنده

ممکن است فروشنده‌های UPnP خواص فاقد استاندارد را به خدمت ContentDirectory اضافه کنند. هر اضافه کردن خاصیتی باید در حیطه فضای نام XML متعلق به فروشنده باشد. نام خاصیت اختصاص‌یافته توسط فروشنده نباید شامل نویسه‌ی خط پیوند («-»، «2D در مبنای ۱۶ از کدهای UTF-8) باشد. نام‌های خاصیت اختصاص‌یافته توسط فروشنده، نسبت به حروف کوچک و بزرگ حساس‌اند. اولین نویسه نام باید یک حرف از کدهای US-ASCII («a» - «z»، «A» - «Z»، «a» - «z»، «A» - «Z») رقم US-ASCII («1» - «9») خط زیر («\_») یا رقم یا حرف یونیکد بزرگ‌تر از U+077F باشد. نویسه‌های بعدی باید یک حرف از کدهای US-ASCII («a» - «z»، «A» - «Z»، «a» - «z»، «A» - «Z») رقم US-ASCII («1» - «9») خط زیر («\_») نقطه («.») نویسه‌ی ترکیبی یونیکد، گسترده‌ساز یا رقم یا حرف یونیکد بزرگ‌تر از U+077F باشد. سه حرف اول، در هیچ ترکیبی، نباید «XML» باشد.

#### ۶-۱ منابع

در این بند تمام مراجع الزامی مورداستفاده در مشخصات UPnP AV فهرست شده و داخل کروشه، نشان هر مرجع استفاده شده، آمده است.

**1-6-1** [AVARCH] – AVArchitecture:1, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-AVArchitecture-v1-20080930.pdf>. Latest version

available at: <http://www.upnp.org/specs/av/UPnP-av-AVArchitecture-v1.pdf>.

**1-6-2** [AVDT] – AV DataStructure Template:1, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-AVDataStructure-v1-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-AVDataStructure-v1.pdf>.

**1-6-3** [AVDT-XSD] – XML Schema for UPnP AV Datastructure Template:1, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/avdt-v1-20080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/avdt-v1.xsd>.

**1-6-4** [AV-XSD] – XML Schema for UPnP AV Common XML Data Types, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/av-v2-20080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/av-v2.xsd>.

**1-6-5** [AVS-XSD] – XML Schema for UPnP AV Common XML Structures, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/avs-v2-20080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/avs-v2.xsd>.

**1-6-6** [AVT] – AVTransport:2, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-AVTransport-v2-Service-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-AVTransport-v2-Service.pdf>.

**1-6-7** [AVT-EVENT-XSD] – XML Schema for AVTransport:2 LastChange Eventing, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/avt-event-v220080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/avt-event-v2.xsd>.

**1-6-8** [CDS] – ContentDirectory:3, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-ContentDirectory-v3-Service-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-ContentDirectory-v3-Service.pdf>.

**1-6-9** [CDS-EVENT-XSD] – XML Schema for ContentDirectory:3 LastChange Eventing, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/cds-event-v120080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/cds-event-v1.xsd>.

**1-6-10** [CM] – ConnectionManager:2, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-ConnectionManager-v2-Service-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-ConnectionManager-v2Service.pdf>.

**1-6-11** [DC-XSD] – XML Schema for UPnP AV Dublin Core. Available at: <http://www.dublincore.org/schemas/xmls/simpledc20020312.xsd>.

**1-6-12** [DC-TERMS] – DCMI term declarations represented in XML schema language. Available at: <http://www.dublincore.org/schemas/xmls>.

**1-6-13** [DEVICE] – UPnP Device Architecture, version 1.0, UPnP Forum, July 20, 2006. Available at: <http://www.upnp.org/specs/architecture/UPnP-DeviceArchitecture-v1.0-20060720.htm>. Latest version available at: <http://www.upnp.org/specs/architecture/UPnP-DeviceArchitecturev1.0.htm>.

**1-6-14** [DIDL] – ISO/IEC CD 21000-2:2001, Information Technology - Multimedia Framework - Part 2:Digital Item Declaration, July 2001.



- 1-6-15** [DIDL-LITE-XSD] – XML Schema for ContentDirectory:3 Structure and Metadata (DIDL-Lite), UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/didl-lite-v220080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/didl-lite-v2.xsd>.
- 1-6-16** [EBNF] – ISO/IEC 14977, Information technology - Syntactic metalanguage - Extended BNF, December 1996.
- 1-6-17** [HTTP/1.1] – HyperText Transport Protocol – HTTP/1.1, R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999. Available at: <http://www.ietf.org/rfc/rfc2616.txt>.
- 1-6-18** [IEC 61883] – IEC 61883 Consumer Audio/Video Equipment – Digital Interface - Part 1 to 5. Available at: <http://www.iec.ch>.
- 1-6-19** [IEC-PAS 61883] – IEC-PAS 61883 Consumer Audio/Video Equipment – Digital Interface - Part 6. Available at: <http://www.iec.ch>.
- 1-6-20** [ISO 8601] – Data elements and interchange formats – Information interchange - Representation of dates and times, International Standards Organization, December 21, 2000. Available at: ISO 8601:2000.
- 1-6-21** [MIME] – IETF RFC 1341, MIME (Multipurpose Internet Mail Extensions), N. Borenstein, N. Freed, June 1992. Available at: <http://www.ietf.org/rfc/rfc1341.txt>.
- 1-6-22** [MR] – MediaRenderer:2, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-MediaRenderer-v2-Device-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-AV-MediaRenderer-v2-Device.pdf>.
- 1-6-23** [MS] – MediaServer:3, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-MediaServer-v3-Device-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-AV-MediaServer-v3-Device.pdf>.
- 1-6-24** [RCS] – RenderingControl:2, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-RenderingControl-v2-Service-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-RenderingControl-v2-Service.pdf>.
- 1-6-25** [RCS-EVENT-XSD] – XML Schema for RenderingControl:2 LastChange Eventing, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/rcs-event-v120080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/rcs-event-v1.xsd>.
- 1-6-26** [RFC 1738] – IETF RFC 1738, Uniform Resource Locators (URL), Tim Berners-Lee, et. Al. December 1994. Available at: <http://www.ietf.org/rfc/rfc1738.txt>.
- 1-6-27** [RFC 2045] – IETF RFC 2045, Multipurpose Internet Mail Extensions (MIME) Part 1:Format of Internet Message Bodies, N. Freed, N. Borenstein, November 1996. Available at: <http://www.ietf.org/rfc/rfc2045.txt>.
- 1-6-28** [RFC 2119] – IETF RFC 2119, Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, 1997. Available at: <http://www.faqs.org/rfcs/rfc2119.html>.
- 1-6-29** [RFC 2396] – IETF RFC 2396, Uniform Resource Identifiers (URI): Generic Syntax, Tim Berners-Lee, et al, 1998. Available at: <http://www.ietf.org/rfc/rfc2396.txt>.

- 1-6-30** [RFC 3339] – IETF RFC 3339, Date and Time on the Internet: Timestamps, G. Klyne, Clearswift Corporation, C. Newman, Sun Microsystems, July 2002. Available at: <http://www.ietf.org/rfc/rfc3339.txt>.
- 1-6-31** [RTP] – IETF RFC 1889, Realtime Transport Protocol (RTP), H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, January 1996. Available at: <http://www.ietf.org/rfc/rfc1889.txt>.
- 1-6-32** [RTSP] – IETF RFC 2326, Real Time Streaming Protocol (RTSP), H. Schulzrinne, A. Rao, R. Lanphier, April 1998. Available at: <http://www.ietf.org/rfc/rfc2326.txt>.
- 1-6-33** [SRS] – ScheduledRecording:2, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/specs/av/UPnP-av-ScheduledRecording-v2-Service-20080930.pdf>. Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-ScheduledRecording-v2Service.pdf>.
- 1-6-34** [SRS-XSD] – XML Schema for ScheduledRecording:2 Metadata and Structure, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/srs-v2-20080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/srs-v2.xsd>.
- 1-6-35** [SRS-EVENT-XSD] – XML Schema for ScheduledRecording:2 LastChange Eventing, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/srs-event-v120080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/srs-event-v1.xsd>.
- 1-6-36** [UAX 15] – Unicode Standard Annex #15, Unicode Normalization Forms, version 4.1.0, revision 25, M. Davis, M. Dürst, March 25, 2005. Available at: <http://www.unicode.org/reports/tr15/tr15-25.html>.
- 1-6-37** [UNICODE COLLATION] – Unicode Technical Standard #10, Unicode Collation Algorithm version 4.1.0, M. Davis, K. Whistler, May 5, 2005. Available at: <http://www.unicode.org/reports/tr10/tr10-14.html>.
- 1-6-38** [UPNP-XSD] – XML Schema for ContentDirectory:3 Metadata, UPnP Forum, September 30, 2008. Available at: <http://www.upnp.org/schemas/av/upnp-v3-20080930.xsd>. Latest version available at: <http://www.upnp.org/schemas/av/upnp-v3.xsd>.
- 1-6-39** [UTS 10] – Unicode Technical Standard #10, Unicode Collation Algorithm, version 4.1.0, revision 14, M. Davis, K. Whistler, May 5, 2005. Available at: <http://www.unicode.org/reports/tr10/tr10-14.html>.
- 1-6-40** [UTS 35] – Unicode Technical Standard #35, Locale Data Markup Language, version 1.3R1, revision 5, M. Davis, June 2, 2005. Available at: <http://www.unicode.org/reports/tr35/tr355.html>.
- 1-6-41** [XML] – Extensible Markup Language (XML) 1.0 (Third Edition), François Yergeau, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, eds., W3C Recommendation, February 4, 2004. Available at: <http://www.w3.org/TR/2004/REC-xml-20040204>.
- 1-6-42** [XML-NS] – The “xml:” Namespace, November 3, 2004. Available at: <http://www.w3.org/XML/1998/namespace>.
- 1-6-43** [XML-XSD] – XML Schema for the “xml:” Namespace. Available at: <http://www.w3.org/2001/xml.xsd>.

**1-6-44** [XML-NMSP] – Namespaces in XML, Tim Bray, Dave Hollander, Andrew Layman, eds., W3C Recommendation, January 14, 1999. Available at: <http://www.w3.org/TR/1999/REC-xml-names-19990114>.

**1-6-45** [XML SCHEMA-1] – XML Schema Part 1: Structures, Second Edition, Henry S. Thompson, David Beech, Murray Maloney, Noah Mendelsohn, W3C Recommendation, 28 October 2004. Available at: <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028>.

**1-6-46** [XML SCHEMA-2] – XML Schema Part 2: Data Types, Second Edition, Paul V. Biron, Ashok Malhotra, W3C Recommendation, 28 October 2004. Available at: <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028>.

**1-6-47** [XMLSCHEMA-XSD] – XML Schema for XML Schema. Available at: <http://www.w3.org/2001/XMLSchema.xsd>.

**1-6-48** [XPath20] – XML Path Language (XPath) 2.0. Anders Berglund, Scott Boag, Don Chamberlin, Mary F. Fernandez, Michael Kay, Jonathan Robie, Jerome Simeon. W3C Recommendation, 21 November 2006. Available at: <http://www.w3.org/TR/xpath20>.

**1-6-49** [XQUERY10] – XQuery 1.0 An XML Query Language. W3C Recommendation, 23 January 2007. Available at: <http://www.w3.org/TR/2007/REC-xquery-20070123>.

## ۲ الگوی ساختار داده صوتی تصویری

مورد زیر، طرح‌ریزی کلی الگوی AVDT را نشان می‌دهد. ممکن است اجزا و ویژگی‌های بیشتری در نسخه‌های آتی الگوهای AVDT اضافه شوند.

سبک نویسه‌ی *forum* برای نشان دادن اسامی مشخص شده توسط کمیته کاری AV هستند. پیاده‌سازی‌ها باید بخش‌هایی را که در شیوه‌ی نویسه *vendor* چاپ شده‌اند، تکمیل کنند.

```
<?xml version="1.0"?>
<AVDT
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="
    urn:schemas-upnp-org:av:avdt
    http://www.upnp.org/schemas/av/avdt.xsd"
  xmlns="urn:schemas-upnp-org:av:avdt">
  <contextID>data structure identification context</contextID>
  <dataStructType>data structure name</dataStructType>
  <fieldTable>
  <field>
  <name>field name</name>
  <dataType csv="csv data type" maxSize="max length">
  field data type
  </dataType>
  <minCountTotal>minimum total occurrences</minCountTotal>
  <maxCountTotal>maximum total occurrences</maxCountTotal>
  <minListSizeTotal>min # of entries in CSV</minListSizeTotal>
  <maxListSizeTotal>max # of entries in CSV</maxListSizeTotal>
  <allowedValueDescriptor>
  <dependentField defaultDependency="1|0">
  <name>field name</name>
  <anyValue></anyValue>
  <valueList>
  <value>enumerated value</value>
```

```

// Other values go here
</valueList>
<valueRange>
  <minimum>minimum value</minimum>
  <maximum>maximum value</maximum>
  <step>increment value</step>
</valueRange>
// Other value ranges go here
</dependentField>
// Other dependent fields go here
<minCount>minimum occurrences of these values</minCount>
<maxCount>maximum occurrences of these values</maxCount>
<minListSize>minimum # of these values in CSV</minListSize>
<maxListSize>maximum # of these values in CSV</maxListSize>
<defaultValue>default value</defaultValue>
<allowAny></allowAny>
<allowedValueList>
  <allowedValue>enumerated value</allowedValue>
// Other allowed values go here
</allowedValueList>
<allowedValueRange>
  <minimum>minimum value</minimum>
  <maximum>maximum value</maximum>
  <step>increment value</step>
</allowedValueRange>
// Other allowed value Ranges go here
</allowedValueDescriptor>
// Other allowed value descriptors go here
</field>
// Other field declarations go here
</fieldTable>
</AVDT>

```

## xml ۱-۲

برای تمامی مستندات XML، الزامی است. نسبت به بزرگ و کوچک بودن حروف، حساس است.

## AVDT ۲-۲

الزامی است. باید «urn:schemas-upnp.org:av:advt» را به عنوان مقداری برای ویژگی xmlns، داشته باشد. این امر به طرح الگوی ساختار داده کمیته کاری UPnP AV اشاره می‌کند. تا وقتی که xmlns مشابهی مورد استفاده است، الگوی ساختار داده باید به صورت پس‌سو، سازگار باشد یعنی باید توسط پیاده‌سازی موروثی<sup>۱</sup>، قابل استفاده باشد. شامل تمامی اجزایی است که خدمت را شرح می‌دهند یعنی شامل اجزای فرعی زیر است:

## Context ID ۱-۲-۲

الزامی است. Xsd:anyType مفهومی را مشخص می‌کند که در آن نوع ساختار داده معنادار است. معمولاً، این جزء شامل یک شناسه منحصر به فرد برای نمونه خدمت خاص افزارهای است که شامل این ساختار داده است.

برای مثال:

Uuid:device-uuID::um:Schemas-upnp.org:sevice:scheduleRecording:1

## datastructureType ۲-۲-۲

الزامی است. Xsd:QName نوع ساختار داده را مشخص می‌کند. نام نوع ساختار داده به فروشنده بستگی دارد و همان‌طور که در بند ۳ سند w3c «فضاهای نام در XML» [XML-NMSP] مشخص شد، باید QName باشد. انواع مشابه ساختارهای داده باید با اسم مشابهی مشخص شوند و انواع ساختارهای داده که متفاوتند باید اسامی متفاوتی داشته باشند.

## fieldTable ۳-۲-۲

الزامی است. بند توضیح را برای فیلدهایی که برای این نوع ساختار داده تعریف شده‌اند، شروع می‌کند. شامل صفر یا چند جزء فرعی زیر است:

## field ۱-۳-۲-۲

الزامی است. برای هر فیلدی که در این نوع ساختار داده در بر گرفته می‌شود یک‌بار تکرار می‌شود و شامل اجزای فرعی زیر است:

## name ۱-۱-۳-۲-۲

الزامی است. Xsd:string نام فیلدی را که در این جزء فیلد توصیف شده است تعیین می‌کند و باید در یکی از قالب‌های زیر باشد:

QName

QName"@NCName

"@"NCName

NCName"@NCName

که در آن QName و NCName، در بند ۳ سند w3c، «فضای نام در XML» [XML-NMSP]، مشخص شده‌اند. برای فیلدهایی که با جزء XML متناظرند (در سند ساختار داده XML (datastructureType)) نام باید شامل نام جزء XML باشد که از قالب QName استفاده می‌کند مثلاً element-name برای فیلدهایی که با یک مشخصه XML متناظرند (در سند XML (datastructureType) ساختار داده) نام باید شامل نام مشخصه XML باشد که از قالبی جز قالب QName استفاده می‌کند، به‌طور مثال

element-name@attribute-name

## ۲-۱-۳-۲-۲ dataType

الزامی است. Xsd:string. نوع اطلاعات این فیلد را تعیین می کند و باید یک QName با پیشوند فضای نام «Xsd» باشد. QName در بند ۳ سند w3c، «فضاهای نام در XML» [XML-NMSP] مشخص شده است و باید یکی و فقط یکی از انواع اطلاعات مشخص شده توسط «XML Schema Part-2» [XML SCHEMA-2] باشد و شامل ویژگی های زیر است:

### - @CSV

اختیاری است. Xsd:string. اگر وجود داشته باشد نشان می دهد که این فیلد رشته شامل فهرست CSV مقادیر انواع داده مشخص شده توسط مشخصه CSV است و باید با نشانه گذاری نوع داده CSV که در بند ۱-۳-۱ مشخص شد، «فهرست های مقادیر جدا شده با ویرگول (CSV)»، مطابقت داشته باشد. برای مثال، مقدار «Xsd:int»، CSV مقادیر صحیح را نشان می دهد. AVDT، هیچ محدودیتی را به مقدار نوع داده که مشخص شده، اعمال نمی کند. اما هر ساختار داده مشخص شده توسط نمونه AVDT فقط از تعداد محدودی انواع داده CSV استفاده می کند و فقط زمانی باید مشخص شود که datatype، برابر با «string» است و field شامل فهرست CSV مقادیر است. تعداد کمینه و بیشینه ثبت ها در فهرست CSV، توسط miniListSize Total، maxListSize Total، maxListSize و minListSize که در زیر تعریف شده اند مشخص شده است.

### - @maxSize

اختیاری است. Xsd:unsignedInt. فقط زمانی معنادار است که datatype، برابر با «string» است. تعداد بیشینه بیت های مجاز برای هر فیلد را نشان می دهد. یادآوری - از آنجایی که برخی مجموعه نویسه ها در هر نویسه (مثل UTF-16) از بیت های متعدد استفاده می کنند، maxsize لزوماً تعداد بیشینه نویسه های مجاز را نشان نمی دهد.

## ۲-۱-۳-۲-۲ minCountTotal

اختیاری است. Xsd:unsignedInt. تعداد کمینه رویدادهای این فیلد در کل سند XML است. مقدار پیش فرض، صفر است، به این معنا که این فیلد اختیاری است و ممکن است در برخی نمونه های این ساختار داده، (datastructureType) در نظر گرفته نشود. مقدار ۱ یا بیشتر، به این معناست که این فیلد الزامی است و باید در هر نمونه از این ساختار داده، حداقل تعداد مشخصی از دفعات، وجود داشته باشد.

## ۲-۱-۳-۲-۲ maxCountTotal

اختیاری است. Xsd:string. تعداد بیشینه رویدادهای این فیلد در کل سند XML است. مقدار آن باید با یک عدد صحیح بی علامت یا مقدار «UNBOUNDED» باشد.

مقدار پیش فرض، عدد یک است به این معنا که این فیلد نباید بیش از یکبار در هر نمونه از این ساختار داده (datastructureType) وجود داشته باشد. مقدار صفر نشان می دهد که این فیلد، ممنوع است و نباید در هیچ یک از نمونه های این ساختار داده وجود داشته باشد. مقدار «UNBOUNDED» نشان می دهد که هیچ حد

از پیش تعیین شده‌ای برای تعداد دفعاتی که این فیلد وجود دارد، نیست. مقدار `maxCountTotal` باید بزرگتر یا مساوی `minCountTotal` باشد.

#### **miniListSizeTotal ۵-۱-۳-۲-۲**

اختیاری است. `Xsd:unsignedInt`. فقط برای فیلد نوع CSV معتبر است یعنی هنگامی که ویژگی `@CSV`، در `name` مشخص می‌شود. تعداد کمینه ثبت‌ها در هر نمونه از این فیلد CSV است. مقدار پیش‌فرض، صفر است به این معنا که هنگامی که این فیلد وجود دارد، باید دست‌کم شامل تعداد مشخصی ثبت در فهرست CSV باشد.

#### **maxiListSizeTotal ۶-۱-۳-۲-۲**

اختیاری است. `Xsd:dtring`. فقط برای فیلد نوع CSV معتبر است یعنی زمانی که ویژگی `@CSV` در `name` مشخص شده است. تعداد بیشینه ثبت در هر نمونه از این فیلد CSV است. مقدار آن باید یک عدد صحیح مثبت یا مقدار «UNBOUNDED» باشد. مقدار پیش‌فرض، عدد ۱ است به این معنا که فیلد CSV نباید شامل بیش از یک ثبت در هر بار باشد. مقدار "UNBOUNDED" نشان می‌دهد که حد از قبل تعیین شده-ای برای تعداد ثبت‌ها در فهرست CSV وجود ندارد. مقدار `maxListSizeTotal` باید بزرگتر یا مساوی `minListSizeTotal` باشد.

#### **allowedValueDescriptor ۷-۱-۳-۲-۲**

الزامی است. شرح مجموعه داده با مقادیر مجاز برای این فیلد را شروع می‌کند. چندین جزء `allowedValueDescriptor` مجاز هستند. پوشش کلی مقادیر مجاز برای این فیلد، به‌طور ساده توالی مقادیر مجاز منفردی در هر `allowedValueDescriptor` است که باید شامل `allowAny` یا `allowedValueList` و یا `allowedValueRage` باشد و شامل اجزای فرعی زیر است:

#### **dependentField ۱-۷-۱-۳-۲-۲**

اختیاری است. مقادیر فیلد وابسته‌ای را شناسایی می‌کند که «زمینه اعتبار» را برای مجموعه مقادیر داده مجازی که در این `allowedValueDescriptor` تعریف شده‌اند تعیین می‌کند. به عبارت دیگر، هنگامی که `dependentField` برای یکی از مقادیر تعریف شده در `anyValue`، `valueList` و یا `valueRange` اجزای فرعی `dependentField` تنظیم می‌شود، این فیلد باید شامل یکی از مقادیر شناسایی شده توسط اجزای فرعی `allowedValueDescriptor` مانند `allowAny`، `allowedValueList` و یا `allowedValueRage` باشد. اگر چندین جز `dependentField` در یک جز معین `allowedValueDescriptor` وجود داشته باشد، «زمینه اعتبار» برای مجموعه داده مقادیر مجاز، در زمانی که همه `dependentField` ها در دامنه/مقدار مشخص شده‌شان تنظیم شده‌اند وجود دارد. یعنی برای چندین ثبت `dependentField`، عملگر «AND» اعمال می‌شود تا زمینه خاصی را برای مقادیر مجازی که دنبال می‌شوند تعریف کنند. یک جزء جا افتاده‌ی `dependentField` نشان می‌دهد که مقادیر مجاز این `allowedValueDescriptor` در تمامی زمینه‌ها به جز در زمینه‌هایی که توسط دیگر بلوک‌های هم‌تای `allowedValueDescriptor` تعریف شده در این فیلد

شناسایی شده‌اند معتبر هستند. باید شامل anyValue یا valueRange و valueList باشند و شامل اجزای فرعی و ویژگی‌های زیر است:

#### - @default Dependency

اختیاری است. Xsd:boolean مقدار ۱ نشان می‌دهد که value/valueRange تعریف شده در این dependentField شامل مقدار پیش فرض (defaultValue) در dependentField است. مقدار پیش فرض برای defaultDependency، صفر است به این معنا که مقدار پیش فرض dependentField در value/valueRange تعریف شده در dependentField در بر گرفته نشده است. این جزء توسط نقاط واپاشی استفاده می‌شود که dependentField را پشتیبانی نمی‌کند تا مجموعه‌ای از مقادیر مجازی را شناسایی کنند که قابلیت‌های افزاره را در زمانی که dependentField شامل مقدار پیش فرض است، منعکس کنند.

#### - name

الزامی است. Xsd:string نام dependentField را شناسایی می‌کند که مقدارش مجموعه‌ای از مقادیر مجازی است تا روی این فیلد تأثیر گذارند. به عبارت دیگر، مجموعه‌ای از مقادیر مجاز برای این فیلد، به مقدار dependentField بستگی دارند. باید از قوانین قالب مشخص شده در جزء فرعی فیلد name، پیروی کند.

#### - anyValue

اختیاری است. Xsd:string وجود این جزء، نشان می‌دهد که dependentField مجاز است برای هر مقدار مجاز توسط نوع داده dependentField تنظیم شود. محتوای این جزء، باید خالی باشد. هیچ مقداری نباید در اجزای ValueList یا ValueRange در بر گرفته شود.

#### - ValueList

اختیاری است. مجموعه‌ای از مقادیر را برای dependentField حساب می‌کند تا این فیلد را به تنظیم مجموعه مقادیر مجاز مشخص شده در allowedValueDescriptor، مجبور کند. اجزای متعدد ValueList نباید مشخص شوند و نباید شامل anyValue شوند و شامل اجزای فرعی زیر است:

#### الف- Value

الزامی. Xsd:any Type مقدار مجاز این فیلد را تعریف می‌کند. مقادیر مجاز، معمولاً توسط کمیته کاری Upnp AV تعریف می‌شوند. اما اگر کمیته کاری اجازه دهد فروشنده‌ها مجاز هستند مقادیر مجاز مشخص شده توسط فروشنده را اضافه کنند. یک جزء مقدار خالی به این معناست که وقتی dependentField خالی است این فیلد باید شامل یکی از مقادیر مجاز مشخص شده در این allowedValueDescriptor باشد.

#### - ValueRange

اختیاری است. بازه و تفکیکی را برای مجموعه‌ای از مقادیر dependentField تعریف می‌کند که این فیلد را به مجموعه‌ای از مقادیر مجاز تعریف شده در این allowedValueDescriptor تحمیل می‌کند. فقط برای



انواع اطلاعات عددی، معتبر است. اجزای چندگانه valueRange مجاز است مشخص شوند. نباید شامل جزء anyValue باشد. شامل اجزای فرعی زیر است:

#### الف - minimum

الزامی است. Xsd:string. یک مقدار عددی تکی است که حدّ پایین تر را در بر می گیرد. و باید کمتر از مقدار بیشینه باشد.

یادآوری - هنگام مواجهه با مقادیر ممیز شناور باید دقت شود تا خطاهای تبدیل و/یا گرد کردن باعث مقایسه‌های نادرست نشوند.

#### ب - maximum

الزامی است. Xsd:string. یک مقدار عددی تکی است که حدّ بالاتر را در بر می گیرد. باید بزرگتر از مقدار کمینه باشد.

یادآوری - هنگام مواجهه با مقادیر ممیز شناور باید دقت شود تا خطاهای تبدیل و/یا گرد کردن باعث مقایسه‌های نادرست نشوند.

#### پ - step

اختیاری است. Xsd:string. مقدار عددی تکی مثبت. تفاوت عددی بین مقادیر پشتیبانی شده مجاور را در valueRange نشان می دهد. مقدار مرحله باید دامنه در برگیرنده را از کمینه تا بیشینه با یک عدد صحیح به بخش‌های مساوی، تقسیم کند. به عبارت دیگر،  $maximum = minimum + N * step$  که در آن N عدد صحیح مثبت است. هنگامی که step حذف می شود و نوع اطلاعات dependentField عدد صحیح است، مقدار پیش فرض برای step، یک است. در غیر این صورت هنگامی که step حذف می شود، همه مقادیر بازه‌ی در برگیرنده از کمینه تا بیشینه، معتبر هستند.

یادآوری - هنگام مواجهه با مقادیر ممیز شناور باید دقت شود تا خطاهای تبدیل و/یا گرد کردن باعث مقایسه‌های نادرست نشوند.

#### ۲-۲-۳-۱-۲-۲ minCount

اختیاری است. Xsd: unsignedInt. تعداد کمینه رویدادهای این فیلد که یکی از مقادیر تعریف شده در allowedValueDescriptor را دارد. تعداد کمینه دفعاتی است که (در کل سند XML) این فیلد باید برای یکی از مقادیر تعریف شده در این allowedValueDescriptor تنظیم شود. مقدار پیش فرض، صفر است، به این معنا که سند XML مجاز است هیچ رویدادی از این فیلد را نداشته باشد که مقدارش در یکی از مقادیر تعریف شده در allowedValueDescriptor تنظیم شده است. مقدار ۱ یا بزرگتر، به این معناست که این فیلد، الزامی است و باید دست کم در تعداد دفعات تعریف شده حضور داشته باشد. علاوه بر این، هر کدام از این رویدادها باید در یکی از مقادیر تعریف شده در این allowedValueDescriptor تنظیم شود. نمونه‌های

دیگر این فیلد، مجاز است روی دهند اما باید مقدار تعریف شده در یک allowedValueDescriptor متفاوت را داشته باشند. برای هر فیلد، مقدار minCount باید کمتر یا مساوی minCountTotal باشد.

**maxCount** ۳-۷-۱-۳-۲-۲

اختیاری است. Xsd:string. تعداد بیشینه روی دادن این فیلد که یکی از مقادیر تعریف شده در این allowedValueDescriptor را دارد، تعداد بیشینه دفعاتی را نشان می‌دهد که این فیلد می‌تواند در یکی از مقادیر تعریف شده در این allowedValueDescriptor تنظیم شود. مقدار maxCount باید یا یک عدد صحیح بی‌علامت یا مقدار "UNBOUNDED" باشد. مقدار ۱ یا بیشتر، نشان می‌دهد که این فیلد نباید بیش از تعداد تعریف شده‌ای از دفعات با مقدار تنظیم شده در یکی از مقادیر تعریف شده در allowedValueDescriptor وجود داشته باشد. مقدار پیش فرض ۱ است. مقدار صفر، به این معناست که ساختار داده، نباید شامل هیچ رویدادی از این فیلد باشد، به جز رویدادهایی که مقدارشان در یک allowedValueDescriptor مختلف، مشخص شده است. در این مورد allowedValueDescriptor باید شامل یک allowedvalueList و بدون allowedValueRange باشد.

مقدار «UNBOUNDED» نشان می‌دهد که حدّ از پیش تعیین شده‌ای برای تعداد رویدادهای این فیلد که ممکن است شامل یکی از مقادیر مشخص شده در این allowedValueDescriptor باشد وجود ندارد. مقدار maxCount باید بزرگتر یا مساوی minCount باشد. برای هر وقوع allowedValueDescriptor مقدار maxCount باید کمتر یا مساوی maxCountTotal در این فیلد باشد. نمونه‌های دیگر این فیلد مجاز است روی دهند اما باید مقدار تعریف شده‌ای در allowedValueDescriptor داشته باشند.

**minListSize** ۴-۷-۱-۳-۲-۲

اختیاری است. Xsd:unsignedInt. فقط برای فیلد نوع CSV، معتبر است یعنی هنگامی که ویژگی @CSV، در جز فرعی name در فیلد، تعریف می‌شود. تعداد کمینه ثبت‌های هر نمونه از این فیلد CSV باید شامل یکی از مقادیر تعریف شده در allowedValueDescriptor باشد. مقدار پیش فرض صفر به این معناست که هرگاه این فیلد وجود دارد، مجاز است شامل هیچ ثبتی که در این allowedValueDescriptor تعریف شده است نباشد. مقدار ۱ یا بیشتر، به این معناست که این فیلد در زمانی که وجود دارد باید حداقل شامل تعداد تعریف شده‌ای از ثبت‌هایی باشد که مقدارش در allowedValueDescriptor تعریف شده است. نمونه‌های دیگر این فیلد، ممکن است روی دهند اما باید مقدار تعریف شده‌ای در allowedValueDescriptor متفاوت داشته باشند.

**maxListSize** ۵-۷-۱-۳-۲-۲

اختیاری است. Xsd:string. فقط برای فیلد نوع CSV معتبر است یعنی زمانی که ویژگی @CSV در جز فرعی نام فیلد، تعریف می‌شود. تعداد بیشینه ثبت‌های هر نمونه فیلد CSV برای در برگرفتن یکی از مقادیر

تعریف شده در `allowedValueDescriptor` مجازند. مقدار `maxListSize` باید عدد صحیح بدون علامت یا مقدار "UNBOUNDED" باشد. مقدار پیش فرض، عدد یک است، به این معنا که فیلد CSV نباید شامل بیش از یک ثبت باشد که مقدارش در این `allowedValueDescriptor` تعریف شده است. مقدار صفر، به این معناست که هیچ ثبتی در هیچ نمونه‌ای از این فیلد CSV برای در برگرفتن یکی از مقادیر تعریف شده در `allowedValueDescriptor` مجاز نیست. در این مورد، `allowedValueDescriptor` باید شامل `allowedValueList` با یک `allowedValue` خالی و تکی و بدون `allowedValueRange` باشد. مقدار UNBOUNDED نشان می‌دهد که هیچ حدّ از پیش تعیین شده‌ای در تعداد ثبت‌هایی که شامل یکی از مقادیر تعریف شده در این `allowedValueDescriptor` هستند وجود ندارد. نمونه‌های دیگر این فیلد، ممکن است روی دهند اما باید مقدار تعریف شده در `allowedValueDescriptor` متفاوت را داشته باشند. مقدار `maxListSize` باید بزرگتر یا مساوی `minListSize` و کمتر یا مساوی `maxListSizeTotal` باشد.

#### ۶-۷-۱-۳-۲-۲ `defaultValue`

اختیاری است `Xsd:anyType`. اگر هیچ مقداری در سند XML وجود نداشته باشد، مقدار پیش فرض اختصاص شده به این فیلد را تعریف می‌کند. محتوا باید با نوع داده (`dataType`) این فیلد هماهنگ باشد و باید به مجموعه‌ای از مقادیر مجاز تعریف شده در این `allowedValueDescriptor` متعلق باشد یعنی `allowAny` و `allowedValueList` و `allowedValueRange`. اگر این فیلد به صورت `dependentField` در فیلد دیگر ظاهر شود، آن جزء `dependentField` باید شامل ویژگی `defaultDependency` با مقدار ۱ باشد.

#### ۷-۷-۱-۳-۲-۲ `allowAny`

اختیاری است. `Xsd:string` وجود این جزء نشان می‌دهد که این فیلد ممکن است برای هر مقدار مجاز شده با این نوع اطلاعات فیلد، تنظیم شود. محتوای این جزء باید خالی باشد. `allowAny` نباید شامل اجزای `allowedValueList` یا `allowedValueRange` شود.

#### ۸-۷-۱-۳-۲-۲ `allowedValueList`

اختیاری است. مجموعه‌ای از مقادیری را که برای این فیلد است و در معرض محدودیت‌های تعریف شده توسط جزء `dependentField` است، به شمار می‌آورد. چندین `allowedValueList` نباید تعریف شوند. `allowedValueList` نباید شامل جزء `allowAny` شود و اجزای فرعی زیر را دارد:

#### الف - `allowedValue`

الزامی است. `Xsd:anyType`. یکی از مقادیری را تعریف می‌کند که برای این فیلد، مجاز است. مقادیر مجاز معمولاً توسط کمیته کاری UPnP AV مشخص می‌شوند. اما، اگر کمیته مربوط اجازه دهد، فروشندگان، مقادیر مجاز تعریف شده توسط فروشنده را اضافه می‌کنند. جزء خالی `allowedValue` به این معناست که محتوای این فیلد مجاز است خالی باشد. `allowedValueList` فقط با یک `allowedValue` خالی، به این

معناست که وقتی این فیلد، وجود دارد، مقدارش باید خالی باشد. برای فیلد نوع CSV (@CSV)، ثبت allowedValue، مقدار احتمالی ثبت در فهرست CSV را نشان می‌دهد. این امر یکی از ترکیبات احتمالی مقادیر برای کل فهرست CSV را نشان نمی‌دهد.

**یادآوری** - برای فیلد ناهمگن نوع CSV، احتساب همه مقادیر مجازی که امکان‌پذیرند عملی نیست. در این مورد، مشخص کردن allowAny، توصیه می‌شود.

#### ۹-۷-۱-۳-۲-۲ allowedValueRange

اختیاری است. بازه و تفکیک مجموعه‌ای از مقادیر عددی را تعریف می‌کند که برای فیلدی که در معرض محدودیت‌های تعریف شده توسط جزء dependentField هستند، مجاز است. فقط برای انواع داده عددی، معتبر است. چندین جزء allowedValueRange ممکن است تعریف شوند. نباید شامل جزء allowAny شود. شامل اجزای فرعی زیر است:

#### الف - minimum

الزامی است. Xsd:String مقدار عددی تکی و شامل حدپائین است. باید کوچک‌تر از مقدار بیشینه باشد. **یادآوری** - هنگام مواجهه با مقادیر ممیز شناور باید دقت شود تا خطاهای تبدیل و/یا گرد کردن باعث مقایسه‌های نادرست نشوند.

#### ب - maximum

الزامی است. Xsd:String مقدار عددی تکی و شامل حد بالا است. باید بزرگ‌تر از مقدار کمینه باشد. **یادآوری** - هنگام مواجهه با مقادیر ممیز شناور باید دقت شود تا خطاهای تبدیل و/یا گرد کردن باعث مقایسه‌های نادرست نشوند.

#### پ - step

اختیاری است. Xsd:string مقدار عددی تکی مثبت است که تفاوت عددی بین مقادیر معتبر مجاور را در allowedValueRange تعریف می‌کند. مقدار step باید بازه شمول را از کمینه تا بیشینه با یک عدد صحیح به بخش‌های مساوی تقسیم کند. به عبارت دیگر،  $maximum = minimum + N * step$  که در آن N، عدد صحیح مثبتی است. هنگامی که step، حذف می‌شود و نوع اطلاعات فیلد عدد صحیح است، مقدار پیش‌فرض step، یک است. در غیر این صورت اگر step حذف شود، تمامی مقادیر در بازه شمول، از کمینه تا بیشینه باید پشتیبانی شود.

**یادآوری** - هنگام مواجهه با مقادیر ممیز شناور باید دقت شود تا خطاهای تبدیل و/یا گرد کردن باعث مقایسه‌های نادرست نشوند.

### ۳ طرح ساختار داده صوتی تصویری

طرح XML ساختار داده AV، ساختار سند AVDT را تعریف می‌کند. یک پرونده قابل خواندن توسط ماشین که شامل این طرح است می‌تواند در [AVDT-XSD] یافت شود. اگرچه طرح XML، شامل هیچ سازوکار توسعه‌پذیری نیست (مثلاً در برگرفتن نشان <Xsd:any>)، اسناد AVDT برای در بر گرفتن اجزای اضافی XML و/یا ویژگی‌هایی فرای آنچه که در طرح مشخص شده است، مجازند. این امر برای دنباله‌های تعریف شده توسط فروشنده و/یا برای ارتقای (استاندارد شده) آتی ساختار AVDT اجازه می‌دهد. در نتیجه هنگام تحلیل سند AVDT، هر جزء و/یا ویژگی تعریف نشده‌ای باید نادیده گرفته شود.

هر ساختار داده (شناسایی شده توسط اجزای <dataStructureType> و <ContextID>) توسط سند AVDT شرح داده شده است که به نوبه خود، معرف نسخه خاصی از طرح AVDT است. از آنجا که طرح AVDT در طول زمان، ارتقا می‌یابد، به هر نسخه، عدد منحصر به فردی اختصاص داده می‌شود که در بخش آخر طرح URN نشان داده شده است. (به بند ۱-۴-۲ نام‌های فضای نام، نسخه فضاهای نام و نسخه طرح رجوع شود):

```
xsi:schemaLocation="
urn:schemas-upnp-org:av:avdt
http://www.upnp.org/schemas/av/avdt.xsd"
```

که در آن عدد ۱ پس از «v»، شماره نسخه است. هر به‌روزرسانی نسخه طرح AVDT باید به صورت پس‌رو با نسخه قبلی سازگار باشد. به خصوص، مجاز است اجزای XML و/یا ویژگی‌ها به نسخه‌های اخیر طرح AVDT اضافه شوند اما نباید حذف شوند. در نتیجه هنگام بررسی عدد نسخه طرح، پیاده‌سازی‌ها به احتمال زیاد مایل هستند مقایسه بین عدد نسخه‌ی بزرگ‌تر یا مساوی را فقط به‌جای واری‌تساوی نسخه‌ها انجام دهند.