

Poznań University of Technology  
Institute of Computing Science



inż. Paweł Szafer

# UPNP DLNA

Master's Thesis

Supervisor: prof. dr hab inż. Jan Brzeziński  
Poznań, 2012

# Spis treści

|   |           |
|---|-----------|
| <b>Wstęp</b>  | <b>3</b>  |
| <b>I Opis technologii</b>   | <b>4</b>  |
| <b>1 Standardy UPnP i DLNA</b>                                      | <b>5</b>  |
| 1.1 Opis standardu UPnP . . . . .                                   | 5         |
| 1.2 Stos protokołów UPnP . . . . .                                  | 6         |
| 1.2.1 Adresowanie . . . . .   | 8         |
| 1.2.2 Wykrywanie . . . . .  | 8         |
| 1.2.2.1 Wykrywanie: Ogłaszanie . . . . .                            | 9         |
| 1.2.3 Opis urządzeń i usług . . . . .                               | 12        |
| 1.2.4 Kontrola . . . . .  | 13        |
| 1.2.5 Zdarzenia . . . . .   | 14        |
| 1.2.6 Prezentacja . . . . .   | 15        |
| 1.3 Klasy urządzeń multimedialnych UPnP . . . . .                   | 16        |
| 1.3.1 Serwer multimedialny . . . . .                                | 18        |
| 1.3.2 Odtwarzacz multimediiów oraz odtwarzacz renderujący . . . . . | 19        |
| 1.3.3 Punkt kontrolny oraz kontroler multimedialny . . . . .        | 21        |
| 1.3.4 Odmiany mobilne . . . . .                                     | 21        |
| <b>2 Multimedia w UPnP AV/DLNA</b>                                  | <b>23</b> |
| 2.1 Odtwarzanie multimediiów . . . . .                              | 24        |
| 2.2 Klasy multimediiów UPnP . . . . .                               | 25        |
| 2.2.1 Klasa <i>object</i> oraz klasa <i>item</i> . . . . .          | 26        |
| 2.2.2 Klasa kontener . . . . .                                      | 27        |
| 2.2.3 Klasa obraz . . . . .   | 27        |
| 2.2.4 Klasa audio . . . . .   | 28        |
| 2.2.5 Klasa wideo . . . . .   | 30        |
| 2.3 Rodzaje multimediiów . . . . .                                  | 31        |
| 2.3.1 Muzyka . . . . .  | 32        |
| 2.3.2 Filmy . . . . .   | 32        |
| 2.3.3 Kodowanie i kompresja filmów . . . . .                        | 33        |
| <b>II Implementacja serwera UPnP</b>                                | <b>35</b> |
| <b>3 Oprogramowanie serwera UPnP AV</b>                             | <b>36</b> |
| 3.1 Wymagania wstępne . . . . .                                     | 36        |

|          |  |           |
|----------|--|-----------|
| 3.2      | Wykorzystane narzędzia . . . . .                             | 37        |
| 3.2.1    | Framework'i UPnP . . . . .                                   | 38        |
| 3.3      | Opis programu . . . . .                                      | 38        |
| 3.4      | Funkcje programu . . . . .                                   | 39        |
| 3.4.1    | Biblioteka multimediiów . . . . .                            | 39        |
| 3.4.2    | Miniatury multimediiów . . . . .                             | 41        |
| 3.4.3    | Zarządzanie aplikacją . . . . .                              | 41        |
| 3.5      | Budowa wewnętrzna modułu serwera UPnP AV . . . . .           | 44        |
| 3.5.1    | Klasa demona . . . . .                                       | 45        |
| 3.5.2    | Klasa główna serwera UPnP . . . . .                          | 45        |
| 3.5.2.1  | Obiekt MediaStore . . . . .                                  | 46        |
| 3.5.2.2  | Obiekt MediaItem . . . . .                                   | 46        |
| 3.5.3    | Modyfikacja frameworka UPnP Coherence . . . . .              | 47        |
| 3.5.3.1  | Język DIDL-Lite . . . . .                                    | 48        |
| 3.5.4    | Klasa serwera RPC . . . . .                                  | 49        |
| 3.6      | Budowa wewnętrzna modułu zarządzającego . . . . .            | 49        |
| 3.6.1    | Model . . . . .  | 49        |
| 3.6.2    | Widok . . . . .  | 50        |
| 3.6.3    | Kontroler . . . . .  | 50        |
| <b>4</b> | <b>Testy</b>   | <b>52</b> |
| 4.1      | Sposób testowania aplikacji . . . . .                        | 52        |
| 4.1.1    | Test wyszukiwania urządzenia . . . . .                       | 54        |
| 4.1.2    | Test przeglądania zawartości . . . . .                       | 54        |
| 4.1.3    | Test odtwarzania materiałów . . . . .                        | 54        |
| 4.1.4    | Polecenia UPnP AV i DLNA . . . . .                           | 55        |
| 4.2      | Napotkane problemy . . . . .                                 | 55        |
| 4.3      | Wyniki testów . . . . .                                      | 55        |
| 4.3.1    | Test wyszukiwania urządzenia . . . . .                       | 55        |
| 4.3.2    | Test przeglądania zawartości . . . . .                       | 56        |
| 4.3.3    | Test odtwarzania materiałów . . . . .                        | 56        |
| <b>5</b> | <b>Przegląd oprogramowania UPnP AV i DLNA</b>                | <b>58</b> |
| 5.1      | Przegląd oprogramowania serwerów UPnP AV i DLNA . . . . .    | 58        |
| 5.1.1    | Oprogramowanie Samsung AllShare . . . . .                    | 58        |
| 5.1.2    | Nero MediaHome 4 . . . . .                                   | 59        |
| 5.1.3    | MiniDLNA . . . . .   | 59        |
| 5.1.4    | Wild Media Server . . . . .                                  | 60        |
| 5.1.5    | PS3 Media Server . . . . .                                   | 60        |
| 5.1.6    | Coherence . . . . .  | 60        |
| 5.2      | Przegląd oprogramowania odtwarzaczy DLNA i UPnP AV . . . . . | 61        |
| 5.2.1    | Oprogramowanie dla komputerów klasy PC . . . . .             | 61        |
| 5.2.2    | Oprogramowanie w sprzęcie audio-wideo . . . . .              | 61        |
| 5.2.3    | Obsługa standardu . . . . .                                  | 62        |
|          | <b>Podsumowanie</b>  | <b>64</b> |
|          | <b>Bibliografia</b>  | <b>65</b> |

# Wstęp

Technologie udostępniania multimediów w sieciach, zarówno lokalnych, jak i sieci Internetowej stają się coraz bardziej powszechne. Jakkolwiek dostęp do multimediów znajdujących się w Internecie przy pomocy komputera jest bardzo prosty, to problem powstaje w przypadku chęci odtworzenia na ekranie telewizora filmu z wakacji, który znajduje się na dysku komputera. Dotychczas trzeba było nagrać taki materiał na dysk DVD, bądź w nowszych telewizorach “najprostszym” rozwiązaniem było podłączenie dysku USB. Sieci komputerowe LAN zostały stworzone w celu łatwej komunikacji między stacjami roboczymi. Udostępnianie plików było i jest jedną z podstawowych usług, do których można wykorzystywać sieć lokalną. Jednakże wykorzystywane do dzisiaj protokoły udostępniania plików nie zostały zaprojektowane do odtwarzania multimediów i do ich oddzielnego rozpoznawania. Odtwarzacz sieciowy DLNA potrafi odtworzyć film, muzykę, obraz, jednak w większości przypadków nie poradzi sobie skutecznie z dokumentami, arkuszami kalkulacyjnymi i innymi nietypowymi dla niego plikami. Natomiast użytkownik takiego odtwarzacza w przypadku filmów oczekuje możliwości wyboru ścieżki dźwiękowej, napisów oraz tak jak w przypadku odtwarzaczy kaset VHS czy płyt DVD możliwości przewijania materiałów. W ostatnich latach wraz z upowszechnianiem się w gospodarstwach domowych różnego rodzaju urządzeń multimedialnych typu komputery PC, laptopy, netbooki, tablety, smartfony, odtwarzacze Blu-Ray, DVD, telewizory, ramki do zdjęć, przenoszenie multimediów między wieloma urządzeniami stało się uciążliwe. Co gorsza odtworzenie

## I Opis technologii

# Standardy UPnP i DLNA

Technologia UPnP, której skrót rozwijany jest jako Universal Plug and Play, co w łatwy sposób można zrozumieć jako podłącz i graj (lub odtwarzaj) jest zbiorem protokołów sieciowych, których celem powstania była możliwość wzajemnego wykrywania się przez urządzenia sieciowe takie jak komputery PC, drukarki, punkty dostępu WiFi, routery czy urządzenia mobilne.

UPnP ma za zadanie umożliwić użytkownikowi na łatwe wykrywanie urządzeń sieciowych oraz wykorzystywanie ich możliwości. Jest ono dedykowane dla małych sieci domowych w celach takich jak udostępnianie danych, komunikacja, drukowanie czy szeroko pojęta rozrywka. Niniejsza praca jest poświęcona w głównej mierze ostatniemu zagadnieniu (rozrywka), którego specyfikacja nazywana UPnP A/V (ang. UPnP Audio/Video) jest rozszerzeniem podstawowego standardu UPnP. UPNP AV została stworzona do odtwarzania plików multimedialnych. Definiuje ona takie pojęcia jak serwery multimedialne (ang. Media Servers), Media Renderers??, odtwarzacze multimedii (ang. Media Players). Określone zostały w bardzo elastyczny sposób metody przesyłania multimedii. Z tego powodu UPNP AV nie rozpropagowało się w swojej powszechnej wersji wśród producentów sprzętu elektronicznego. Każdy programista i każda firma mogła zaimplementować odtwarzacz lub serwer w taki sposób, że mogły one ze sobą nie współpracować (pomimo spełnienia wymogów standardu) lub powinny być nad wyrost skomplikowane. DLNA (ang. Digital Living Network Alliance) jest międzynarodową organizacją, powstałą w 2003 roku z inicjatywy firmy Sony. Stworzyła ona standard o tej samej nazwie (DLNA), który korzysta z protokołu UPNP A/V. Standard ten jest jednak bardziej restrykcyjny (np w postaci określenia formatów multimedii odtwarzanych przez urządzenia), dzięki czemu łatwiejszy do implementacji w większości urządzeń, a ponadto wprowadził kilka innowacyjnych cech takich jak ochrona DRM (ochrona przed kopiowaniem).

## 1.1 Opis standardu UPNP

W 1999 została założona, w głównej mierze dzięki dotacjom firmy Microsoft, fundacja UPnP. Powołana została w celu stworzenia standardu, który umożliwi interakcję urządzeń oraz usług w obrębie sieci lokalnej oraz proste określanie ich możliwości (np. drukowanie).

UPnP jest oparty o architekturę klient - serwer, gdzie urządzenia - serwery, oferują swoje usługi dla klientów (UPnP Control Points). UPnP główny nacisk kładzie na brak konfiguracji urządzeń przez użytkownika, zgodnie z częścią nazwy zarówno fundacji, jak i standardu - plug and play (podłącz i działaj). Opisująca fundacja ustandaryzowała sposób komunikacji szeregu urządzeń, począwszy od włączników światła, drukarek, kończąc na odtwarzaczach audio-wideo. Architektura UPNP AV została stworzona do przesyłania materiałów multimedialnych. UPNP AV określa sposób komunikacji, między serwerem multimedii, a odtwarzaczem. Zbiór protokołów został zaprojektowany tak, aby być niezależny od rodzaju urządzenia, odtwarzanej treści i sposobu przesyłania danych.

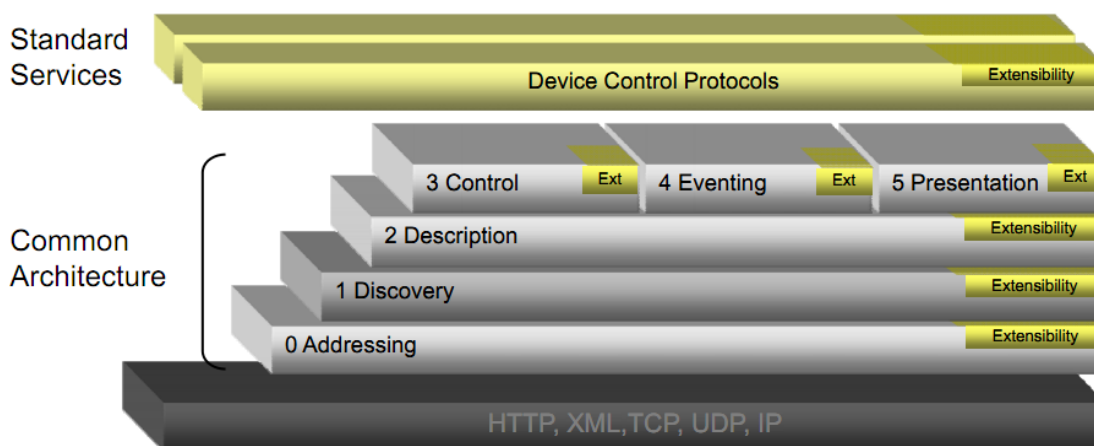
UPnP AV określa sposób komunikacji, między serwerem multimediiów, a odtwarzaczem. Zbiór protokołów został zaprojektowany tak, aby być niezależny od rodzaju urządzenia, odtwarzanej treści i sposobu przesyłania danych.

## 1.2 Stos protokołów UPnP

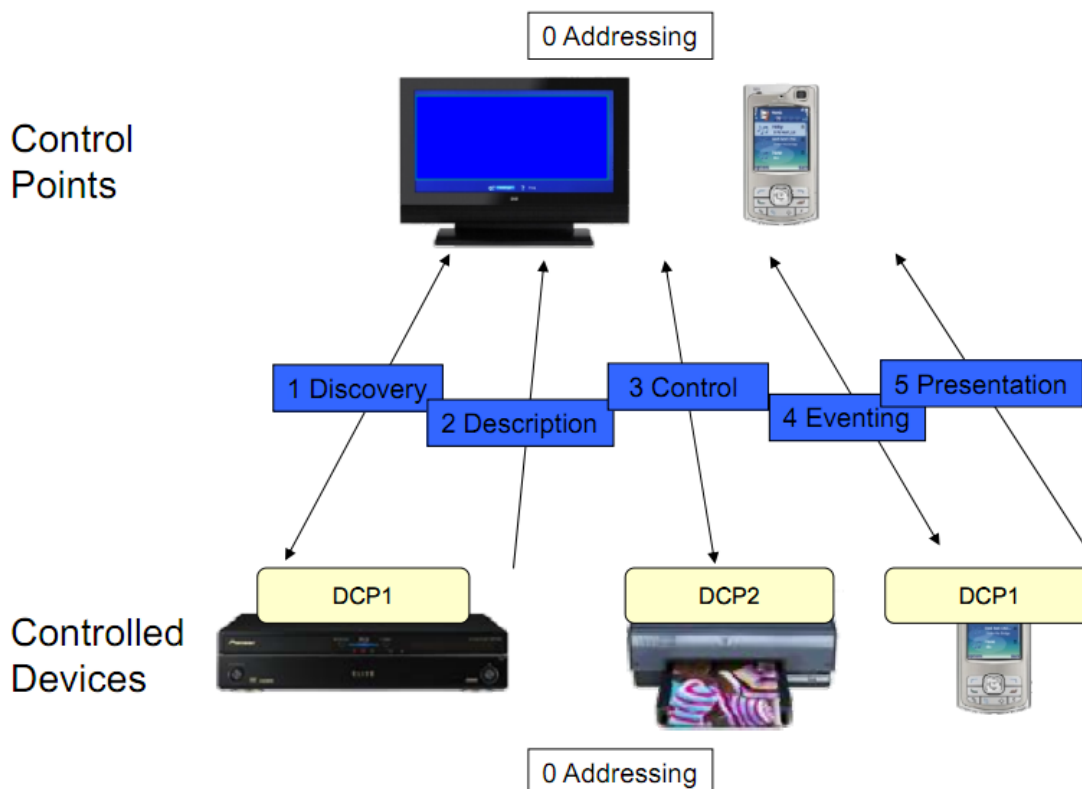
Głównym wymaganiem UPnP jest działanie w obrębie sieci IP, niezależnie od sposobu komunikacji. Połączenie może zostać zrealizowane poprzez sieć Ethernet, połączenia bezprzewodowe takie jak WiFi 802.11abgn, Bluetooth (pomijany jest tu fakt braku możliwości technicznych niektórych protokołów do szybkiego przesyłania danych, co w przypadku niektórych typów multimediiów, takich jak filmy czy muzyka, jest niezbędne).

Raport techniczny UPnP definiuje jedynie sposoby działania w obrębie sieci obsługiwanej przez protokół IPv4. Aneks [10] do głównego raportu określa różnice jakie występują, gdy urządzenie działa w obrębie sieci IPv6. Jeżeli nie napisano inaczej, w niniejszej pracy autor jako pierwszą pozycję określa przypadek sieci IPv4, natomiast drugi przypadek to sieć zgodna z protokołem IPv6.

Rys. 1.1: Protokoły



Rys. 1.2: Komunikacja



Urządzenia UPnP można sklasyfikować w dwóch kategoriach, punktów kontrolnych (ang. Control Points) oraz urządzeń kontrolowanych (ang. Controlled Devices).

W protokole UPnP definiujemy następujące czynności wykonywane przez urządzenia:

- adresowanie IP (ang. IP addressing),
- wykrywanie (ang. discovery),
- opis usługi (ang. description),
- kontrola (ang. control),
- powiadomienia o zdarzeniach(ang. eventing lub event notification),
- prezentacja (ang. presentation).

(<http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0-20080424.pdf>)

Architektura protokołu wykorzystuje stos protokołów przedstawiony w tabeli

Tabela 1.1: Test tabeli

|             |        |      |      |
|-------------|--------|------|------|
| UPNP Vendor |        |      |      |
| UPNP Forum  |        |      |      |
| UPNP Device |        |      |      |
| SSDP        |        | GENA | SOAP |
| HTTPMU      | HTTP U | HTTP | HTTP |
| IP          |        |      |      |

Pierwsza, najwyższa warstwa są to wiadomości zawierające specyficzne informacje dostarczane przez producenta urządzenia dotyczące jego funkcjonalności czy możliwości. Kolejnym el-



ementem są informacje zdefiniowane przez organizację UPnP Forum. Wszystkie te informacje są przekazywane poprzez następujące protokoły:

- SSDP (ang. Simple Service Discovery Protocol),
- GENA (ang. General Event Notification Architecture),
- SOAP (ang. Simple Object Access Protocol).

Do wzajemnego wykrywania się urządzeń UPnP wykorzystuje protokół SSDP. Urządzenia kontrolowane, które można nazwać serwerami usług wykorzystują SSDP do rozgłaszania swojego istnienia. W przypadku punktów kontrolnych SSDP jest wykorzystywany do wykrywania urządzeń pierwszego typu.

W protokole SSDP wykorzystuje się rozgłaszanie HTTP typu unicast lub multicast, z wykorzystaniem UDP (ang. User Datagram Protocol) jako protokołu transportu. Standardowo adresem multicast jest adres 239.255.255.255 oraz port numer 1900.[1]

Drugi protokół transmisji tj. GENA wykorzystywany jest do przekazywania powiadomień o zdarzeniach. W tym przypadku typowo wykorzystywany jest protokół HTTP w sposób “tradycyjny” (z wykorzystaniem protokołu TCP) oraz transmisja UDP typu multicast.

Punkty kontrolne UPnP do wywoływania pożądanej usługi i kontrolowania urządzeń wykorzystują protokół SOAP.

### 1.2.1 Adresowanie

Zarówno serwer, jak i klient UPnP musi komunikować się z wykorzystaniem protokołu IP. Organizacja UPnP Forum zdefiniowała, że każde urządzenie, które spełnia wymagania specyfikacji musi posiadać oprogramowanie serwera lub klienta protokołu DHCP (ang. Dynamic Host Configuration Protocol). Podczas podłączania się do sieci podejmuje próbę uzyskania adresu IP od serwera DHCP (jeżeli urządzenie nie jest serwerem DHCP). Jeżeli urządzenie nie otrzyma adresu IP powinno wykorzystać tzw. autoadresowanie IP (urządzenie wybiera adres z określonej puli np. 169.254/16 lub w przypadku IPv6 FE80::/64). Raport techniczny stworzony przez organizację UPnP Forum określa dokładnie w jaki sposób urządzenie przydziela automatycznie adres IP, tak aby uniknąć kolizji.

Zarówno w praktyce w większości urządzeń, jak i w aplikacji, którą opisuje niniejsza praca problem adresowania nie jest najważniejszy, gdyż jest rozwiązywany przez oprogramowanie dołączone do sprzętu, bądź systemu operacyjnego.

Wywołania punktów kontrolnych odbywają się poprzez wykorzystanie adresów SOAP SSDP UPNP\_FORUM\_PUBLIC\_RYSUNKI

### 1.2.2 Wykrywanie

Proces wykrywania polega na rozgłaszaniu przez nowo podłączone urządzenia informacji o swoim istnieniu. Wykorzystywany jest do tego protokół SSDP oraz adres 239.255.255.250 oraz port 1900 (dla IPv6 - [FF02::C]:1900).

Na potrzeby protokołu UPnP wyróżniane są następujące komunikaty SSDP (**rys nr 2.3**):

- ogłaszanie (ang. *advertise*),
- szukanie (ang. *search*),
- odpowiedzi (ang. *responses*).

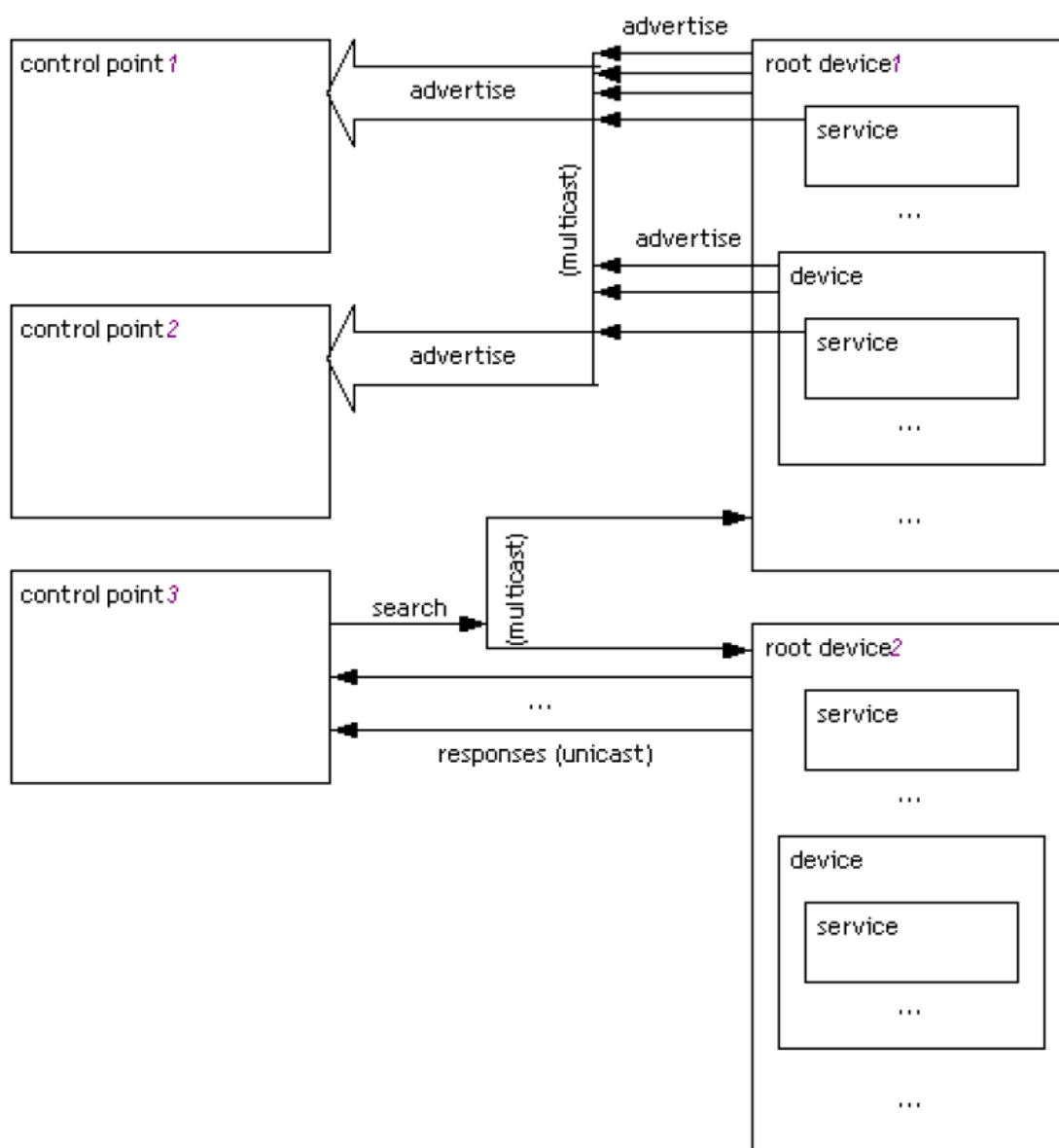
Punkty kontrolne nasłuchują urządzeń na porcie 1900 lub rozgłaszają pakiet z danymi o typie, identyfikatorzeżądanego urządzenia lub usługi. Urządzenia oferujące swoje usługi wysyłają na wcześniej wspomniany adres (239.255.255.250:1900) pakiety SSDP przy pomocy rozgłaszania typu *multicast*.

Odpowiedzi są wysyłane przy pomocy rozgłaszania typu *unicast*.

Urządzenie nowo dołączone do sieci wysyła rozgłoszenie o swoim istnieniu, urządzenia wbudowane w nie również wysyłają rozgłoszenie o sobie oraz o usługach, które świadczą. Kontrolowane odłączenie od sieci wiąże się z obowiązkiem ogłoszenia zdarzenia przez urządzenie. Przyjęta jest w tym przypadku zasada, że powinno ono wysłać tyle komunikatów odłączenia, ile wysłało komunikatów ogłaszających swoje działanie.

Rozgłaszanie wiąże się niestety z obciążaniem sieci, często niepotrzebnym. W tym celu zalecany czas życia pakietu (*TTL* - ang. *time-to-live*) dla każdej wiadomości wynosi 4 (powinna istnieć możliwość modyfikacji tej wartości).

Rys. 1.3: Sposoby wykrywania urządzeń



### 1.2.2.1 Wykrywanie: Ogłaszanie

W głównej części tego podrozdziału został opisany zarys sposobu ogłaszania się urządzeń w sieci. Jest to jednak bardzo newralgiczna cecha tego protokołu, gdyż bez możliwości wzajemnego wykry-

wania nie ma możliwości, aby urządzenia efektywnie się ze sobą komunikowały. Każde urządzenie/usług wysyła pakiety ogłaszające (*advertise*) swoje działanie na adres 239.255.255.250:1900. Punkty kontrolne nasłuchują na porcie 1900, oczekując na komunikaty wysyłane przez urządzenia. Opisywane komunikaty muszą być wysyłane często oraz co określony interwał czasowy, gdyż zarówno urządzenie, jak i punkt kontrolny mogą zostać dołączone do sieci w różnym czasie.

Każda wiadomość “ogłaszająca” zawiera możliwości urządzenia (np. odtwarzanie muzyki). Ilość wiadomości jest zależna od ilości wbudowanych urządzeń/usług w główne urządzenie. Urządzenia wbudowane w komunikatach dołączają informacje o urządzeniu nadrzędnym (np. informacja o systemie operacyjnym). Oprócz wcześniej przedstawionych cech, komunikaty powinny zawierać czas do wygaśnięcia ogłoszenia (jeżeli po czasie wygaśnięcia urządzenie nadal chce oferować swoje usługi, wówczas powinno wysłać nowy komunikat z nowym czasem wygaśnięcia). Dzięki takiemu podejściu, pomimo, że urządzenia powinny ogłaszać swoje odłączenie od sieci, to nawet, gdy komunikat wyłączenie nie może zostać wysłany, wiadomość rozgłaszająca wygaśnięcie i nie zostanie przedłużona. Ogranicza to możliwość istnienia w sieci usług już nie działających.

Możliwe jest wysyłanie i odbieranie ogłoszenia przez sprzęt zgodny ze specyfikacją UPnP, jedynie w przypadku, gdy komunikat jest stworzony według schematu z tabeli 1.2

**Tabela 1.2:** Stos protokołu UPnP w przypadku rozgłaszania – do poprawienia

|                               |
|-------------------------------|
| producent oprogramowania UPnP |
| organizacja UPnP              |
| architektura urządzenia UPnP  |
| komunikat SSDP                |
| HTTPMU ( <i>multicast</i> )   |
| UDP                           |
| IP                            |

Pierwsza warstwa opisywanego stosu zawiera informacje specyficzne dla dostawcy urządzenia. Najczęściej producenci dołączają w tym miejscu adres URL do opisu urządzenia, bądź też identyfikator urządzenia (UUID - *ang. Universally unique identifier*). Kolejną warstwą stosu są informacje dostarczane przez organizację UPnP Forum, np. typ urządzenia. Kolejnym dwie warstwy zostaną dokładnie opisane w niniejszym dokumencie, a ich pełen opis znajduje się w specyfikacji protokołu UPnP.

**Rys. 1.4:** Przykładowy komunikat przyłączający do sieci:

|  |
|--|
| NOTIFY * HTTP/1.1  |
| HOST: 239.255.255.250:1900                                   |
| CACHE-CONTROL: max-age = seconds until advertisement expires |
| LOCATION: URL for IPP Printer with 'ipp' scheme              |
| NT: search target  |
| NTS: ssdp:alive  |
| SERVER: OS / version, IPP / 1.1, product / version           |
| USN: advertisement UUID                                      |

Trzy pozostałe warstwy nie są tematem niniejszej pracy. Warto jedynie zwrócić uwagę na fakt, iż komunikaty HTTP nie są przesyłane typowo przez protokół TCP, a przez protokół UDP. Jak zostało wspomniane wcześniej urządzenie dołączone do sieci rozgłasza swoje istnienie. Komunikat rozgłaszający wykorzystuje metodę NOTIFY protokołu HTTP. W każdym komunikacie dołączane są informacje w następujących nagłówkach:

- NT - Notification Type - typ ogłoszenia,
- NTS - Notification Subtype - podtyp ogłoszenia,
- USN - Unique Service Name - Unikalna nazwa usługi / UUID
- LOCATION - URL z rozszerzonymi informacjami o urządzeniu,
- CACHE-CONTROL - czas do wygaśnięcia ogłoszenia

Opisywana metoda protokołu HTTP (NOTIFY) wykorzystywana jest przy ogłaszaniu podłączenia, bądź też odłączenia od sieci. W komunikacie podłączenia pole *NTS* jest podtypu *ssdp:alive*, natomiast rozłączenie cechuje podtyp *ssdp:byebye*.

Organizacja stanowiąca zasady protokołu UPnP w przypadku wykrywania urządzeń, poza wzajemnym informowaniem się o obecności (bądź też nieobecności), zagwarantowała również szukanie się wzajemne urządzeń UPnP. Do wyszukiwania służy metoda M-SEARCH, której ogólny format przedstawia **tabela nr.** Pola *HOST* i *MAN* nie mogą zostać zmienione i zawsze muszą mieć taką wartość jak podana w tabeli. Pole *MX* określa maksymalny czas oczekiwania w sekundach. Jego wartość powinna mieścić się w przedziale 1, a 120. Ostatnie pole, *ST* (*Search Target*) określa rodzaj wyszukiwanego urządzeń. Dostępne rodzaje to:

- ssdp:all - wyszukiwanie wszystkich urządzeń i usług,
- upnp:rootdevice - wyszukiwanie jedynie urządzeń głównych,
- uuid:device-UUID - wyszukanie konkretnego urządzenia,
- urn:schemas-upnp-org:device:deviceType:v - wyszukiwanie urządzenia według typu. Lista typów możliwych do odnalezienia znajduje się w ...
- urn:schemas-upnp-org:device:serviceType:v - wyszukiwanie urządzenia według typu usługi, którą świadczy. Lista typów **jak wyżej**
- urn:domain-name:device:deviceType:v - takie jak przeszukiwanie urządzenia według typu, z tym, że nazwę domeny typ urządzenia i jego wersję definiuje producent urządzenia UPnP,
- urn:domain-name:device:serviceType:v - takie jak przeszukiwanie urządzenia według typu usługi, z tym, że nazwę domeny typ urządzenia i jego wersję definiuje producent urządzenia UPnP.

Bezpośrednio powiązana z komunikatem M-SEARCH jest odpowiedź na nią, którą urządzenie odpowiadające wysyła na adres źródłowy wcześniejszego pakietu. Odpowiedź jest zgodna z protokołem HTTP. Jej schemat widnieje w **tabeli nr 4.** Odpowiedź nie zawiera treści (*HTTP BODY*), lecz musi posiadać jedną pustą linię poniżej nagłówka. Zarówno dla metody *M-SEARCH*, jak i odpowiedzi na nią nie ma potrzeby określania czasu życia pakietu *TTL*.

**Tabela 1.3:** Metoda M-Search

```
M-SEARCH * HTTP/1.1
HOST: 239.255.255.250:1900
MAN: „ssdp:discover”
MX: seconds to delay response
ST: search target
```

**Tabela 1.4:** Odpowiedź na metodę M-SEARCH

HTTP/1.1 200 OK  
 CACHE-CONTROL: max-age = seconds until advertisement expires  
 DATE: when response was generated  
 EXT:  
 LOCATION: URL for UPnP description for root device  
 SERVER: OS/version UPnP/1.0 product/version  
 ST: search target **OPISAC STRONA 21**  
 USN: advertisement UUID

### 1.2.3 Opis urządzeń i usług

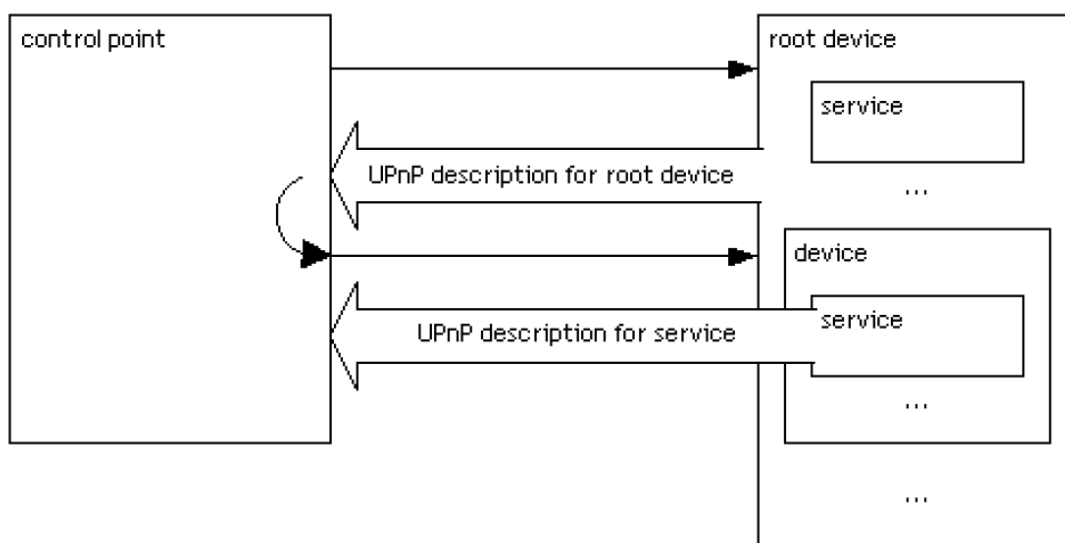
Po udanym wyszukaniu urządzenia, punkt kontrolny posiada jedynie podstawowe informacje o urządzeniu, które zostały zawarte we wcześniejszych komunikatach (Wykrywanie). Do interakcji z urządzeniem punkt kontrolny potrzebuje więcej danych, niż te zawarte w pierwszych komunikatach. W tym celu może pobrać dokładne informacje z adresu URL przesyłanego wraz z komunikatem o przyłączeniu do sieci (ang. *advertising*). Pobiera informacje nazywane są opisem usługi (ang. *description*).

Uproszczony schemat pobierania informacji o urządzeniu znajduje się na rysunku 1.5. Pojedynczy schemat pobierania opisu można podzielić na dwie operacje, uzyskania informacji o urządzeniach oraz o usługach, które one oferują. W najprostszym schemacie cały proces wykonywany jest w kilku krokach:

1. Pobranie informacji o głównym urządzeniu (ang. *root device*),
2. Pobranie informacji o usługach udostępnianych przez urządzenie (ang. *services*).

Opis urządzenia zawiera informacje o fizycznych i logicznych kontenerach (**opisać termin kontener**). Natomiast opis usług określa w sposób bardzo dokładny oferowane przez urządzenie funkcje. Pojedyncze urządzenie fizyczne może zawierać wiele urządzeń logicznych, każde z nich może być skonfigurowane jako główne (ang. *root device*). Ponadto każde urządzenie logiczne może oferować jedną lub wiele usług.

**Rys. 1.5:** Schemat pobierania opisu usługi



Opis urządzenia/usługi jest napisany przez dostawcę oprogramowania. Jest napisany w języku XML i zalecane jest, aby był zgodny z wytycznymi przedstawionymi w szablonie opisu urządzenia. [11] Najważniejszym elementem z punktu widzenia autora jest sposób opisu usługi, który powinien być stworzony tak, aby punkty kontrolne jak największej liczby producentów (stan idealny - wszystkich producentów) sprzętu i oprogramowania potrafiły uzyskać przydatne dla nich informacje.

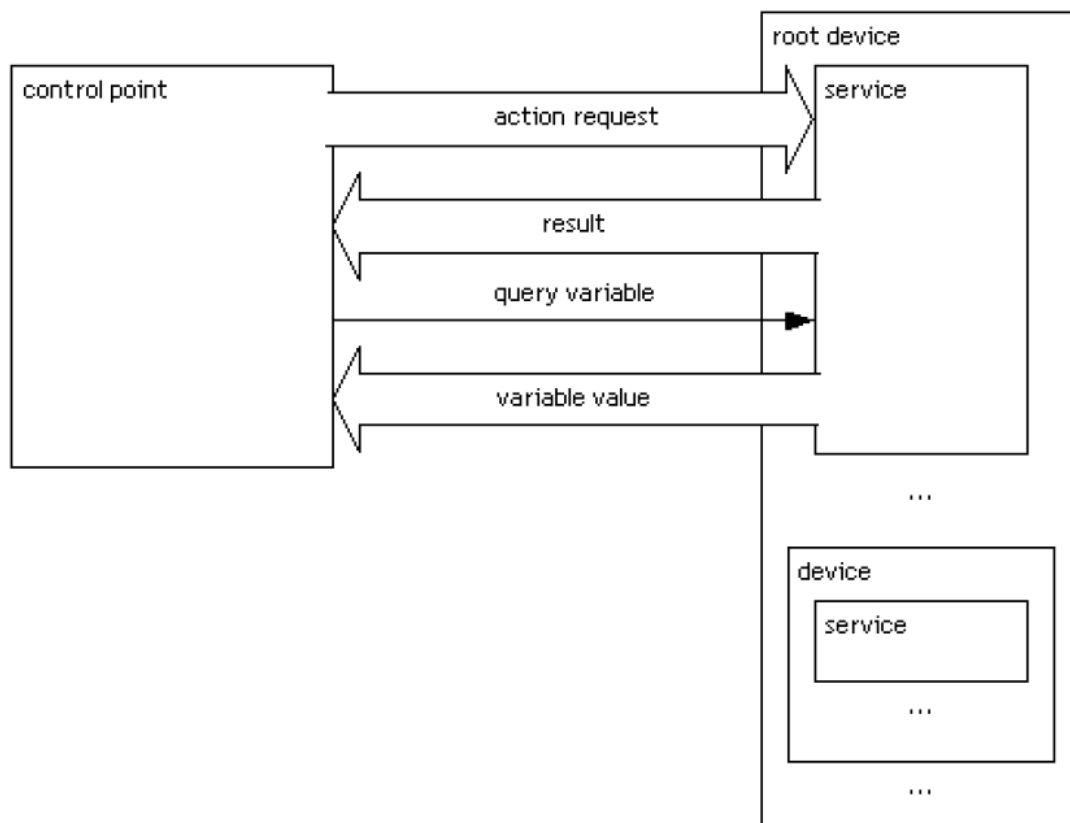
W celu otrzymania opisu urządzenia (bądź usługi) wykorzystywana jest metoda GET protokołu HTTP. Adres URL znajduje się w wiadomościach z ogłoszeniem o przyłączeniu do sieci.

### 1.2.4 Kontrola

Po uzyskaniu opisu urządzenia oraz usługi, punkt kontrolny może wywołać akcje, które mogą wykonywać dostawcy usług. Wywoływanie usługi przez punkt kontrolny można określić jako rodzaj wywołań zdalnych RPC (ang. *Remote Procedure Call*). Każde polecenie, niezależnie czy wywoływane przez punkt kontrolny czy jako odpowiedź zwrotna od usługi, jest enkapsulowane w komunikatach SOAP (ang. *Simple Object Access Protocol*) poprzez wywołania protokołu HTTP. Wywoływanie procedur przez punkt kontrolny może (lecz nie musi) być przedstawione poprzez modyfikację zmiennych określających stan usługi. O każdym takim zdarzeniu powinny zostać poinformowane wszystkie zainteresowane punkty kontrolne.[12] Punkt kontrolny może wykorzystywać wywołania zdalne, tak długo, aż wcześniej wysłane ogłoszenie nie wygaśnie lub nie zostanie odwołane (przesłanie nowego ogłoszenia, wydłuża możliwość wykonywania wywołań kontrolnych).

Przykładowe wywołanie akcji do usługi znajdującej się w urządzeniu UPnP zostało umieszczone na rysunku 1.6

Rys. 1.6: Wywołanie akcji przez punkt kontrolny



### 1.2.5 Zdarzenia

W opisie urządzenia UPnP przesyłana jest lista akcji możliwych do wykonania przez usługę oraz lista zmiennych modelujących aktualny stan usługi (w trakcie jej działania). Jako zdarzenie (ang. *event*) została zdefiniowana zmiana wartości wcześniej wspomnianych zmiennych. Każdy punkt kontrolny może zasubskrybować (z ang. *subscribe*) otrzymywanie informacji o takich zmianach. Punkt kontrolny, który zamówił subskrypcję zdarzeń nazywany subskrybentem (ang. *subscriber*), natomiast usługę, która takową subskrypcję realizuje (wysyła informację o zmianie stanu zmiennych) określamy jako wydawcę (ang. *publisher*). Wiadomości przesyłane są poprzez protokół GENA (ang. *General Event Notification Architecture*), który jest rozszerzeniem protokołu HTTP zaprojektowanym na potrzeby organizacji UPnP Forum.

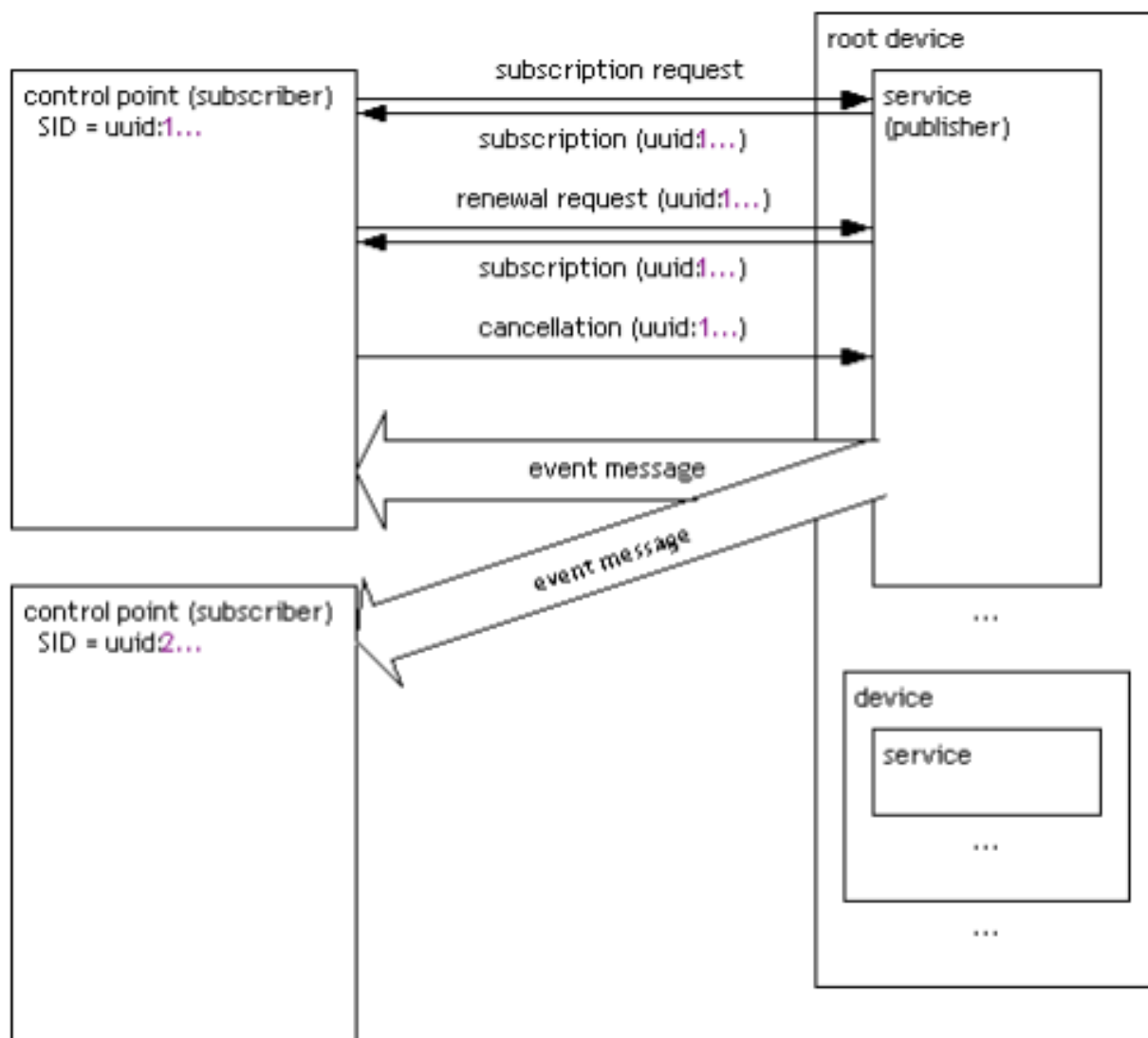
Punkt kontrolny w celu otrzymywania subskrypcji wysyła zgłoszenie subskrypcji (ang. *subscription message*). Jeżeli zgłoszenie jest zaakceptowane wydawca zwraca potwierdzenie w postaci czasu ważności subskrypcji. Subskrypcja musi być odnawiana przez subskrybenta poprzez wysłanie kolejnych zgłoszeń. Punkt kontrolny, który otrzymuje informacje o zmianie stanu zmiennych może w każdej chwili zrezygnować z ich prenumeraty poprzez wysłanie wiadomości unieważniającej (ang. *cancellation message*) do wydawcy. [12]

Do wydawcy należy zadanie wysyłania informacji o zdarzeniach. Komunikat zdarzenia (ang. *event message*) zawiera jeden lub więcej zmiennych, wraz z ich aktualnymi wartościami i jest zapisany w notacji języka XML (ang. *Extensible Markup Language*). Pierwsza wiadomość wysłana do subskrybenta jest nazywana wiadomością inicjującą (ang. *initial event message*) i zawiera wszystkie zmienne (wraz z ich wartościami). Wydawca rozsyła zawsze wszystkie zmienione wersje do każdego punktu kontrolnego, który zgłosił chęć otrzymywania powiadomień (nie ma możliwości przesyłania jedynie niektórych zmiennych). [12]

Zmienne, których wartości są zbyt duże lub których wartości ulega bardzo często zmianie, przez co ich wysłanie w komunikacie zdarzenia jest bezcelowe lub niemożliwe, wydawca może oznaczyć jako tagiem *non-evented*, czyli jako niezgłaszane w rozsyłanych pakietach. [1] Rozmiar (wielkość) zmiennych jest zgodna ze schematem dokumentu XML [15].

Przykładem zdarzeń może być zmiana poziomu głośności urządzenia UPnP, które to zdarzenie jest wysyłane do zainteresowanego punktu kontrolnego. Innym przykładem, dotyczącym multimedialnych, jest dodanie pliku, bądź plików multimedialnych i wysłanie powiadomienia o tym fakcie do subskrybentów. Dzięki temu punkty kontrolne mogą wykonać natychmiastową aktualizację listy plików w sposób wygodny i nie wymagający interwencji użytkownika.

**Rys. 1.7:** Komunikacja pomiędzy punktem kontrolnym, a usługą w przypadku zapisywania do subskrypcji



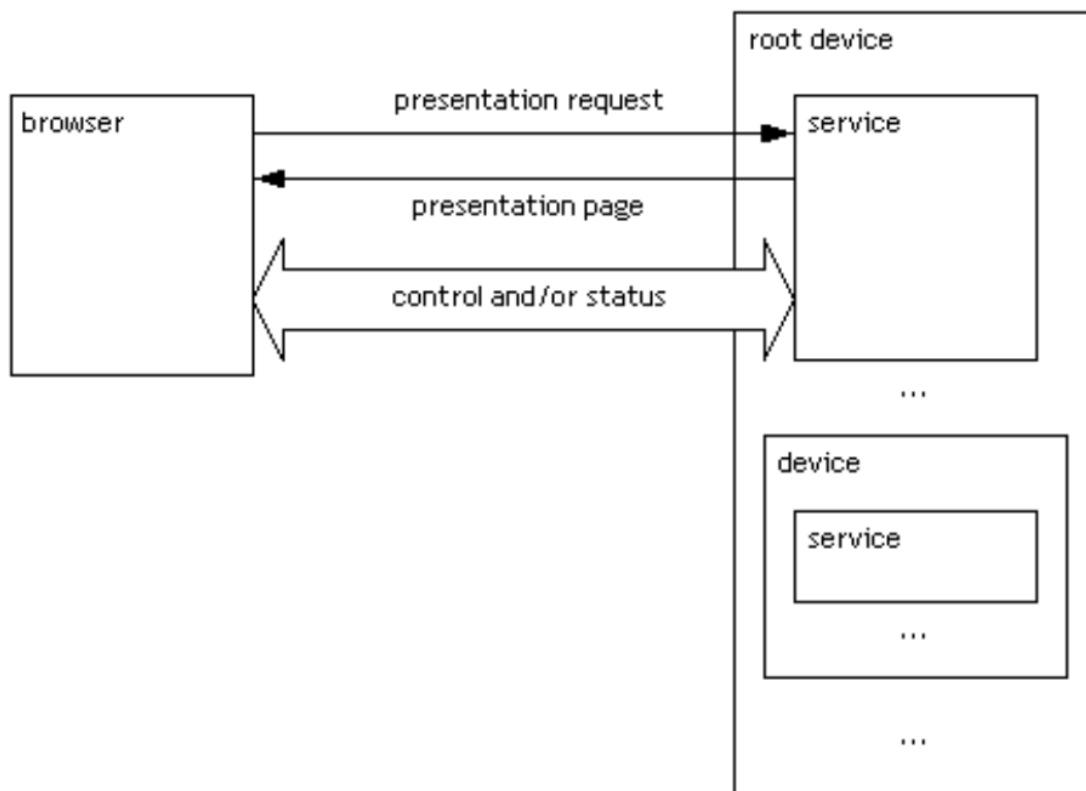
## 1.2.6 Prezentacja

Warstwa prezentacji jest ostatnią w stosie protokołu UPnP. Wyświetla ona interfejs użytkownika w postaci strony HTML, który umożliwia kontrolę i sprawdzanie statusu urządzenia oraz usług wbudowanych w to urządzenie. Dostęp do interfejsu użytkownika urządzenia umożliwia adres URL, dostarczany wraz z opisem urządzenia (warstwa opisu) 1.2.3. Dostarczanie prezentacji nie jest obowiązkowe i wiele urządzeń takiego interfejsu nie zapewnia. Uzyskanie opisywanej strony HTML urządzenia odbywa się zgodnie ze standardem HTTP (wykorzystując metodę HTTP GET).[1]

Wygląd prezentacji nie jest narzucony w żaden sposób przez zrzeczenie UPnP Forum i jest zależna w pełni od preferencji producenta urządzenia. Wymanage jest jednakże, aby strona była zgodna ze standardem HTML i powinna być w wersji 3.0 lub późniejszej.[1]

Często do kontroli urządzenia poprzez warstwę prezentacji wykorzystywane są pozostałe warstwy stosu UPnP. Jednakże nie każdy interfejs UPnP umożliwia modyfikację wszystkich danych (rozdział dotyczący warstwy zdarzeń 1.2.5), co zmusza programistów do tworzenia kilku interfejsu kontroli usług UPnP.[1]



**Rys. 1.8:** Żądanie strony prezentacji od urządzenia UPnP przez punkt kontrolny

### 1.3 Klasy urządzeń multimedialnych UPnP

Specyfikacja UPnP definiuje kilka rodzajów urządzeń, które dotyczą protokołu UPnP w sieci IP. W niniejszym rozdziale zostaną opisane rodzaje spotykanych urządzeń UPnP ze szczególnym ukierunkowaniem na rodzaje przeznaczone do współpracy z materiałami multimedialnymi. Urządzenia te są zgodne z jedną z kategorii opisywanego protokołu nazywanego UPnP AV (od ang. *Audio / Video*).

Organizacja DLNA (ang. *Digital Living Network Alliance*) zaczerpnęła multimedialne rodzaje urządzeń ze specyfikacji UPnP oraz dodała elementy wyróżniające DLNA (np drukarkę multimedialną). W niniejszym rozdziale zostaną przedstawione zarówno nazwy urządzeń DLNA, jak i multimedialnych urządzeń UPnP AV.

Jak zostało opisane w poprzednim rozdziale wyróżniamy trzy główne rodzaje urządzeń:

- punkty kontrolne,
- usługi,
- urządzenia.

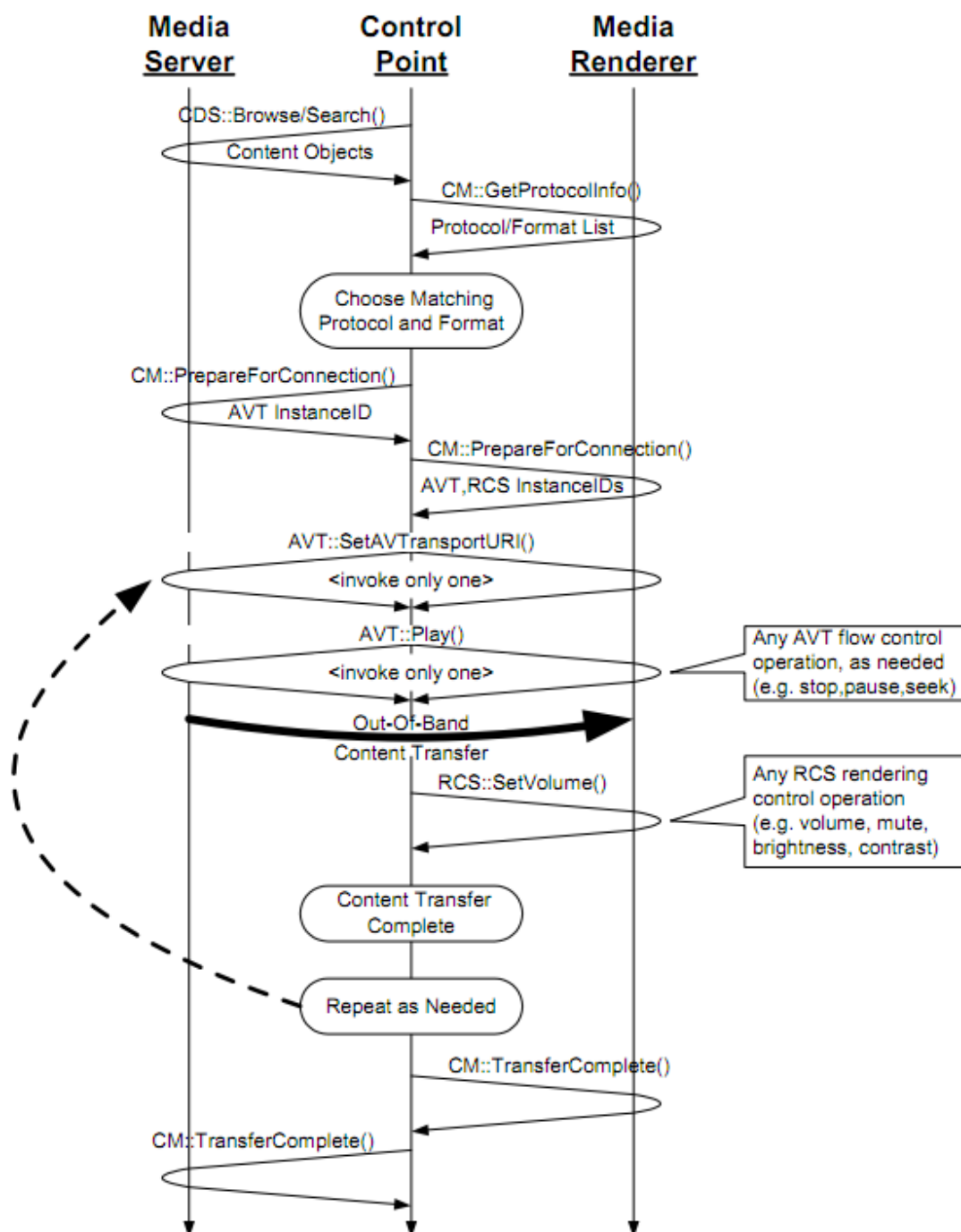
Urządzenia multimedialne podzielone zostały na następujące kategorie:

- serwer multimedialny (ang. *Media Server*) / cyfrowy serwer multimedialny DMS (ang. *Digital Media Server*),
- \* / cyfrowy odtwarzacz multimedialny DMP (ang. *Digital Media Player*),
- serwer renderujący (ang. *Media Renderer*) / cyfrowy serwer renderujący DMR (ang. *Digital Media Renderer*),

- punkt kontrolny / cyfrowy kontroler multimediiów DMC (ang. *Digital Media Controller*),
- drukarka multimediiów DMP<sub>r</sub> (ang. *Digital Media Printer*).

W powyższej liście pierwsza nazwa jest nazwą nadaną przez organizację UPnP, druga przez organizację DLNA. W przypadku, gdy odpowiednik, w którymś z protokołów nie występuje, na miejscu nazwy usługi lub urządzenia znajduje się znak myślnika.

Rys. 1.9: Schemat odtwarzania w UPnP AV

**Playback: General Interaction Diagram****1.3.1 Serwer multimedialny**

Serwer multimedii w protokołach UPnP AV oraz DLNA jest niezbędnym elementem sieci. Serwer udostępnia stworzone, pobrane, nagrane pliki multimedialne. Jest to jego główne zadanie. W protokole DLNA jego rolą jest również zapewnienie ochrony treści (praw autorskich). Serwer

multimediów UPnP AV nie zapewnia ochrony treści. Serwer multimedialny jest to zazwyczaj oprogramowanie zainstalowane na urządzeniach podłączonych do sieci Ethernet, takich jak:

- komputer typu PC,
- nagrywarka cyfrowa,
- smartfon,
- kamera cyfrowa,
- odtwarzacz BluRay / DVD.

Na urządzenie lub oprogramowanie serwera UPnP składają się następujące usługi:

- usługa zarządzania treścią (ang. *Content Directory Service*),
- usługa menedżera połączeń (ang. *ConnectionManager Service*),
- opcjonalna: usługa transportu AV (ang. *AVTransport Service*).

Pierwsza z usług, dotycząca zarządzania treścią umożliwia punktom kontrolnym przeglądanie zasobów udostępnianych przez serwer. Jedną z metod jest przeglądanie - *ContentDirectory::Browse()*, która pozwala uzyskać informacje o pliku wraz z jego metadanymi - nazwa, artysta itd. W metadanych mogą również być zawarte informacje o kodowaniu, transporcie, dzięki którym punkt kontrolny potrafi określić czy dany odtwarzacz ma możliwość odtworzenia zawartości. Druga z usług wykorzystywana jest do zarządzania ustanowionych już połączeń. Ostatnia z opisywanych usług, służy do zarządzania odtwarzanym materiałem. Umożliwia wydawanie poleceń takich jak: odtwarzaj, pauza, stop, przewiń wprzód, przewiń w tył itd.

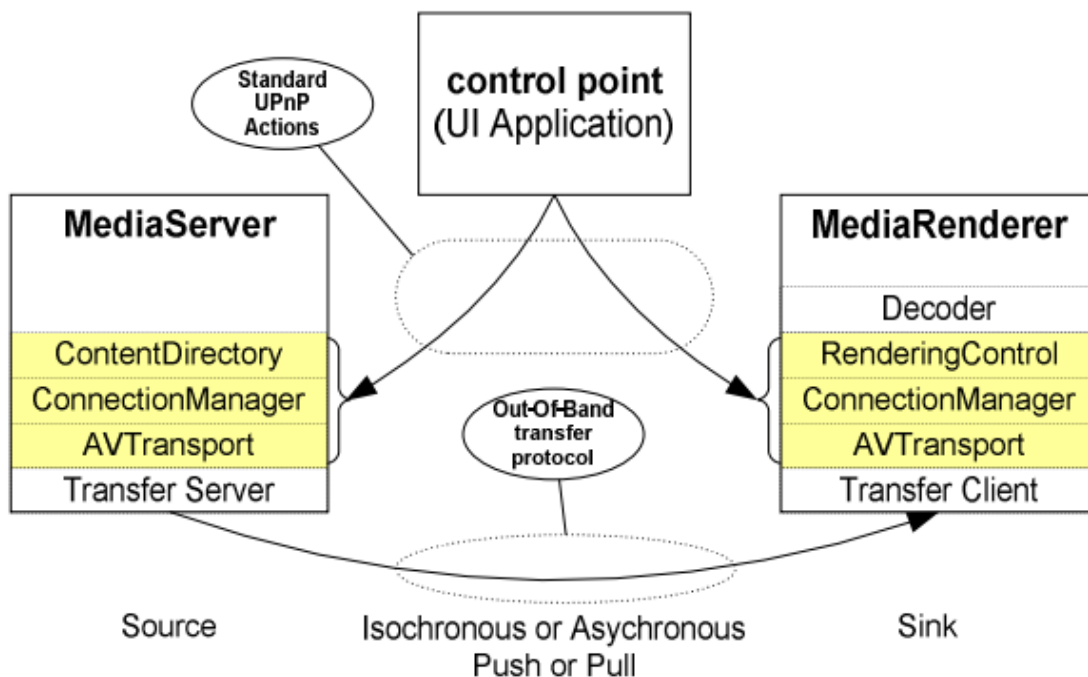
Wszystkie powyżej opisane usługi „wewnętrzne” są również integralną częścią specyfikacji DLNA (ich nazwy w większości publikacji pozostały takie same).[20]

### 1.3.2 Odtwarzacz multimediów oraz odtwarzacz renderujący

W przypadku opisywanego oprogramowania różnica, między standardami UPnP, a DLNA jest najbardziej widoczna. Zróznicowanie te polega głównie na rozdzieleniu ogólnie pojętego zadania odtwarzania materiałów multimedialnych na mniejsze, pośrednie zadania.

W przypadku UPnP AV w odtwarzaniu treści multimedialnych uczestniczy punkt kontrolny, który wyszukuje materiały udostępniane przez serwer multimediów oraz zleca ich odtworzenie. Polecenie odtworzenia może zostać wykonane w postaci dwóch metod. Metodą *push*, w której serwer multimediów wysyła treść do wybranego przez punkt kontrolny odtwarzacz renderujący, bądź też metodą *pull*, w której odtwarzacz renderujący (ang. *MediaRenderer*) pobiera dane z serwera multimediów, renderuje je i/lub odtwarza (może je renderować i wysyłać do odtwarzania do innego odtwarzacza). Uproszczony schemat opisanej powyżej procedury znajduje się na rysunku 1.10.

Rys. 1.10: Architektura odtwarzania UPnP AV



W przypadku architektury DLNA wyróżniamy dwie usługi potrafiące odtwarzać materiały:

- odtwarzacz multimedialny (ang. *Digital Media Player*),
- odtwarzacz renderujący (ang. *Digital Media Renderer*).

Pierwszy z nich służy do wyszukiwania i renderowania treści multimedialnych. Natomiast zadaniem drugiego jest renderowanie oraz odtwarzanie zadań zleconych przez kontroler multimediiów. Różnica między pierwszym, a drugim jest taka, że odtwarzaczowi renderującemu trzeba zlecić zadanie odtwarzanie, natomiast odtwarzacz multimedialny może podjąć taką decyzję „samodzielnie”.

Urządzeniami tego typu może być:

- telewizor,
- odtwarzacz DVD/BluRay,
- amplituner,
- smartfon,
- komputer PC.

Podobnie, jak w przypadku serwera UPnP, odtwarzacz posiada integralne usługi, które musi, bądź powinien mieć zaimplementowane:

- usługa kontroli renderowania,
- usługa menedżera połączeń (ang. *ConnectionManager Service*),
- opcjonalna: usługa transportu AV (ang. *AVTransport Service*).

Dwie ostatnie usługi komunikują się ze swoimi odpowiednikami w implementacji serwera multimedialnego (ich funkcje się uzupełniają). Usługa kontroli renderowania pozwala punktowi kontrolnemu sterować sposobem przetwarzania treści przez odtwarzacz renderujący. Oznacza to zmianę takich funkcji jak: jasność, kontrast, głośność, wyciszenie itd. Usługa kontroli renderowania może obsługiwać wiele odtwarzaczy renderujących, co umożliwia jednoczesną zmianę ustawień w wielu urządzeniach UPnP.[20]

### 1.3.3 Punkt kontrolny oraz kontroler multimedialny

W przypadku UPnP A/V nie wyspecyfikowano oddzielnej usługi / urządzenia do kontroli odtwarzacza. Taką rolę spełnia punkt kontrolny w obrębie całej specyfikacji UPnP. Punkt kontrolny w UPnP koordynuje działania serwera multimediiów i odtwarzacza. Zazwyczaj wysyłanie poleceń jest efektem interakcji z użytkownikiem poprzez interfejs użytkownika. Punktu kontrolnego, w opisywanej roli, specyfikacja UPnP AV nie określa jako pełnoprawnego urządzenia UPnP (nie dostarcza żadnych usług UPnP oraz nie jest widoczny jako urządzenie UPnP w sieci), lecz jest niezbędnym pośrednikiem, między użytkownikiem, a urządzeniami UPnP.

Specyfikacja DLNA wyróżnia punkt kontrolny do zarządzania odtwarzaczem multimedialnym, bądź renderującym. Kontroler multimedialny (ang. *Digital Media Controller*) odtwarza zawartość, znaną w udostępnianych zasobach serwera multimedialnego lub zarządza odtwarzaną zawartością odtwarzacza renderującego. Kontrolerami multimedialnymi zazwyczaj są:

- smartfony,
- tablety,
- itd.

Uproszczony sposób działania algorytmu punktu kontrolnego, współpracującego z pojedynczym serwerem multimedialnych oraz serwerem renderującym wygląda następująco:

1. Wykrywanie urządzeń Audio / Video (przy pomocy mechanizmów UPnP).
2. Zlokalizowanie pożądanej treści, przy pomocy metod serwera UPnP, takich jak *ContentDirectory::Browse()* oraz *ContentDirectory::Search()*, które umożliwiają odnalezienie zasobu.
3. Pobierz wspierane, przez odtwarzacz renderujący, formaty danych i protokoły (poprzez metodę *ConnectionManager::GetProtocolInfo()*).
4. Sprawdź możliwości odtworzenia materiału przez pożądany odtwarzacz, wyznacz najlepszy sposób transmisji oraz format danych.
5. Skonfiguruj serwer oraz odtwarzacz renderujący - wykorzystywane są takie polecenia jak: *ConnectionManager::PrepareForConnection()*, *AVTransport::Play()*, *AVTransport::Stop()*, *AVTransport::Pause()*, *AVTransport::Seek()* itd.
6. Wyznacz zawartość do przesłania.
7. Rozpocznij przesyłanie wybranej treści.
8. Możliwe wywoływanie działań na odtwarzaczu (np. pauza).
9. Wybierz opcję: powtórz odtworzenie lub kolejny materiał.
10. Po zakończeniu działania zamknij połączenia z serwerem multimediiów.

### 1.3.4 Odmiany mobilne

Specyfikacja DLNA dla większości typów urządzeń (wyjątkiem jest drukarka *DMP*) przewiduje możliwość ich odmiany mobilnej. Spełniają one te same funkcje co typy główne. Rozróżnienie dużej ilości typów zostało wprowadzone dla ułatwionej orientacji konsumentów, pośród różnych produktów.[21]

Poza powtarzającymi się typami wyróżniamy dwa, które istnieją jedynie w wersji mobilnej:

- menedżer wysyłania multimediiów (ang. *Mobile Digital Media Uploader M-DMU*),
- menedżer pobierania multimediiów (ang. *Mobile Digital Media Downloader M-DMD*).[21]

Pierwszy typ służy do wysyłania (całości) materiału do serwera multimediiów (mobilnej wersji bądź standardowej) poprzez sieć bezprzewodową. Przykładem takich urządzeń mogą być: cyfrowy

aparat fotograficzny, kamera video, smartfon. Menedżer pobierania multimediiów, również jest urządzeniem działającym w obrębie sieci bezprzewodowej, które pobiera materiały udostępniane przez serwer DLNA (niezależnie od wersji mobilnej, bądź standardowej). Menedżerem pobierania może być smartfon lub przenośny odtwarzacz muzyki.[21]

Odmiany mobilne urządzeń i usług nie są wyróżnione w dokumentacji UPnP.

# Multimedia w UPnP AV/DLNA

Specyfikacja UPnP AV została stworzona do umożliwienia oraz ustandaryzowania sposobu komunikacji ze sobą urządzeń multimedialnych w obrębie sieci lokalnej oraz zapewnienie współpracy między produktami różnych dostawców. Cel ten w specyfikacji UPnP AV został w dużej mierze osiągnięty. Specyfikacja UPnP AV pozostawia jednak swobodę w możliwościach obsługiwanych formatów danych. Oznacza to, że pomimo komunikowania się ze sobą różnych urządzeń, cel dla użytkownika końcowego, czyli odtworzenie treści, nie zawsze może być spełniony. Liberalność w kwestii wspieranych formatów zadziałał w tym przypadku na niekorzyść specyfikacji UPnP AV.

Z powyżej opisanego powodu producenci urządzeń powołali konsorcjum DLNA. W specyfikacji DLNA autorzy skupili się w również na formatach odtwarzanych plików. Natomiast ze specyfikacji UPnP zapożyczyli między innymi w prawidłowy sposób działającą komunikację między urządzeniami. Jako format definiuje się możliwy sposób zakodowania pliku.

Certyfikacja DLNA jest stosowana przez większość producentów urządzeń multimedialnych, a spośród w chwili obecnej najprawdopodobniej najpopularniejszego przeznaczenia DLNA (odtwarzanie filmów), większość producentów nowoczesnych telewizorów wyposaża je w oprogramowanie odtwarzacza DLNA.

Zgodnie ze standardem UPnP AV oraz DLNA plik multimedialny można zaklasyfikować do jednego z następujących formatów:

- audio (ang. *audio*),
- obraz (ang. *image*),
- audio-wideo (ang. *AV* / *audio-video*).

W niniejszym rozdziale zostaną opisane obsługiwane typy multimediiów, ich klasyfikacje w opisanych specyfikacjach oraz formaty kodowania plików, przechowywania oraz sposób ich obsługi.

Typy plików (lub ich kodowania) konieczne do obsługi oraz opcjonalne, zdefiniowane przez DLNA, znajdują się w tabelach 2.1, . . . . Objaśnienia do niniejszej tabeli znajdują się w rozdziałach poświęconym poszczególnym klasom multimediiów.

**Tabela 2.1:** Typy obsługiwanych plików w DLNA - standardowe urządzenia[23]

| Klasa multimediiów         | Wymagane typy plików | Opcjonalne typy plików          |
|----------------------------|----------------------|---------------------------------|
| Obraz (ang. <i>Image</i> ) | JPEG                 | PNG, GIF, TIFF                  |
| Audio                      | LPCM                 | AC3, AAC, MP3, WMA9, ATRAC3plus |
| Wideo (ang. <i>Video</i> ) | MPEG2                | MPEG1, MPEG4, WMV9              |

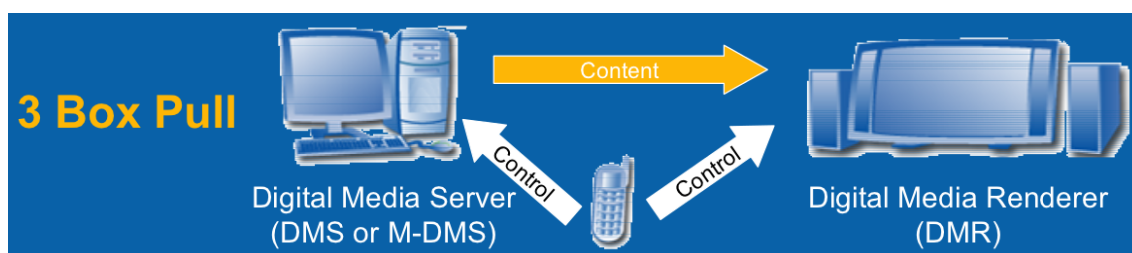


**Tabela 2.2:** Typy obsługiwanych plików w DLNA - urządzenia mobilne[23]

| Klasa multimediów          | Wymagane typy plików | Opcjonalne typy plików  |
|----------------------------|----------------------|---|
| Obraz (ang. <i>Image</i> ) | JPEG                 | PNG, GIF, TIFF  |
| Audio                      | MP3, MPEG4 AAC LC    | MPEG4 (HE AAC, AAC LTP, BSAC),<br>AMR, ATRAC3plus, G.726, WMA, LPCM |
| Wideo (ang. <i>Video</i> ) | MPEG4 AVC (AAC LC)   | VC1, H.263, MPEG4,<br>MPEG2, MPEG4 AVC (np. BSAC)                   |

## 2.1 Odtwarzanie multimediów

W odtwarzaniu multimediów w DLNA (analogicznie w UPnP AV) muszą brać udział minimum dwa urządzenia. Jedno to serwer multimediów udostępniający materiał, drugie to odtwarzacz DLNA (rysunek 2.1). Istnieje również schemat w którym udział bierze więcej urządzeń, a trzecim urządzeniem jest kontroler multimedialny pełniący rolę „pilota” zarządzającego, co ma być odtwarzane oraz zarządzać odtwarzaniem (2.2).

**Rys. 2.1:** Uproszczony schemat komunikacji 2 urządzeń**Rys. 2.2:** Uproszczony schemat komunikacji 3 urządzeń

Pierwsze dwa kroki w celu odtwarzania materiału to podłączenie urządzenia do sieci:

1. Urządzenie dołącza do sieci lokalnej (pobiera adres z serwera DHCP lub autorytatywnie przydziela sobie adres IP - metoda Auto-IP),
2. Urządzenie wysyła informacje o sobie z wykorzystaniem protokołu SSDP na adres 239.255.255.255:1900.

W dalszej kolejności następuje prawidłowa procedura odtwarzania:

1. Punkt kontrolny (w notacji DLNA - odtwarzacz multimedialny lub kontroler multimedialny) wysyła żądanie opisu urządzenia i usługi (usług) do serwera multimedialnego.
2. Serwer multimedialny odsyła opis urządzenia i usługi (usług) w postaci pliku XML.

3. Punkt kontrolny za pomocą komunikat SOAP wysyła do serwera akcje, którą chce wykonać. W przypadku odtwarzania mogą być to komunikaty przeglądania multimedialnych (ang. *browse*), ich wyszukiwania (ang. *search*). Ilość komunikatów jest zależna od drzewa wyszukiwania pożądanego pliku przez użytkownika.
4. Wyszukiwanie kończy się wyborem pliku z wykorzystaniem metod protokołu HTTP np. HTTP GET.

Odtwarzacz multimedialny ma możliwość, działania w typowy sposób dla tradycyjnego odtwarzacza muzyki lub video/DVD. Może przewijać materiał w przód, w tył. Przewijanie jest wykonywane przez podawanie pożądanego bajtów pliku (np. bajt 50 000 - 150 000) przez odtwarzacz multimedialny. Wykonywanie przewijania i skoków w filmie jest możliwe tylko wtedy gdy serwer multimedialny udostępnia rozmiar pliku lub długość pliku (niektóre odtwarzacze pożądadają jednostek w minutach dla długości muzyki czy filmu, pomimo, że prawidłowo wykorzystują pobieranie tzn. pobierają określoną ilość bajtów).

Odtwarzacz multimedialny zgodnie z przytaczanymi specyfikacjami mają możliwość układania list odtwarzania (ang. *playlist*), jednakże autor nie odnalazł odtwarzacza, który umożliwiałby jej utworzenie.

## 2.2 Klasy multimedialnych UPnP

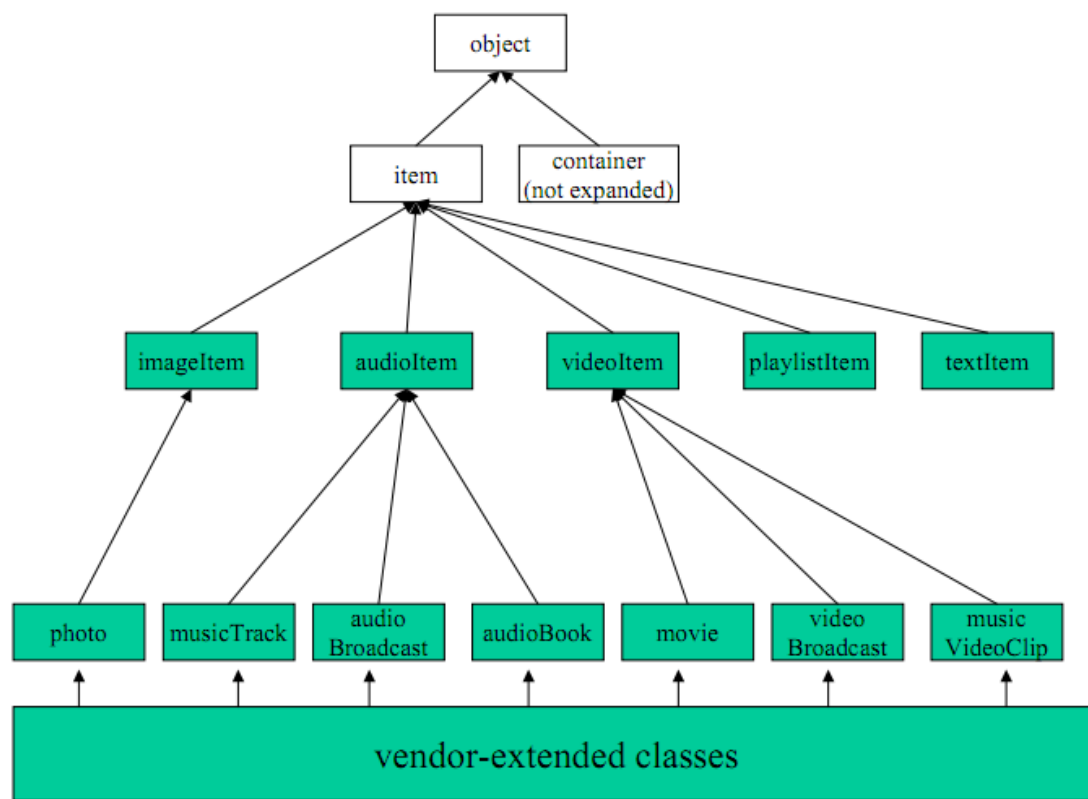
Klasą główną urządzeń UPnP jest *item*, w którym są zawarte następujące podtypy:

- rysunek (*imageItem*),
  - obraz (*photo*),
- audio (*audioItem*)
  - utwór muzyczny (*musicTrack*),
  - nadawanie materiału audio (*audioBroadcast*),
  - audiobook (*audiobook*),
- wideo (*videoItem*),
  - film (*movie*),
  - nadawanie materiału wideo (*videoBroadcast*),
  - teledysk (*musicVideoClip*),
- lista odtwarzania (*playlistItem*),
- tekst (*textItem*).[22]

Każdy z powyższych typów, może posiadać podtypy określone przez wystawcę oprogramowania. Schemat typów został dodatkowo przedstawiony na rysunku nr 2.3.

Rys. 2.3: Schemat typu *item* wraz ze standardowymi podtypami

upnp\_av\_contentdirectory\_service:2



Poza typowymi *item* wyróżniamy również tak zwane kontenery (ang. *containers*). Reprezentują one kolekcje obiektów (fizyczną, bądź logiczną np. posortowaną według typów filmów). Dokładny opis obu typów, wraz ze sposobem stosowania, znajduje się w rozdziale implementacyjnym niniejszej pracy magisterskiej.

W przypadku DLNA zrezygnowano w materiałach z przyrostka *item* i pozostały nazwy takie jak: *audio*, *AV*, *image*. Jednakże w implementacji DLNA (w postaci XML) pozostawiono nazwy, takie jak w UPnP AV, dlatego w niniejszej pracy autor używa nazw pochodzących z UPnP AV. Wyjątek stanowi sytuacja, gdy nazwa została zdefiniowana jedynie w specyfikacji DLNA, a w specyfikacji autorstwa UPnP Forum jej odpowiednik nie występuje.

Typ pliku jest określany poprzez opcję *res* (ang. *resource*).

### 2.2.1 Klasa *object* oraz klasa *item*

Wyróżniamy dwie klasy główne: *object* i *item*. Pierwsza z nich to klasa główna, w której znajduje się cała udostępniana odtwarzaczowi UPnP lub DLNA treść multimedialna (ang. *ContentDirectory*). Klasa *object* określa właściwości, które są wspólne dla każdego z udostępnianych plików, jak i dla ustrukturyzowanej logicznej kolekcji materiałów. Niniejsza lista przedstawia wymagane właściwości dostarczane wraz z obiektem:

- @id - unikalny identyfikator klasy głównej,
- @parentID - identyfikator klasy nadrzędnej,
- @restricted - ...???
- dc:title - nazwa obiektu,

- upnp:class - klasa zawartości zgodna z UPnP[22].

Dodatkowymi właściwościami opisywanej klasy są:

- dc:creator - ...?
- res - URL ????
- upnp:writeStatus - ????????????[22]

Klasa *item* jest typem określającym każdy indywidualny materiał multimedialny np. zdjęcie w albumie. Obiekt ten nie może zawierać w sobie innych obiektów (nie może być „katalogiem”). Klasa dziedziczy wszystkie opcje po klasie głównej *object* oraz wprowadza następujące właściwości (wszystkie są opcjonalne):

- @refID - odniesienie do pliku głównego (w przypadku gdy plik znajduje się w logicznym kontenerze). Jeżeli udostępniany zasób jest odniesieniem, wówczas pole opisywana opcja jest obowiązkowa,
- upnp:bookmarkID - .....?

### 2.2.2 Klasa kontener

Klasa kontener (ang. *container*) dziedziczy po typie głównym *object* i reprezentuje kolekcję innych obiektów (innych kontenerów lub innych podtypów dziedziczącej po klasie *object*). Nie określa ona żadnych obowiązkowych właściwości, które trzeba dostarczyć drugiej stronie komunikacji. Opcjami dodatkowymi są natomiast pola:

- @childCount - liczba obiektów, które znajdują się w kontenerze,
- upnp:CreateClass - .....?
- upnp:SearchClass - .....?
- @searchable - wartość 0 lub 1???????
- @neverPlayable - ????????????

### 2.2.3 Klasa obraz

Typ obraz (ang. *imageItem*) wywodzi się z klasy głównej *item* i dziedziczy od niej wymagane opcje. Ponadto typ obraz powinien zawierać następujące opcje (minimum jedno wystąpienie):

- res - ścieżka do obrazu
  - res@resolution - rozdzielczość obrazu,
- upnp:longDescription - długi opis obrazu,
- upnp:storageMedium - rodzaj pamięci,
- upnp:rating - ocena,
- dc:description - opis obrazu,
- dc:publisher - wydawca,
- dc:date - data powstania obrazu,
- dc:rights - prawa do obrazu (???) [22]

Gdy przedstawianym obrazem jest zdjęcie może ono być podtypu zdjęcie *photo:imageItem*, które dziedziczy wszystkie właściwości typ obraz oraz umożliwia układanie zdjęć w albumy, dzięki opcjonalnej właściwości:

- *upnp:album* - nazwa albumu.

Obraz lub zdjęcie może być udostępnione w kilku rozmiarach oraz według kilku typów. Zdjęcie może być udostępnione w następujących rozmiarach:

- SM (ang. *small*) - małe,
- MED (ang. *medium*) - średnie,
- LRG (ang. *large*) - wielkie,
- TN (ang. *thumbnail*) - miniatura.

Obsługiwany obowiązkowo typ grafik to JPEG\_SM. Opcjonalnymi typami, które może obsługiwać odtwarzacz są JPEG\_MED, JPEG\_LRG, JPEG\_TN, PNG\_TN. Ponadto miniaturami obrazów, urządzenie może udostępniać ikony (np. ikona z logiem producenta oprogramowania). Ikony mogą być formatowane zgodnie z następującymi rozmiarami: PNG\_SM\_ICO, PNG\_LRG\_ICO, JPEG\_SM\_ICO, JPEG\_LRG\_ICO. W przypadku ikon, jeżeli urządzenie wspiera ich udostępnianie, wówczas musi udostępniać ich wszystkie rozmiary.

Na podstawie dokumentacji DLNA autor opracował tabelę 2.3 zestawiającą formaty plików oraz ich właściwości. Tabela zawiera standardowe formaty danych i nie uwzględnia formatów opcjonalnych (takich jak PNG czy BMP), które mogą być obsługiwane przez urządzenie. Jednakże bez względu na typ pliku, właściwości dotyczące rozmiaru (na podstawie którego klasyfikowana jest nazwa skrócona obrazu np. MED) muszą być uwzględnione, a obraz udostępniony z odpowiednią nazwą profilu (np. PNG\_MED).

Obrazy udostępniane są również w postaci miniatur. Miniatury mogą odnosić się nie tylko do obrazów i zdjęć, lecz do pozostałej udostępnianej zawartości, czyli muzyki i filmów. Każdy album muzyczny może mieć miniaturę z okładką albumu, zaś film miniaturę z plakatem, wybraną sceną lub inny dowolny obraz.

**Tabela 2.3:** Tabela porównawcza standardowych formatów obrazów w technologii DLNA

| Typ MIME   | Profil*      | Właściwości   |
|------------|--------------|---|
| image/jpeg | JPEG_SM      | maksymalny rozmiar**: 640 x 480                             |
|            |              | paleta kolorów: niekalibrowane lub sRGB                     |
|            | JPEG_MED     | maksymalny rozmiar: 1024 x 768                              |
|            | JPEG_LRG     | maksymalny rozmiar: 4096 x 4096                             |
|            | JPEG_TN      | maksymalny rozmiar: 160 x 160                               |
|            | JPEG_SM_ICO  | maksymalny rozmiar: 48 x 48                                 |
| image/png  | JPEG_LRG_ICO | maksymalny rozmiar: 120 x 120                               |
|            | PNG_TN       | maksymalny rozmiar: 160 x 160                               |
|            | PNG_SM_ICO   | maksymalny rozmiar: 48 x 48                                 |
|            |              | paleta kolorów: odcienie szarości, 24 bitowa paleta kolorów |
|            | PNG_LRG_ICO  | maksymalny rozmiar: 120 x 120                               |

\*na profil składa się typ pliku (np. JPEG) oraz rozmiar: SM - mały, MED - średni, LRG - wielki, TN - miniatura, SM\_ICO - mała ikona, LRG\_ICO - duża ikona

\*\* rozmiar podawany w pikselach

## 2.2.4 Klasa audio

Typ audio służy do udostępniania plików audio, które są przeznaczone jedynie do słuchania. Oznacza to, że klasa ta nie dotyczy ścieżki dźwiękowej w materiałach wideo, takich jak filmy czy transmisja telewizyjna. Klasą nadrzedną dla audio, jest klasa *item*, od której dziedziczone są wszystkie jej właściwości. Podobnie jak w klasie opisującej obraz, powinna ona zawierać opcję

*res* wraz z danymi dotyczącymi zasobu. Ponadto opcjonalnymi właściwościami możliwymi do załączenia są:

- *upnp:genre* - gatunek pliku audio,
- *dc:description* - opis pliku,
- *upnp:longDescription* - rozszerzony opis pliku,
- *dc:publisher* - wydawca,
- *dc:language* - język,
- *dc:relation* - .....?,
- *dc:rights* - .....?

Klasa audio posiada następujące podtypy, określające rodzaj dostępnego materiału dźwiękowego:

- muzyka (ang. *musicTrack*),
- transmisja audio (ang. *audioBroadcast*),
- książka audio (ang. *audioBook*).

Pierwszy z podtypów jest jednocześnie najbardziej popularnym do udostępniania w klasie audio. Muzykę można klasyfikować w albumy, listy odtwarzania, udostępniać informacje o artyście. Wszystkie te atrybuty są opcjonalne, a ich pełna lista wygląda następująco:

- *upnp:artist* - wykonawca,
- *upnp:album* - album muzyczny,
- *upnp:originalTrackNumber* - oryginalny numer utworu,
- *upnp:playlist* - lista odtwarzania,
- *upnp:storageMedium* - rodzaj pamięci??,
- *dc:contributor* - ??????
- *dc:date* - data???????

Transmisja audio daje bardzo dużo możliwości rozszerzenia funkcji odtwarzaczy o wzbogacenie ich o takie opcje jak radio internetowe, dzięki przekazywaniu transmisji na żywo przez serwer. Opisany podtyp można przesyłać wraz z opcją *res* zawierającą dane o zasobie oraz opcjonalnymi atrybutami, takimi jak:

- *upnp:region*
- *upnp:radioCallSign*
- *upnp:radioStationID*
- *upnp:radioBand*
- *upnp:channelNr*
- *upnp:signalStrength*
- *upnp:signalLocked*
- *upnp:tuned*
- *upnp:recordable* .????????????????????????????

Ostatni z typów dziedziczący po klasie audio, czyli audiobook, służy do określania charakterystyki tego typu książek. Pole z zasobem (*res*) powinno być zawarte w przesłanych danych co najmniej raz. Poza nim mogą zostać dołączone następujące atrybuty:

- *upnp:storageMedium*,

- upnp:producer,
- dc:contributor,
- dc:date .????????

### 2.2.5 Klasa wideo

Klasa wideo opisuje materiały audiowizualne udostępniane w sieci. Są to materiały, których udostępnianie jest najtrudniejsze, zarówno z perspektywy programisty, jak i użytkownika. Jest to spowodowane bardzo dużą rozległością w sposobach kodowania materiałów oraz wciąż niepełną kompatybilnością wielu urządzeń z wieloma formatami plików. W plikach wideo warto zwrócić uwagę na dwa podstawowe pojęcia: kontener multimedialny oraz format pliku wideo. Pojęcie te występuje również w plikach graficznych i audio, jednakże stopień komplikacji dla sprzętu komputerowego występuje dopiero w plikach zawartych w opisywanej klasie.

W ogólnie rozumianym kontenerze multimedialnym może być przechowywanych wiele strumieni obrazów, audio, wideo, napisów, informacji o strumieniach i innych metadanych. Pozwala on na przechowywanie wszystkich tych danych w jednym pliku[25]. Natomiast jako format wideo definiowany jest sposób kodowania strumienia wideo. Analogiczne definicje dotyczą formatów audio oraz obrazów.

Podstawowy typ wideo (ang. *videoItem*) dziedziczy po klasie głównej *item*. Zawiera minimum jedną informację o zasobie *res*. Ponadto z materiałem mogą być przekazane następujące atrybuty:

- upnp:genre,
- upnp:genre@id,
- upnp:genre@type,
- upnp:longDescription,
- upnp:producer,
- upnp:rating,
- upnp:actor,
- upnp:director,
- dc:description,
- dc:publisher,
- dc:language,
- dc:relation,
- upnp:playbackCount,
- upnp:lastPlaybackTime,
- upnp:lastPlaybackPosition,
- upnp:recordedDayOfWeek,
- upnp:srsRecordScheduleID.????????

Jak jest widoczne na powyższej liście ilość atrybutów możliwych do przesłania jest bardzo duża. Jest to bazowa klasa dla pozostałych typów wideo, którymi są:

- film (*movie:videoItem*),
- transmisja wideo (*videoBroadcast:videoItem*),
- teledysk / klip muzyczny (*musicVideoClip:videoItem*).

Klasa film reprezentuje materiały filmowe i umożliwia załączenie dodatkowych danych odnośnie filmu w wyspecyfikowanych, przez autorów dokumentacji UPnP AV, atrybutach. Na atrybuty składają się:

- *upnp:storageMedium* - nośnik pamięci,
- *upnp:DVDRegionCode* - kod regionalny materiału DVD,
- *upnp:channelName* - nazwa kanału,
- *upnp:scheduleStartTime* - planowany start transmisji,
- *upnp:scheduleEndTime* - planowany koniec transmisji,
- *upnp:programTitle* - tytuł programu (odcinka),
- *upnp:seriesTitle* - tytuł serii,
- *upnp:episodeCount* - liczba odcinków,
- *upnp:episodeNr* - numer odcinka.

Wszystkie z powyższych opcji są możliwe do udostępnienia w postaci atrybutów XML, lecz nie są obowiązkowe.

## 2.3 Rodzaje multimedialnych

Filmy, a dokładnie ich pełne wsparcie i obsługa, były najciekawszym i zarazem najtrudniejszym tematem do realizacji dla autora niniejszej pracy. Z tego powodu niniejszy podrozdział zostanie głównie poświęcony opisowi, w jaki sposób w erze komputerów przechowywane są tego rodzaju materiały.

Ilość formatów danych multimedialnych jest bardzo duża. Najbardziej pożądanymi formatami są formaty kompresujące dane. Formaty plików zapewniające tzw kompresję bezstratną (formaty *RAW* - surowe) są formatami pozwalającymi zachować najwyższą jakość obrazu lub dźwięku. Przykładami takich formatów są dla plików graficznych - *RAW*, natomiast audio - format *FLAC* (ang. *Free Lossless Codec*). Innym popularnym formatem dźwięku, który jest uważany za bardzo bliski oryginałowi, dzięki pokrywaniu całego zakresu pasma częstotliwości słyszalnych jest kodowanie *PCM*, stosowane bardzo często w formacie *WAV* (<http://pl.wikipedia.org/wiki/PCM>). Spośród plików wideo można wyróżnić formaty danych o niskiej kompresji stratnej. Przykładem takich plików są formaty *MPEG-2* oraz *MPEG-4*. Drugi z nich pozwala na przesyłanie materiału HD i posiadają kompresję w stosunku 4:1 (skąd to wiem). Wykorzystywany jest współcześnie w przesyłaniu cyfrowej telewizji naziemnej oraz częściowo satelitarnej.

W przypadku zdjęć prym wiodzie format *JPEG*, który jest ustandaryzowany przez organizację *ISO/IEC* [26], jest formatem kompresji stratnej. Jednakże kompromis między kompresją, a jakością obrazu (kompresja 10:1) jest na tyle zadowalający, że stał się najpopularniejszym formatem danych graficznych (z pominięciem zastosowań profesjonalnych, gdzie często wykorzystywane są formaty bezstratne, bądź inne specjalistyczne, specyficzne dla oprogramowania obróbki grafiki).

Najpopularniejszym formatem audio dla przechowywania muzyki wśród końcowych użytkowników jest format *MP3* (ang. *MPEG-1/MPEG-2 Audio Layer 3*). Jest standardem *RFC*.... W swojej pierwotnej, a zarazem najpopularniejszej wersji pozwala na zapis muzyki w postaci dźwięku jednokanałowego (mono) lub dwukanałowego (stereo) i oferuje najwyższy stopień kompresji, przy zachowaniu dobrej jakości dźwięku. W przypadku ścieżek dźwiękowych do filmów częściej zastosowanie ma standard *AC3* [wikipedia]

W przypadku materiałów filmowych trudnym zadaniem jest jednoznaczne określenie najpopularniejszego formatu tego typu danych. W kolejnych podrozdziałach został dokładnie opisany ten rodzaj multimedialnych.



### 2.3.1 Muzyka

Obrazy

### 2.3.2 Filmy

Film można rozumieć zgodnie z następującym schematem:

- Kontener filmowy (np. AVI, MKV)
  - Format wideo (np. MPEG-2),
  - Format audio (np. AC3),
  - Napisy,
  - Rozdziały,
  - Metadane opisujące powyższe dane.

Istnieje wiele rodzajów kontenerów. Część z nich ma przeznaczenie do udostępniania multimedialnych w sieci internetowej (Flash), inne są dedykowane dla urządzeń mobilnych (MP4). Najpopularniejszymi kontenerami multimedialnymi są:

- AVI (rozszerzenie AVI),
- Matroska (rozszerzenie MKV),
- Flash (rozszerzenie FLV lub F4V).
- MP4,
- DivX.

Kontenery różnią się metadanymi opisującymi pliki oraz możliwościami we wspieraniu przechowywanych formatów danych. Tabela porównawcza formatów AVI i MKV pokazuje podstawowe różnice między tymi formatami (tabela 2.4).

Materiał wideo w filmach może być kodowany w formatach, takich jak:

- MPEG-1,
- MPEG-2
- MPEG-4 AVC / H.264,
- WMV (ang. *Windows Media Video*),
- RealVideo.

Kontenery filmowe pozwalają przechowywać różne formaty wideo (część z nich widoczna jest powyżej), różne formaty audio (jeden plik może zawierać kilka ścieżek audio w różnych formatach np. MP3 lub AC3). Niektóre umożliwiają przechowywanie w kontenerze plików z napisami (we wspieranych formatach), posiadają możliwości podziału materiału na sekcję lub przechowywania materiału trójwymiarowego 3D. Wszystkie dane co przechowywane jest w danym pliku zawierają się metadanych pliku. Każdy rodzaj konteneru posiada (niestety) zróznicowany sposób tworzenia metadanych. Utrudnia to w znaczący sposób łatwe wsparcie jak ich największej ilości.

Oddzielnym tematem są formaty płyt Blu-Ray, bądź DVD. Formaty te posiadają specjalną strukturę plików oraz schemat katalogów, według których umieszczone są odpowiednie dane. W uproszczeniu można powiedzieć, że nośnik Blu-Ray jest kontenerem multimedialnym, takim jak AVI czy MKV. Podobna sytuacja ma miejsce w przypadku płyt DVD.

Płyta Blu-Ray, zgodna ze standardem posiada drzewo, składające się z dwóch plików „index.bdmv” (zawierający informacje o zawartości katalogów płyty BD), MovieObject.bdmv (zawierający informacje o obiektach filmowych) oraz katalogów: PLAYLIST (lista odtwarzania filmu / rozdziały), CLIPINF (informacje o klipach wideo), STREAM (strumień audio wideo w kontenerze

m2ts - ang. MPEG-2 Transport Stream), AUXDATA (pliki dźwiękowe oraz czcionki dla plików z napisami), BACKUP (katalogów zawierający kopie zapasowe plików index.bdmv oraz MovieObject.bdmv).

**Tabela 2.4:** Porównanie kontenerów AVI i MKV

| Format | Rozdziały | Napisy | Formaty wideo                  | Formaty audio                                  |
|--------|-----------|--------|--------------------------------|--|
| AVI    | TAK       | NIE    | MPEG-1, MPEG-2, WMV, RealVideo | MP3, WMA, AAC, AC-3, DTS, PCM                  |
| MKV    | TAK       | TAK    | MPEG-1, MPEG-2,                | MP3, WMA, AAC, AC-3,                           |
|        |           |        | MPEG-4 AVC, WMV, RealVideo     | DTS, PCM, RealAudio, Vorbis, APE, Flac, DTS-HD |
|        |           |        |                                |  |

Film może posiadać wiele ścieżek dźwiękowych np. różne wersje językowe lub lektora. Płyty DVD oraz Blu-Ray w standardzie swoim mają możliwość zamieszczania wielu ścieżek dźwiękowych. Podobna sytuacja ma miejsce w przypadku kontenerów danych takich jak AVI oraz Matroska. Oprogramowanie udostępniające materiały multimedialne powinno zapewniać możliwość wyboru ścieżki dźwiękowej. W tym celu konieczne jest poprawne odczytywanie metadanych dostarczanych wraz z kontenerem multimedialnym. Filmy DVD oraz Blu-Ray ponadto, że umożliwiają zmianę ścieżki dźwiękowej, wprowadziły do zestawów kina domowego możliwość wyświetlania napisów w filmach (przedtem napisy były cechą wyróżniającą kina). W przypadku płyt DVD napisy są przechowywane w formacie bitmapowym, co wykracza poza standardy UPnP oraz DLNA (dla napisów), dlatego też nie są one obsługiwane do udostępniania. Płyty Bluray posiadają standard zbliżony do płyt DVD (napisy PGS - ang. *Presentation Graphic Stream*). Łatwiejszym przypadkiem są kontenery plików. Napisy w przypadku tego typu filmów mają postać pliku tekstowego. Plik z napisami może być wbudowany w kontener, a może być udostępniany jako oddzielny plik (zazwyczaj w tym samym folderze i tej samej nazwie co film).

Istnieje wiele standardów napisów do filmów, które mogą być w prosty sposób przechowywane na dysku twardym komputera stacjonarnego. Głównymi standardami spotykanymi powszechnie w Internecie są MicroDVD oraz SubRip. Istnieje wiele formatów, jednakże standard DLNA pomija te cechy filmów i nie specyfikuje jakie rodzaje napisów powinny być obsługiwane oraz w jaki sposób. Informacje o sposobie udostępniania napisów zostały określone jedynie w specyfikacji UPnP AV dla urządzeń renderujących [<http://www.upnp.org/specs/av/UPnP-av-RenderingControl-v3-Service-20101231.pdf>].

Brak jednoznacznych rozwiązań w kwestii napisów do filmów, sprawił, że producenci w wielu przypadkach we własnym zakresie zapewniają ich wsparcie. Odtwarzacze multimedialnych podłączonych przez złącze USB w telewizorach ma zazwyczaj możliwość odtwarzania napisów w wersji tekstowej, w formatach MicroDVD lub SubRIP. Takie samo podejście zostało zastosowane w odtwarzaczach DLNA. Obsługują one jeden z dwóch rodzajów napisów, a w gestii serwera leży, aby w odpowiednim formacie takie napisy udostępnić (lub samodzielnie przekonwertować do obsługiwanego formatu).

### 2.3.3 Kodowanie i kompresja filmów

Niniejszy podrozdział jest poświęcony jedynie tematyce kodowania filmów, z pominięciem roli kontenerów multimedialnych. Kodowanie służy do przedstawienia materiału wideo w reprezentacji kodu binarnego. Kompresja natomiast to takie kodowanie, które zmniejsza wielkość strumienia

wideo, przy spadku jakości. Obecnie najpopularniejszymi koderem jest kodek MPEG-4 AVC (ang. *Advanced Video Coding*). Jest to obecnie najbardziej wydajny standard kodowania, pozwalający uzyskać bardzo dobrą jakość obrazu, o rozdzielczości do Full HD (1920 x 1080i). Coraz powszechniej jest wykorzystywany w transmisjach telewizyjnych, w kompresji filmów Blu-ray. Opisywany kodek jest zamiennie nazywany MPEG-4 AVC lub H.264. W fazie początkowej były to dwa różne standardy, jednakże w kolejnych etapach projekty zostały połączone. Standard jest opisany w dokumencie *MPEG-4 Part 10, Advanced Video Coding* [35].

Urządzenia DLNA (i nie tylko) posiadają wbudowane dekodery formatów wideo. Obecnie najpopularniejszym dekodery znajdującym się w kartach graficznych, telewizorach i innych urządzeniach odtwarzających jest dekodery MPEG-4 AVC/H.264. Są to najnowocześniejsze dekodery umożliwiające odtworzenie prawie każdego dostępnego współcześnie materiału wideo.

## **II Implementacja serwera UPnP**

# Oprogramowanie serwera UPnP AV

Większość współczesnych odtwarzaczy Blu-ray, telewizorów, dekodek i tym podobnych urządzeń RTV, posiada złącze Ethernet (lub moduł *WiFi*) i umożliwia podłączenie urządzenia do sieci lokalnej i Internetu. Takie urządzenia posiadają oprogramowanie umożliwiające wyszukiwać oraz odtwarzać materiały udostępniane przez serwery UPnP AV oraz DLNA. Autor na potrzeby niniejszej pracy stworzył w pełni funkcjonalny serwer UPnP AV współpracujący z urządzeniami UPnP i DLNA. Głównym celem aplikacji było umożliwienie kooperacji dwóch urządzeń ze sobą, w taki sposób, aby odtwarzacz potrafił wykorzystać jak największą możliwą ilość funkcji określoną w specyfikacjach UPnP AV i DLNA.

## 3.1 Wymagania wstępne

Zdefiniowanie środowiska do implementacji oprogramowania było bardzo ważnym wyborem. Autor do codziennej pracy używa systemu operacyjnego z rodziny Linux, jednakże większość użytkowników komputerów typu PC używa systemu operacyjnego Microsoft Windows. Dlatego też środowisko implementacji ma zapewniać kompatybilność z oboma systemami operacyjnymi. Pierwszym wyborem ze względu na biegłą znajomość języka było środowisko programistyczne języka Java.

W dzisiejszych czasach coraz więcej urządzeń sieci lokalnej nie jest oparta o architekturę PC. Przykładami takich urządzeń mogą być routery lub smartfony. Pierwsze z nich, coraz częściej są wyposażone w złącza USB oraz posiadają załączone oprogramowanie do udostępniania multimediów. Z doświadczeń własnych autora, wynika jednak, że zakres możliwości oraz wydajność takiego oprogramowania pozostawia wiele do życzenia. Skutecznie udostępniane są jedynie zdjęcia, natomiast użytkownicy natrafiają na problemy w kwestii odtwarzania filmów oraz muzyki.

W tabeli 3.1 zostały zdefiniowane platformy konieczne do obsługiwanego przez serwer UPnP AV.

**Tabela 3.1:** Platformy obsługiwane przez serwer UPnP AV

| Klasa urządzenia | System operacyjny |
|------------------|-------------------|
| Router           | DD-WRT            |
|                  | Tomato            |
| PC               | Windows           |
|                  | Linux             |

W przypadku środowiska Java wymienione we wcześniejszych akapitach wymagania mogą być z łatwością realizowane. Jednakże obciążenie procesora (system operacyjny Ubuntu 10.10) dla prostej aplikacji UPnP wynosiło około 5%. Analiza materiałów udostępnianych w Internecie oraz doświadczenie z użytkowania aplikacji Java, poddawała w wątpliwość wydajną pracę serwera zaprojektowanego w tym języku na mało wydajnych urządzeniach (np. routery). Po kilku próbach

uruchomienia serwera Tomcat na routerze firmy Linksys E3000, autor podjął decyzję o dalszym szukaniu środowiska do stworzenia aplikacji. Ostateczny wybór padł na język Python do realizacji oprogramowania.

Pozostałe wymagania zostały narzucone przez standardy UPnP AV i DLNA, co prosto można określić jako wymaganie pełnej współpracy z certyfikowanymi urządzeniami DLNA lub UPnP. Głównymi wymaganiami są:

- współpraca z podstawowymi urządzeniami DLNA - telewizorami, odtwarzaczami BluRay itd.,
- odtwarzanie (o ile podstawowy odtwarzacz urządzenia ma taką możliwość) kontenerów muzycznych i filmowych np. plików Matroska - MKV,
- odtwarzanie napisów w filmach i muzyce (np. do karaoke, specjalny tryb serwera), zarówno z plików tekstowych, jak i z kontenerów (w tym przypadku tylko pliki Matroska, MKV),
- przewijanie materiału,
- udostępnianie materiału w pełnej jakości,
- dodawanie / usuwanie udostępnianej zawartości bez konieczności ponownego uruchamiania oprogramowania,
- tworzenie albumów, klasyfikacji treści w logicznych katalogach.

W oprogramowaniu autor jako opcjonalne wymaganie zdefiniował renderowanie filmów do gorszej jakości w locie. W chwili obecnej większość nowoczesnego sprzętu na rynku obsługuje dużą ilość formatów i mimo swoich możliwości to oprogramowanie odtwarzacza DLNA jest przestarzałe lub stworzone z minimalnymi wymaganiami certyfikacji, co powoduje komunikat o niekompatybilności materiału, pomimo, że w rzeczywistości taka sytuacja nie ma miejsca. Z tego powodu autor skupił się podczas tworzenia oprogramowania serwera na ominięciu przeszkód postawionych przez większość producentów i wymuszenie odtwarzania materiałów na każdym urządzeniu.

Nie wszystkie urządzenia posiadają interfejs graficzny użytkownika. Interfejs użytkownika w serwerze UPnP AV/DLNA jest dostarczane wraz z oprogramowaniem serwera www i interfejsem strony internetowej do zarządzania serwerem oraz uruchamiania go.

## 3.2 Wykorzystane narzędzia

Aplikacja serwera UPnP AV została napisana w języku Python. Jej szkielet został oparty o zmodyfikowaną wersję framework'a *Coherence*, opisanego w rozdziale 3.2.1. Druga aplikacja, zapewniająca zarządzanie serwerem została napisana w języku Python, HTML oraz Javascript z wykorzystaniem popularnego framework'a Django.

Aplikacja jest zarządzana przez narzędzie kontroli wersję Git na darmowym serwerze Github.com

Wykorzystane narzędzia na potrzeby opisywanego zestawu aplikacji to:

- platforma Eclipse 3.6,
  - języki programowania: Python, HTML, Javascript, **Django**, T-SQL,
  - framework'i
    - \* *Coherence DLNA*,
    - \* *Django*,
    - \* *JSONRPC*,
    - \* *Storm*.
- GIMP do tworzenia grafiki aplikacji
- odtwarzacze UPnP AV / DLNA do testów
- GIT - do zarządzania wersjami oprogramowania.

### 3.2.1 Framework'i UPnP

W internecie można odnaleźć bardzo dużo framework'ów, które obsługują UPnP AV oraz DLNA. Przed wyborem języka programowania autor skupił się na poszukiwaniu odpowiedniego środowiska, który wspiera komunikację w sieci multimedialnej, posiada darmową, otwartą do modyfikacji licencję oraz nie obciąża w znaczący sposób systemu operacyjnego. Autor nie odnalazł framework'a, który natywnie wspierałby specyfikację DLNA. Jest to spowodowane prawdopodobnie koniecznością ponoszenia kosztów licencji DLNA. Liczba framework'ów, które wspierają protokół UPnP oraz UPnP AV, pomimo że nie jest duża, to jest w zadowalający sposób zaimplementowana, dzięki czemu autor nie musiał zaprogramować niższych warstw opisywanego protokołu.

Pierwszym, prawdopodobnie najnowszym i najczęściej aktualizowanym jest oprogramowanie *Cling* [27]. Jest to biblioteka, której zadaniem docelowym (jest wciąż w fazie rozwoju i wiele funkcji nie jest jeszcze zaimplementowanych) jest pełne wsparcie protokołu UPnP. Nie implementuje zmian wprowadzonych przez zrzeszenie DLNA. Jest napisana w języku Java, możliwe jest jej wykorzystanie zarówno w oprogramowaniu PC, jak i oprogramowaniu Javy dla smartfonów z systemem operacyjnym Android. Autor odrzucił jej wykorzystanie, gdyż w ówczesnym stadium rozwoju implementowała zbyt małą liczbę koniecznych do komunikacji funkcji oraz niemożliwe byłoby jej wykorzystanie na urządzeniach o bardzo małych możliwościach wydajnościowych (np. routery).

Drugim brany pod uwagę jako podstawę do aplikacji serwera UPnP framework'iem był *CyberLink for Java* [28]. Jest to pakiet oprogramowania umożliwiający twórcom aplikacji komunikację w sieci UPnP. Posiada wiele popularnych dla oprogramowania UPnP przykładów, np. włączanie i wyłączanie żarówki. Niestety w chwili porównywania możliwości framework'ów opisywany projekt nie był rozwijany i jego ostatnia aktualizacja datowana była na 2003 rok. Natomiast w trakcie pisania niniejszej pracy (już po stworzeniu oprogramowania) autor zauważył wznowienie projektu poprzez dodanie wsparcia dla pisania aplikacji UPnP do obsługi urządzeń mobilnych z systemem operacyjnym Android. *CyberLink for Java* jest rozwijany na licencji nie zezwalającej na modyfikację kodu źródłowego.

Autor rozważał również wykorzystanie bibliotek popularnego w środowisku *Linux* programu MiniUPnP [29] lub MiniDLNA [30]. Niestety oba umożliwiają wykorzystanie we własnym oprogramowaniu bibliotek klienckich, a celem projektu było stworzenie serwera UPnP AV.

Ostatnim opisanym w niniejszym podrozdziale będzie *Coherence DLNA/UPNP Framework for Digital Living* [31]. Pomimo, że ostatni to spełnił najwięcej wymagań stawianych przez autora i dlatego został przez niego wybrany jako szkielet napisanej aplikacji. *Coherence* jest zarówno programem udostępniającym prosty serwer multimedialny, serwer renderujący, posiada wiele wtyczek do popularnych programów (np. Rhythmbox, Totem, Picasa). Jest na bieżąco rozwijany, zaś jego licencja - MIT [32], zezwala na dowolną modyfikację. Jest napisany w języku Python. Dla deweloperów najważniejsze jest, że oprogramowanie zapewnia komunikację (wykrywanie, ogłaszanie urządzeń) oraz posiada „szablony”, to jest przykładowe aplikacje z funkcjonalnością udostępniania albumów, filmów itd. Niestety testy oprogramowania wykazały, że sposób implementacji protokołu nie zapewnia komunikacji z wieloma urządzeniami (w tym mobilnymi). Autor zdecydował się wykorzystać w swoim oprogramowaniu opisywany framework oraz wprowadzić w nim niezbędne modyfikacje, co dzięki liberalnej licencji MIT było w kwestiach prawnych możliwe.

## 3.3 Opis programu

Zaprojektowana aplikacja do udostępniania materiałów multimedialnych, jest to w rzeczywistości zestaw dwóch niezależnych od siebie programów. Pierwsza aplikacja jest serwerem UPnP AV który komunikuje się bezpośrednio z odtwarzaczami DLNA oraz UPnP AV. Natomiast druga aplikacja służy do zarządzania współdzielonymi zasobami oraz ustawieniami „technicznymi” serwera.

Aplikacja posiada wbudowane wyszukiwanie multimedialnych różnych rodzajów w lokalizacjach, które użytkownik wcześniej zdefiniował. W przypadku podkatalogów stosowane jest przeszukiwanie

rekursywne. Serwer UPnP AV interpretuje zawartość katalogów i określa te pliki, które są filmami, plikami tekstowymi (które są napisami do filmów), muzycznymi oraz zdjęciami. Pliki w odtwarzaczu mogą być przeglądane w katalogach oddających strukturę w systemie plików serwera oraz w kategoriach (wirtualnych katalogach) zdefiniowanych przez użytkownika.

## 3.4 Funkcje programu

Aplikacja składa się z dwóch modułów:

- serwer UPnP AV,
- aplikacji zarządzającej.

Funkcje pierwszego z modułów to:

- ogłaszanie aktywności dla innych urządzeń UPnP,
- wysyłanie, odpowiadanie na aktywność UPnP,
- wyszukiwanie multimediiów: filmów, plików z napisami, muzyki, obrazów,
- tworzenie biblioteki multimediiów:
  - zgodnie ze strukturą katalogów,
  - według typów plików (sortowanie według typów multimedialnych MIME),
  - według typów zawartości:
    - \* Muzyka,
    - \* Wideo,
    - \* Zdjęcia.
- udostępnianie w językach:
  - angielski,
  - polski,
- udostępnianie informacji o plikach
  - rozdzielczość,
  - rodzaj,
  - rozszerzenie,
  - czas trwania,
  - jakość,
  - data dodania,
  - rozmiar pliku,
  - lokalizacja pliku.

### 3.4.1 Biblioteka multimediiów

Biblioteka multimediiów jest zbiorem plików udostępnianych przez serwer UPnP AV. W opisywanej aplikacji biblioteka została podzielona na kilka głównych kategorii. Pierwszą z nich jest odwzorowanie zgodnie z położeniem w systemie plików (gdy to możliwe wyświetlana jest tylko nazwa katalogu nadrzędnego, w którym znajduje bądź znajdują się pliki multimedialne). Oznacza to, że przy pomocy odtwarzacza UPnP AV lub DLNA użytkownik może nawigować do przeglądania zawartości serwera i do sekcji „Katalogi” (ang. „*Directories*”). W dalszej kolejności wyświetlane są nazwy katalogów z plikami multimedialnymi.



Kolejnym krokiem nawigacji może być „Typy MIME” (ang. „*Mimetypes*”), w którym selekcjonowane materiały są dzielone według swoich typów MIME.

Dalszymi dostępnymi ścieżkami są następujące kategorie:

- Wideo (ang. *Video*),
- Muzyka (ang. *Music*),
- Zdjęcia (ang. *Images*),

Pliki są segregowane przez algorytm zgodnie ze swoją kategorią (zgodnie z typem MIME), a następnie umieszczane w odpowiadającej kategorii bibliotece. Typ MIME w swojej nazwie dla każdej z bibliotek w pliku posiada cechę wspólną. W niniejszym przypadku pliki graficzne posiadają w typie MIME słowo *image*, muzyka słowo *audio*, natomiast materiały wideo, słowo *video*. Upraszcza to w dużym stopniu sposób kategoryzowania plików. Przykładowy sposób selekcji plików, zgodnie z jego rozszerzeniem został przedstawiona w tabeli 3.2. Typ MIME jest rozpoznawany, dzięki rozszerzeniu pliku, dlatego wybór jakiego typu jest plik, należy do aplikacji. Oprogramowanie stworzone na potrzeby niniejszej pracy magisterskiej posiada wbudowany słownik, odwzorowujący rozszerzenie na typ MIME.

**Tabela 3.2:** Katagolowanie typów MIME - przykład

| Typ MIME         | Rozszerzenie pliku | Kategoria |
|------------------|--------------------|-----------|
| audio/mpeg       | mp3                | Muzyka    |
| audio/x-wav      | wav                | Muzyka    |
| image/png        | png                | Zdjęcia   |
| image/jpeg       | jpeg, jpg          | Zdjęcia   |
| video/mpeg       | mpeg mpg           | Wideo     |
| video/x-matroska | mkv                | Wideo     |
| video/x-msvideo  | avi                | Wideo     |

Dla każdego odnalezionego katalogu oraz pliku multimedialnego tworzony jest obiekt *MediaItem*, który przechowuje niezbędne informacje dla serwera UPnP AV. Dzięki odczytaniu informacji o obiekcie możliwy jest zwrócenie odtwarzaczom posiadanych informacji o pliku, katalogu lub katalogu wirtualnym.

Parametry możliwe do zwrócenia dla każdego z wcześniej wymienionych typów są umieszczone w tabeli 3.3

**Tabela 3.3:** Parametry dla kategorii UPnP

| Pobierany parametr   Rodzaj | Katalog | Wideo | Audio | Zdjęcia |
|-----------------------------|---------|-------|-------|---------|
| Ścieżka URL                 | x       | x     | x     | x       |
| Ilość podkatalogów          | x       |       |       |         |
| Podkatalog                  | x       |       |       |         |
| Obiekt <i>MediaItem</i>     | x       |       |       |         |
| Katalog nadrzędny           | x       | x     | x     | x       |
| Inne katalogi nadrzędne     | x       | x     | x     | x       |
| ID                          | x       | x     | x     | x       |
| Nazwa                       | x       | x     | x     | x       |
| Okładka (miniatura)         |         | x     | x     | x       |
| Typ MIME                    | x       | x     | x     | x       |
| Rozmiar                     |         | x     | x     | x       |
| Czas trwania                |         | x     | x     |         |
| Bitrate                     |         | x     | x     |         |
| Kanały audio                |         | x     |       |         |
| Napisy                      |         | x     |       |         |
| Rozdzielczość               |         | x     |       | x       |

Biblioteka multimediiów umożliwia dodawanie nowych kategorii w trakcie działania aplikacji bez konieczności przerywania jej działania. Odpowiada za to parametr zdefiniowany w specyfikacji UPnP - parametr aktualizacji (ang. *updateID*), który jest numerem sekwencyjnym określającym ilość zmian w danej bibliotece. Jeżeli odtwarzacz posiada dane o plikach z nieaktualnym parametrem aktualizacji, wówczas pobiera od nowa listę multimediiów. Aplikacja w podobny sposób reaguje na dodanie i usunięcie pliku (plików) do istniejącego już, w ścieżce wyszukiwania, katalogu.

### 3.4.2 Miniatury multimediiów

Miniatury multimediiów nie są obowiązkową treścią, którą musi udostępniać serwer DLNA. Jednakże sposób udostępniania ich nie odbiega znacząco od sposobu udostępniania innych materiałów, dlatego autor uznał, że warto tą funkcję również zaimplementować.

Większość współczesnych systemów operacyjnych tworzy miniatury filmów oraz obrazów na własne potrzeby oraz często pobiera okładki albumów muzycznych z Internetu. Autor zdecydował się w pierwszej kolejności korzystać z miniatur stworzonych przez system operacyjny, zaś w dalszej kolejności z miniatur tworzonych przez aplikację (w przypadku gdy miniatura systemu operacyjnego nie istnieje lub nie spełnia wymogów co do rozmiarów). Miniatura tworzona przez aplikację ma rozmiar 120 x 120 pikseli i jest zapisana w standardzie JPEG z tą samą nazwą pliku jaką posiada główny plik multimedialny, z przedrostkiem TH (od ang. *Thumbnail* - miniatura).

### 3.4.3 Zarządzanie aplikacją

Serwer zarządzający serwerem multimediiów DLNA jest to aplikacja internetowa przygotowana w oparciu o framework *Django*. Serwer WWW nasłuchuje domyślnie na porcie 8080 (zmiana portu jest możliwa poprzez modyfikację plików konfiguracyjnych *Django*). Autor zaleca wykorzystywać serwer dostarczany wraz z *Django*, lecz nie ma żadnych przeciwwskazań, aby wykorzystywać inne oprogramowanie serwerowe (np. serwer *Apache*). Aplikacja zarządzająca podobnie, jak serwer UPnP AV, jest dostępna w dwóch wersjach językowych: polskim i angielskim.

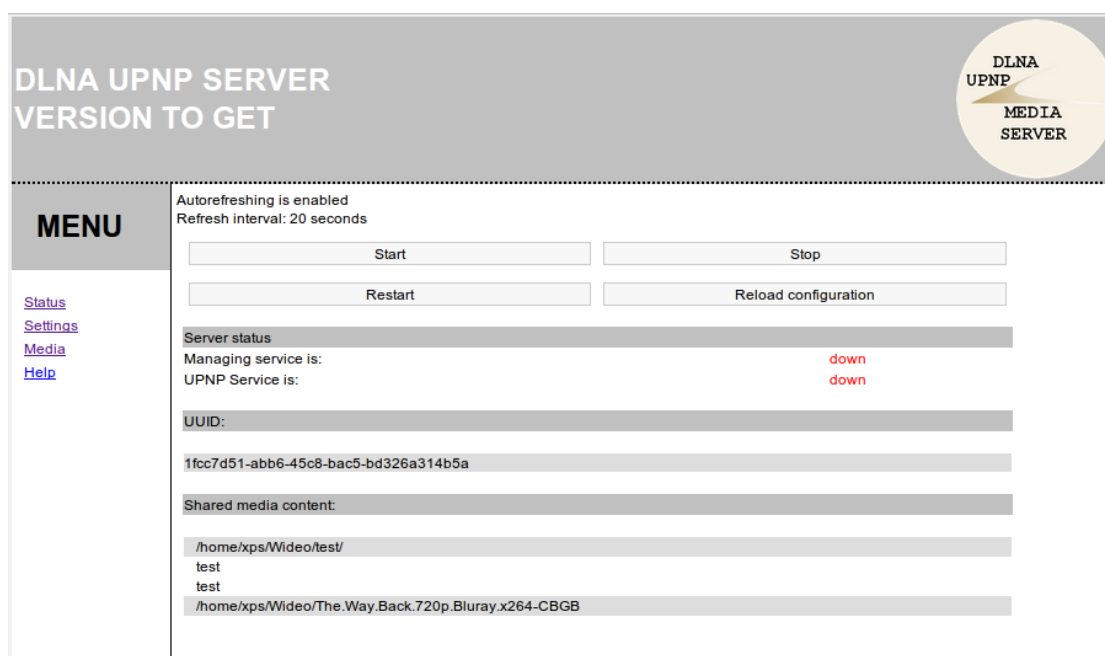
Do zarządzania opcjami aplikacji konieczne jest wykorzystanie przeglądarki WWW i otworzyć stronę np. <http://localhost:8080/>.

W głównym menu można wybrać następujące opcje (pierwsza nazwa w języku polskim, druga w języku angielskim):

- status - *status*,
- ustawienia - *settings*,
- multimedia - *media*,
- pomoc - *help*.

Pierwszy ekran widoczny na rysunku 3.1 wyświetla status serwera UPnP oraz serwera RPC. Serwer RPC jest niezbędny do zmiany jakichkolwiek opcji, natomiast serwer UPnP AV jest niezbędny do udostępniania multimediów w sieciach UPnP AV oraz DLNA. Obie usługi mogą działać niezależnie, jednakże autor dla uproszczenia sposobu używania aplikacji uniemożliwił selektywne uruchamianie usług.

**Rys. 3.1:** Status serwera UPnP AV



Druga dostępna opcja, czyli „Ustawienia” (ang. *settings*) umożliwia modyfikację podstawowych funkcji serwera UPnP AV. W niniejszej sekcji możliwa jest zmiana następujących opcji:

- nazwa serwera UPnP AV,
- adres IP serwera UPnP AV (domyślnie serwer działa na wszystkich adresach sieci lokalnej),
- port serwera UPnP AV (domyślnie losowy),
- wzorzec ignorowania - np. ignorowanie plików o podanym rozszerzeniu - \*.avi (domyślnie brak wzorca),
- stwórz oddzielny konterer typów MIME (domyślnie włączona),
- transkodowanie (domyślnie wyłączone),
- własna ikona serwera,
- maksymalna ilość przeszukania rekursywnego (domyślnie nieorganiczna) - ile folderów „w dół” rekursywnie ma przeszukiwać algorytm tworzenia biblioteki multimediów.

Opisywana opcja została przedstawiona na rysunku 3.2

Rys. 3.2: Ustawienia serwera UPnP AV

DLNA UPnP SERVER VERSION TO GET

**MENU**

- [Status](#)
- [Settings](#)
- [Media](#)
- [Help](#)

Autorefreshing is enabled

Name: buuu

IP Address: 192.168.1.129

Port: 39806

Ignore patterns:

Do mimetype containers: ☒

Transcoding: ☐

Icon: ☐

Max child items: ☐

Save

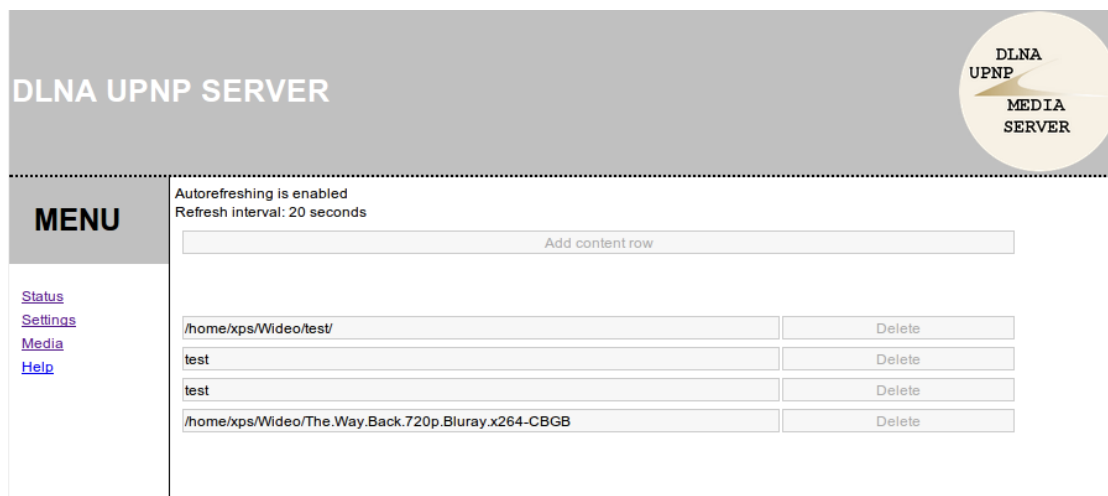
Cancel

Kolejną opcją jest wybieranie katalogów do wyszukiwania multimediów. Została ona przez autora przedstawiona w wersji skróconej jako „Multimedia” (ang. *Media*). W niej możliwe jest dodawanie katalogów do przeszukiwania. Po dodaniu lub usunięciu wiersza ze ścieżką konieczne jest ręczne zapisanie zmiany poprzez wybranie przycisku „Zapisz” (ang. *Save*). Przycisk nie jest widoczny na rysunku 3.3, który przedstawia opisywaną w niniejszym akapicie część aplikacji.

Dodanie multimediów jest możliwe poprzez ręczne wpisanie absolutnej ścieżki do katalogu. Ścieżka relatywna może powodować niepoprawne wyszukiwanie materiałów np. ścieżka „~/Wideo” może być poprawnie przeszukana, gdy serwer jest uruchomiony przez tego samego użytkownika, co pożądaný katalog Wideo, lecz niepoprawnie, gdy uruchomił go inny użytkownik systemu. Drugim sposobem dodania katalogu jest wybór opcji „Przeglądaj” (ang. *Browse*) i wybranie folderu, który zostanie przeszukany. W drugim przypadku ścieżka do folderu zostanie dodana do linii absolutnej ścieżki do katalogu (jest to opcja wygodniejsza dla niektórych użytkowników).

W przypadku braku katalogu lub innych problemów serwer WWW wykonuje sprawdzenie poprawności wpisanych danych. W przypadku braku katalogu (lub innego błędu wyszukania folderu) obok wiersza ze ścieżką pojawia się czerwony znak 'X'. Jednakże aplikacja zarządzająca zezwoli na dodanie ścieżki do bazy danych i wyszukanie jej przez serwer UPnP AV i to do niego należy ewentualne zadanie ignorowania niepoprawnych ścieżek.

Rys. 3.3: Wybieranie multimediów dla serwera UPnP AV



Ostatni z dostępnych ekranów, to jest „Pomoc” (ang. *Help*). Znajduje się w nim opis wszystkich opcji dostępnych w serwerze WWW, przybliżony sposób wyszukiwania serwera w odtwarzaczu DLNA oraz uproszczona specyfikacja serwera DLNA. Istnieją w niej informacje o wyszukiwanych plikach multimedialnych, sposobie dołączania załączników do plików (np. napisy), opis tworzenia miniatur oraz wymagane oprogramowania do działania opisywanej aplikacji. Plik pomocy dostępny jest również w postaci pliku PDF.

## 3.5 Budowa wewnętrzna modułu serwera UPnP AV

Serwer multimedialny UPnP AV / DLNA został wykonany w języku Python. Do jego realizacji wykorzystany został framework *Coherence DLNA/UPNP Framework for Digital Living* którego opis znajduje się w rozdziale 3.2.1. Zestaw oprogramowania został podzielony na dwie niezależne aplikacje, serwer multimedialny UPnP AV oraz serwer zarządzający. Każdy z nich składa się z wielu podmodułów. *Dla rozróżnienia modułu serwera UPnP AV od aplikacji głównej, pierwszy moduł w niniejszej pracy będzie nazywany serwerem multimediów.*

Serwer multimedialny UPnP AV udostępnia swoje funkcje poprzez protokół UPnP w wersji tylko do odczytu, oznacza to, że nie można modyfikować opcji serwer przy pomocy poleceń protokołu UPnP. Zarządzanie serwerem jest możliwe, poprzez wykorzystanie poleceń RPC wysyłanych do serwera RPC, będącego modulem wewnętrznym (podmodulem) serwer UPnP AV. Polecenia do integralnego serwera RPC wysyłane są z drugiej aplikacji, serwera zarządzającego. Poprzez zarządzanie rozumiane są takie funkcje jak:

- uruchomienie / zatrzymanie serwera UPnP,
- uruchomienie / zatrzymanie / uruchomienie ponowne serwera RPC,
- zatrzymanie / uruchomienie ponowne serwera zarządzającego,
- zmiana interfejsu sieciowego serwera UPnP, bądź RPC,
- zmiana portu zarządzającego serwerem RPC,
- dodawanie, modyfikacja, usuwanie ścieżek, z których są udostępniane multimedia,
- tworzenie, modyfikacja, usuwanie wirtualnych bibliotek UPnP AV,
- wymuszanie ponownego przeskanowania folderów w celu znalezienia plików multimedialnych,
- zmiana loga serwera UPnP, widocznego w odtwarzaczu UPnP AV bądź DLNA.

Dokładny opis funkcjonalny oraz interfejs użytkownika, serwera zarządzającego zostały przedstawione w podrozdziale 3.4.3.

W całej aplikacji wykorzystywane są bazy danych *SQLite* (zawierającymi np. ścieżki do przeszukiwania). Jest to wybór stosunkowo oczywisty w kontekście umożliwienia działania na urządzeniach o niskiej wydajności. Baza danych *SQLite* została stworzona z myślą o przechowywaniu małej ilości rekordów. Obsługuje się ją poprzez zewnętrzną bibliotekę. W wypadku opisywanej aplikacji wykorzystany został framework *SQLAlchemy* [36], który dzięki swojej rozbudowanej dokumentacji, mógł zostać stosunkowo szybko i w łatwy sposób wdrożony przez autora. Przy dużej ilości danych powoduje to nadmierne obciążenie aplikacji. Jednakże w przypadku wielu małych aplikacji, jak i niniejszego serwera UPnP, rozwiązanie to eliminuje konieczność instalowania dodatkowego oprogramowania bazodanowego (np. *MySQL*) działającego zazwyczaj w trybie klient - serwer (wprowadza to dodatkowy narzut komunikacyjny). [33]

Główna aplikacja, która udostępnia multimedia w sieci UPnP AV i/lub DLNA ma możliwość uruchomienia w tle (w trybie demona zgodnie z nazewnictwem systemu Linux lub usługi, zgodnie z nazewnictwem systemu Windows) lub w trybie pierwszoplanowym (ang. *foreground*), w którym wszystkie komunikaty są wyświetlane w oknie wiersza poleceń. W pierwszej kolejności aplikacja otwiera pliki baz danych, z których wczytywane są ustawienia oraz ścieżki wyszukiwania plików multimedialnych. W przypadku poprawnego wczytania wszystkich danych aplikacja tworzy główny obiekt *MediaStore*, który jest klasą nadrzedną dla wszystkich udostępnianych obiektów *MediaItem*. Ich opis oraz sposób działania został przedstawiony w dalszej części niniejszej pracy.

Jej uogólniony schemat działania widoczny jest na diagramach stanów oraz komunikacji (rysunku ).

### 3.5.1 Klasa demona

Zadaniem klasy demona, *MediaDaemon* jest uruchamianie aplikacji w tle. Nie została ona zaprojektowana jako niezbędna klasa na potrzeby działania aplikacji, aby relatywnie proste było uruchomienie programu, zarówno w tle, jak i w trybie pierwszoplanowym.

Klasa demona przyjmuje następujące parametry linii komend (pierwszy parametr, jest parametrem domyślnym):

- *start*,
- *stop*,
- *restart*.

Opisywany obiekt jest zabezpieczony przed wielokrotnym uruchomieniem (co mogłoby oznaczać modyfikację tych samych danych i ich uszkodzenie), poprzez tworzenie pliku o nazwie *daemon.pid*, który przechowuje numer identyfikacyjny uruchomionego procesu demona. Wykorzystywany jest on również do ponownego uruchamiania oraz zatrzymywania aplikacji serwera UPnP AV.

### 3.5.2 Klasa główna serwera UPnP

Klasa główna, którą jest serwer multimediiów, *MediaServer*, jest klasą inicjalizującą działanie w sieci UPnP serwera multimedialnego.

W pierwszej kolejności wczytywana jest baza danych multimediiów, następnie zaś są tworzone obiekt *MediaStore*, który w swoim algorytmie działania ma zaimplementowane przeszukiwanie ścieżek katalogów, wczytywanie ustawień dostarczonych przez opisywaną w niniejszym rozdziale klasę oraz tworzenie obiektów *MediaItem*, które dla użytkownika są kontenerami multimediiów bądź docelowymi plikami multimedialnymi.

### 3.5.2.1 Obiekt *MediaStore*

Klasa *MediaStore* jest pierwszą klasą (w ścieżce uruchamiania aplikacji), która jest dziedziczona po klasach framework'a *Coherence*. Od momentu uruchomienia tego obiektu serwer UPnP AV jest widoczny w odtwarzaczach UPnP oraz DLNA. Otrzymuje ona od klasy nadrzędnej w trakcie uruchamiania parametry działania, wśród nich ścieżki do katalogów, w których mają być wyszukiwane multimedia. Na podstawie otrzymanych danych tworzy następujące kontenery główne:

- kontener „Muzyka” (ang. *Music*),
- kontener „Wideo” (ang. *Video*),
- kontener „Zdjęcia” (ang. *Images*),
- kontener „Katalogi” (ang. *Directories*),
- kontener „Typy MIME” (ang. *MIMETypes*) - opcjonalny.

Po utworzeniu powyższych kontenerów, algorytm wczytuje w kolejności wszystkie pliki multimedialne i przydziela je do odpowiednich kategorii. W przypadku wczytywania folderu (wewnątrz którego jest wybrany do udostępniania plik), tworzony jest kolejny kontener, który jest podkontenerem, dla biblioteki „Katalogi”.

Wczytanie pliku multimedialnego lub tworzenie konteneru, precyzyjnie określając, jest stworzeniem obiektu *MediaItem* (jeden plik lub kontener, oznacza pojedynczy obiekt *MediaItem*). Podtyp ten został opisany w kolejnym podrozdziale.

### 3.5.2.2 Obiekt *MediaItem*

Klasa *MediaItem* przechowuje informacje o pojedynczym obiekcie schematu UPnP. Wyróżniamy tu takie typy jak:

- *container* - kontener,
- *VideoItem* - obiekt wideo,
- *MusicItem* - obiekt audio,
- *ImageItem* - obiekt graficzny.

Pierwszy z nich, kontener, odzwierciedla w systemie plików, katalogi. Pozostałe odzwierciedlają odpowiednie, dla swojego typu, pliki.

Opisywany obiekt zawiera informacje niezbędne do jego wyszukania oraz do odtworzenia przez odtwarzacz UPnP AV / DLNA. Posiada takie informacje, jak

- numer identyfikacyjny obiektu,
- nazwa (przyjazna dla użytkownika - nazwa pliku bez rozszerzenia),
- typ MIME,
- ścieżka do pliku,
- adres URL do pliku (przy jego pomocy odtwarzacz odtwarza materiał),
- nazwę hosta,
- kontenery nadrzędne, „rodziców” (o ile istnieją),
- kontenery podrzędne, „dzieci” (o ile istnieją),
- miniatura (jeżeli obiekt nie jest kontenerem).

Obiekt *MediaItem* tworzy dla odtwarzacza i przechowuje metadane dotyczące plików. Są one pobierane przy wykorzystaniu różnych narzędzi. Najprostszym przypadkiem jest typ graficzny, dla którego wykorzystywana jest wbudowana klasa języka Python - *Image*. Przy jej pomocy uzyskiwane są informacje o rozmiarze pliku i jego rozdzielczości.

Długość oraz przepływność (ang. *bitrate*) dla plików audio jest pobierana przy pomocy zewnętrznej klasy *mutagen.mp3* lub *wave*.

Dla materiałów wideo aplikacja wykorzystuje dwa sposoby na uzyskanie pomocnych dla siebie informacji. Metadane wbudowane w określony kontener multimedialny, bądź też zewnętrzne narzędzie *ffprobe*, który w prosty sposób dostarcza informacji o czasie trwania materiału, przepływności, rozdzielczości oraz ilości kanałów audio w ścieżce dźwiękowej.

Ostatnim zadaniem klasy *MediaItem* jest tworzenie miniatur. Mogą być tworzone przy pomocy dwóch narzędzi: *ffmpegthumbnailer* dla plików wideo lub przy pomocy klasy *Image* języka Python dla plików graficznych (klasa ta posiada specjalną metodę do tworzenia miniatur). W przypadku filmu wybierana jest klatka, która stanowi 33% długości materiału. Pliki takie przechowywane są w katalogu domowym użytkownika (zarówno w systemach rodziny Windows, jak i Linux), w ukrytym podkatalogach „*thumbnails/dlna*”.

Aplikacja „stara się” tworzyć miniatury filmów oraz obrazów, na własne potrzeby, jak najrzadziej to możliwe. Jeżeli miniatura danego pliku istnieje w systemie operacyjnym, oprogramowanie potrafi ją zlokalizować, ma prawa do jej odczytu oraz spełnia wymagania co do rozmiaru i typu (rozmiar 120 x 120 pikseli, typ JPEG lub PNG), wówczas do odtwarzacza przekazywana jest wersja miniatury utworzona przez system (uprzednio jest kopiowana do katalogu miniatur programu).

### 3.5.3 Modyfikacja frameworka UPnP Coherence

*Coherence DLNA/UPNP Framework for Digital Living* dostarcza programiście zestaw narzędzi, które kompleksowo zapewniają tworzenie komunikatów UPnP (DLNA). Framework może działać zarówno jako klient, jak i serwer. W niniejszej aplikacji wykorzystywana jest tylko część serwerowa zewnętrznego oprogramowania. Teoretycznie programista powinien ograniczyć się do stworzenia biblioteki multimediiów, wczytaniu jej do odpowiedniego obiektu serwera, a na końcu uruchomieniu go z odpowiednimi parametrami wejściowymi. Niestety w trakcie testów i implementacji serwera UPnP AV autor napotkał na szereg problemów we współpracy serwera z różnymi urządzeniami. Problemy rozpoczęły się na komunikacji z urządzeniami firm Samsung, LG. O ile z odtwarzaczami drugiej firmy serwer współpracował na akceptowalnym poziomie, o tyle z firmą Samsung nie współpracował w ogóle.

Z tego powodu autor pracy zdecydował się zmodyfikować niektóre klasy framework’a, tak aby aplikacja działała z każdym dostępnym dla autora do testowania urządzeniem. Zezwała na to liberalna licencja, na której udostępniany jest program *Coherence*, MIT [32].

Niektóre urządzenia (aplikacje) zostały zaprogramowane w sposób absurdalny dla użytkownika. To znaczy tak, aby współpracować tylko z urządzeniami tej samej firmy. Jednym z nich są odtwarzacze firmy *Microsoft* (*Microsoft Xbox 360*, *Windows Media Player*), które materiały wideo w jakości *HD Ready*, tylko gdy serwerem również jest urządzenie firmy *Microsoft*. Z tego powodu oprogramowanie musi „przedstawiać się” jako serwer multimedialny *Windows Media Connect*.

Ciekawym wątkiem jest format plików *Matroska*, w odtwarzaczach DLNA firmy Samsung. Odtwarzacz wbudowany w urządzenie (który odtwarza materiały poprzez złącze USB) potrafi odtworzyć opisywany kontener multimedialny. Urządzenie posiada sprzętowy dekodery H.264, a odtwarzacz DLNA mimo wszystko nie zgadza się na odtworzenie materiału. Autor podejrzewa, że powodem może być brak licencji na odtwarzanie tego typu plików lub zbyt dokładne podążanie za specyfikacją DLNA (lecz z drugiej strony filmy AVI są odtwarzane). W celu rozwiązania problemu autor zaprogramował serwer tak, aby dla niektórych urządzeń pliki *MKV* były przedstawiane poprzez inny typ MIME - *video/x-msvideo*. W przypadku gdyby, jakiś odtwarzacz nie współpracował z typem MIME *video/x-msvideo* (co w trakcie testów się nie zdarzyło), a użytkownik chciał zmienić ręcznie typ MIME, zmiana ta jest możliwa poprzez edycję tekstowego pliku konfiguracyjnego w głównym katalogu aplikacji.

Poza opisanymi w poprzednich akapitach zmianami (i zmianami opisanymi w podrozdziale



3.5.3.1) autor zmodyfikował metodę przeglądania (*upnp\_Browse*) oraz metodę wyszukiwania (*upnp\_Search*), zgodnie z wymaganiami większości odtwarzaczy DLNA.

### 3.5.3.1 Język DIDL-Lite

DIDL-Lite (ang. *Digital Item Declaration Language Lite*) jest językiem, który dostarcza schemat opisu urządzeń UPnP AV oraz DLNA. Jest to język oparty na schemacie dokumentu XML. W przypadku, gdy druga strona komunikacji nie umie zinterpretować części dokumentu powinna go ignorować, jednakże zazwyczaj odtwarzacze w urządzeniach znanych marek, nie odnajdują serwera, aby uniknąć ewentualnych błędów.

Bardzo ważnym elementem dokumentu w języku DIDL-Lite jest jego nagłówek. Określa on, z jakimi schematami jest zgodny dany plik XML. Przykładowy nagłówek DIDL-Lite został przedstawiony na rysunku 3.4.

**Rys. 3.4:** Przykładowy nagłówek dokumentu w języku DIDL-Lite

```
<xsd:schema targetNamespace="urn:schemas-upnp-org:metadata-1-0/DIDL-Lite/" xmlns:upnp="urn:schemas-upnp-org:metadata-1-0/upnp/" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:didl-lite="urn:schemas-upnp-org:metadata-1-0/DIDL-Lite/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0">
```

W wypadku urządzeń firmy Samsung, wymagany nagłówkiem, dzięki któremu urządzenie zostanie wyświetlone na liście serwerów odnalezionych jest:

- *xmlns:sec="http://www.sec.co.kr/"*

Informacja ta została uzyskana na podstawie podsłuchiwania pakietów TCP (wykorzystując narzędzie *tcpdump* lub *Wireshark*) wymienianych między serwerem DLNA firmy Samsung (aplikacja *AllShare*), a odtwarzaczem DLNA tej samej firmy (również aplikacja *AllShare*).

Ponadto autor na potrzeby obsługi urządzeń DLNA zaimplementował tagi zdefiniowane przez konsorcjum DLNA w specyfikacji. Część z parametrów była już zaimplementowana w pierwotnej wersji framework'a, jednakże dla wyjaśnienia ich znaczenia, na poniższej liście zostaną przedstawione wszystkie dostępne w dołączonym do pracy oprogramowaniu:

- DLNA.ORG\_PN - format multimediiów zgodny z DLNA (niestety niezgodne z typami MIME, tak jak w UPnP AV),
- DLNA.ORG\_OP - wartość '0' lub '1' - definiuje czy serwer obsługuje zaawansowane operacje przesyłania (np. przewijanie),
- DLNA.ORG\_PS - wartość dziesiętna - możliwe tempo odtwarzania materiału,
- DLNA.ORG\_CI - wartość '0' lub '1' - definiuje czy materiał jest transkodowany,
- DLNA.ORG\_FLAGS - dodatkowy ciąg 33 bitowych znaków określające dodatkowe parametry serwera oraz opisywanego pliku.

Przykładowymi formaty multimediiów w konwencji DLNA są: MP3 (typ MIME - audio/mpeg), WMABASE (typ MIME - audio/ms-wma), JPEG\_LRG / JPEG\_MED (typ MIME - image/jpeg), PNG\_LRG / PNG\_MED (typ MIME - image/png), MPEG\_TS\_PAL / MPEG\_TS\_NTSC (video/mpeg).

Plik *DIDL-Lite* musi być dla wielu urządzeń specjalnie sformatowany z dokładnie sprecyzowaną ilością parametrów. Parametry nadmiarowe często powodują, że urządzenie uznaje serwer UPnP AV / DLNA za niewłaściwy i nie umożliwia przeglądania jego zawartości. W pierwotnej wersji oprogramowania *Coherence* programiści przygotowali specjalnie parsowane dokumenty, dla konsol do gier, takich jak *Microsoft Xbox 360* oraz *Sony Playstation 3*. Autor dodał wsparcie dla urządzeń firm *Samsung* (telewizory oraz urządzenia mobilne), *Philips* (telewizory), *LG*

(telewizory oraz odtwarzacze Blu-Ray) oraz dla oprogramowania UPnP AV/DLNA dla komputerów stacjonarnych działających pod kontrolą systemów operacyjnych Windows oraz Linux (środowisko graficzne Gnome).

### 3.5.4 Klasa serwera RPC

Zewnętrzne oprogramowanie, dołączone do aplikacji, *Coherence DLNA/UPNP Framework for Digital Living*, nie ma zaimplementowanej modyfikacji parametrów serwera poprzez interpretację komunikatów UPnP. Autor podjął decyzję o zaimplementowaniu oddzielnego serwera poleceń RPC, umożliwiającego zmianę opcji poprzez modyfikację rekordów w bazie danych. Było to zdaniem autora prostsze do stworzenia rozwiązanie.

W tym wypadku została napisana klasa, która modyfikuje zmienne definiujące opcje serwera oraz modyfikuje dane wewnątrz bazy danych *SQLite*. Serwer RPC umożliwia nie tylko zmianę parametrów, ale również pobieranie ich wartości, na potrzeby opisywanego już wcześniej drugiego modułu oprogramowania, serwisu WWW. Informowanie zainteresowanych klas o modyfikacji parametrów odbywa się poprzez zaimplementowany zgodnie z wzorcem projektowym obserwatora.

Parametry możliwe do zmiany zostały przedstawione we wcześniejszym podrozdziale 3.4.3.

## 3.6 Budowa wewnętrzna modułu zarządzającego

Moduł zarządzający jest to aplikacja internetowa, w postaci strony WWW. Zapewnia on graficzny interfejs użytkownika, który umożliwia sprawdzenie statusu serwer UPnP AV / DLNA oraz zmianę istotnych dla użytkownika parametrów.

Aplikacja została napisana we frameworku *Django*, w języku programowania *Python*. Większość zapytań obsługuje poprzez zapytania AJAX (ang. *Asynchronous JavaScript and XML*), dzięki czemu strona WWW nie musi być odświeżana przez użytkownika w celu sprawdzenia stanu pozostałych modułów oprogramowania.

Program napisany we framework'u *Django* jest zgodny z wzorcem projektowym *MVC* (Model - Widok - Kontroler, ang. *Model - View - Controller*). Z tego powodu każda z części wzorca została opisana w kolejnych podrozdziałach. Widok oraz kontroler w nomenklaturze *Django* nazywane są w sposób odwrotny od ogólnie przyjętego schematu, to jest w pliku widoku (ang. *view*) zdefiniowana jest obsługa (kontrola) strony, zaś folder szablonów (ang. *templates*) jest widokiem wzorca MVC.

### 3.6.1 Model

Model definiuje strukturę tabel bazy danych aplikacji. Program posiada własną bazę danych (niezależną od bazy danych drugiego modułu). Baza danych, jak wszystkie tego typu struktury w zestawie aplikacji, jest bazą *SQLite*. Istnieją w niej cztery tabele, takie jak (podlista dla każdego elementu to pola tabeli):

1. Zawartość (ang. *Content*)
  - ID - numer identyfikacyjny,
  - Ścieżka (ang. *path*) - ścieżka przeszukiwania plików do dodania do biblioteki multi-mediów,
2. Parametry (ang. *DBContainer*)
  - ID - numer identyfikacyjny,
  - adres IP,

- port,
- nazwa,
- UUID - identyfikator serwer UPnP AV,
- Utworzony - wartość bool'owska definiująca, czy UUID jest utworzony,
- id sesji - identyfikator sesji WWW,
- kontener MIME - wartość bool'owska określająca czy serwer UPnP AV tworzy bibliotekę multimedialną kontenerów MIME,
- transkodowanie - włączenie lub wyłączenie transkodowania materiałów,
- obserwacja zmiany folderów - definiuje czy każda zmiana zawartości folderu, ma być automatycznie aktualizowana

3. Adres serwera RPC (ang. *DBAddress*)

- ID - numer identyfikacyjny,
- adres IP,
- port

4. Status usługi (ang. *ServiceStatus*)

- ID - numer identyfikacyjny,
- nazwa,
- stan - definiuje czy serwer podczas ostatniego sprawdzania był uruchomiony czy nie.

W powyższym schemacie, zwraca uwagę fakt zdublowania informacji o adresie IP oraz porcie. W tabeli „Parametry” adres ip oraz port są to informacje uzyskane na temat serwera UPnP, natomiast druga tabela zawiera dane o adresie IP oraz porcie serwera RPC, umożliwiające uzyskanie danych do tabeli „Parametry” oraz synchronizację ustawień między tabelą modułu zarządzającego „Parametry”, a jej odpowiednikiem w drugim module.

### 3.6.2 Widok

Rolę widoku w *Django* pełni katalog szablonów *templates*. Udostępniane są w nim szablony stron internetowych napisanych w języku *HTML*, *Javascript* (wraz z biblioteką *jQuery*) oraz wbudowane, w framework, tagi, które umożliwiają wczytanie danych w trakcie inicjalizacji (pobierania) strony. Dane do wspomnianych tagów udostępniane są przez kontroler aplikacji.

W opisywanej aplikacji oprócz skryptów *Javascript* oraz stron internetowych, dołączone są pliki kaskadowych arkuszy stylów *CSS* oraz elementy graficzne interfejsu użytkownika.

### 3.6.3 Kontroler

Częścią aplikacji, która zwraca użytkownikowi stronę internetową oraz uzyskuje do niej niezbędne dane (takie jak parametry serwera UPnP), jest kontroler. W nazewnictwie przyjętym w *Django*, plik nazywa się widokiem (ang. *views*).

Pierwsze zapytanie o plik główny jako jedynym nie jest zapytaniem wykonywanym techniką asynchroniczną AJAX. Wczytywany jest tu plik szablonu statusu oraz dodawane są do niego dane pobrane z tabeli „Parametry”. Dalsze informacje o zmianie statusu pobierane są poprzez cykliczne zapytania do serwera, wykonywane w tle. Wszystkie dane przesyłane w komunikatach AJAX są przesyłane w formacie *JSON* [39]. Pliki *JSON* tworzone są na dwa sposoby. Pierwszym sposobem jest parsowanie słownika za pomocą biblioteki *simplejson*, która tworzy ciąg znaków w pożądanym formacie. Drugim sposobem jest tworzenie pliku za pomocą „ręcznego” wpisywania do ciągu znaków odpowiednio spreparowanych komunikatów. Drugi sposób, pomimo, że prosty

i prymitywny, w przypadkach, gdy aplikacja przesyła krótkie komunikaty, takie jak widoczne na rysunku 3.5.

**Rys. 3.5:** Status serwera UPnP w postaci *JSON*

```
{ „Status” : „Running” }
```

W kontrolerze następuje uruchomienie demona serwer multimedków. Usługa jest uruchamiana w tle poprzez uruchomienie potoku zaimplementowanego w klasie *subprocess* języka Python.

# Testy

Oprogramowania odtwarzaczy UPnP AV oraz DLNA są bardzo zróżnicowane względem ilości udostępnianych funkcji. Powstają dość duże rozbieżności w sposobie obsługi serwerów, nawet w cechach wspólnych odtwarzaczy, czyli w sposobie wyszukiwania, przeglądania oraz odtwarzania multimedialnych. Z tego powodu serwer musi wiele urządzeń w różnoraki sposób obsługiwać, to znaczy preparować odpowiedzi przygotowane dla danego typu oprogramowania. W tym przypadku przez typ rozumiany jest producent, wytwórca aplikacji. Z wcześniej przedstawionych powodów oprogramowanie serwera multimedialnego UPnP AV musiało przejść szereg testów współpracy z różnymi urządzeniami. Autor planował stworzyć jednolity sposób testowania aplikacji, lecz w większości przypadków nie było to możliwe. Każdy z odtwarzaczy wymaga stworzenia własnej procedury wykonania poleceń protokołu UPnP AV / DLNA (odtwarzanie, przewijanie, pokaz slajdów itp.).

## 4.1 Sposób testowania aplikacji

Aplikacja została poddana kilkudziesięciu testom użytkowym oraz systemowym (nie zostały zaprogramowane i wykonane testy jednostkowe wytworzonego oprogramowania). Autor stworzył procedury sprawdzania poprawności działania aplikacji dla obu modułów aplikacji, serwera multimedialnego UPnP AV oraz serwera zarządzającego. W przypadku pierwszego modułu testom poddane zostały schematy działania dla większości stosowanych (w praktyce) poleceń specyfikacji UPnP AV oraz DLNA. Z powodu zróżnicowania aplikacji na różnych urządzeniach (oraz programach) odtwarzających multimedia, sprawdzanie zgodności modułu ze zleconymi zadaniami zostało wykonane poprzez praktyczne wykonanie odtwarzania treści multimedialnych.

W przypadku drugiego modułu, konieczne do wykonania testy, były prostsze oraz mniej skomplikowane. Konieczne było przetestowanie współpracy modułu z kilkoma (najpopularniejszymi) przeglądarkami internetowymi oraz reakcja na wydawanie poleceń, takich jak dodanie katalogu z multimediami czy zmiana ustawień.

Ponadto autor wykonał kilka testów „wspólnych” dla obu modułów. Gdy w jakimkolwiek stopniu zachodzi interakcja między modułami wykonane zostało sprawdzenie czy reakcja (np. przeszukanie katalogów) jest poprawna. Za bezwzględnie konieczny, autor uznał, test w sytuacjach awaryjnych, gdy jeden z modułów nie odpowiada. Oprogramowanie w sytuacjach awaryjnych musi powiadomić użytkownika o zaistniałym błędzie lub samodzielnie spróbować rozwiązać problem. Druga sytuacja ma miejsce, tylko w jednym przypadku, to znaczy moduł zarządzania może ponownie uruchamiać (lub wysłać polecenie ponownego odczytu ustawień) moduł serwera multimedialnego (jeżeli ten nie obsługuje innych odtwarzaczy multimedialnych).

W sytuacji wykrycia błędnej współpracy na linii serwer i odtwarzacz multimedialny, po wprowadzeniu niezbędnych poprawek, testy dla wszystkich urządzeń były wykonywane ponownie.

Zmiana sposobu działania serwera (np. modyfikacja jednego z nagłówków odpowiedzi) umożliwiała współpracę z jednym odtwarzaczem, natomiast mogła utrudniać działania innych.

Specyfikacja DLNA określa dwa typy odtwarzaczy DLNA:

- odtwarzacze (stacjonarne),
- odtwarzacze mobilne.

Autor, na potrzeby sprawdzenia odtwarzania materiałów z serwera multimedialnego, stworzył następujący podział opisywanego oprogramowania do testowania:

- oprogramowanie dla komputerów klasy PC,
- oprogramowanie w szeroko opojętym sprzęcie „telewizyjnym”,
- oprogramowanie mobilne.

Powyższy podział powstał w sposób, który można nazwać naturalnym. Aplikacja, która współpracowała ze sprzętem telewizyjnym, we wczesnych fazach, nie działała poprawnie na pozostałych klasach. Sprzętem, na którym autor sprawdzał poprawność aplikacji, w początkowym etapie prac, były:

- telewizor marki Samsung,
- telewizor marki Phillips,
- odtwarzacz Blu-ray marki LG,
- smartfon Samsung Galaxy S2 (system operacyjny Android, oprogramowanie AllShare),
- smartfon Samsung Galaxy S (system operacyjny Android, oprogramowanie AllShare),
- smartfon Sony Ericsson (system operacyjny Android, oprogramowanie inne).

Najnowsze telewizory marki Samsung z oprogramowaniem *SMART HUB* posiadają oprogramowanie AllShare. Oprogramowanie o tej samej nazwie jest również dostępne dla komputerów PC z systemem operacyjnym Microsoft Windows oraz dla smartfonów z systemem operacyjny Android. Oprogramowanie, mimo tej samej nazwy, działa na każdym z urządzeń w inny sposób. Dokładne porównanie funkcji programów firmy Samsung znajduje się w rozdziale 5.

Poza opisanymi powyżej urządzeniami przetestowane zostały pojedyncze modele (dzięki udostępnieniu sprzętu przez jeden ze znanych marketów elektronicznych) telewizorów firm, takich jak Sony Bravia, Phillips oraz LG Smart TV.

W kolejnych podrozdziałach zostały przedstawione jedynie testy, które zostały dokładnie udokumentowane, tj. telewizora Samsung, odtwarzacz Blu-ray LG oraz telefonu Samsung Galaxy S2.

Zadaniami, (które na każdym z urządzeń były interpretowane w inny sposób), koniecznymi do bardzo dokładnego przetestowania były:

- przeglądanie zawartości folderów,
- odtwarzanie różnych kontenerów multimedialnych - filmów,
- odtwarzanie napisów,
- przewijanie filmów.

Serwer w trakcie testów miał już wcześniej zdefiniowane źródła odtwarzanych materiałów oraz udostępniał następujące pliki:

- pliki filmowe - avi, mkv, mp4,
- pliki audio - wav, mp3,
- pliki graficzne - jpg, png, miniatury filmów oraz zdjęć.

### 4.1.1 Test wyszukiwania urządzenia

Pierwszym elementem, który umożliwia współpracę urządzenia audio-wideo w protokołach UPnP AV oraz DLNA jest ich wyszukiwanie. W trakcie tego testu sprawdzane jest czy każdy odtwarzacz wykrył serwer multimedialny oraz wyświetlił jego prawidłową nazwę i miniaturę urządzenia.

### 4.1.2 Test przeglądania zawartości

Przed otwarciem materiału multimedialnego konieczne jest jego wyszukanie. Odbywa się to poprzez przeglądanie zawartości udostępnianych treści przez serwer multimedialny.

Sprawdzenie poprawności opisywanej operacji odbywało się na urządzeniach w kilku symulowanych sytuacjach. Pierwsza z nich, to uruchomienie serwera oraz przeglądanie zawartości na urządzeniu odtwarzającym. W tym momencie sprawdzane jest czy odtwarzacz wyświetla obecność wszystkich plików, które są współdzielone przez stworzone oprogramowanie. Test obejmuje również kontrolę poprawnego wyświetlania miniatur i prawidłowe wyświetlanie informacji o każdym pliku. Szczególnie ważne dla muzyki i treści wideo jest prawidłowe odczytanie długości materiału, gdyż istnienie tego parametru umożliwia w wielu urządzeniach przewijanie odtwarzanego pliku (w przypadku jego braku przewijanie jest zablokowane).

Przeglądanie zawartości umożliwia sprawdzenie czy serwer poprawnie przypisuje pliki do poszczególnych typów MIME oraz do typów odtwarzanych materiałów.

### 4.1.3 Test odtwarzania materiałów

W celu sprawdzenia materiałów autor przygotował kilkanaście plików muzycznych, graficznych oraz wideo. Każdy z innym rozszerzeniem, aby sprawdzenie odtwarzania było wykonane w jak najszerszym zakresie. Niniejszy test został wykonany na zasadzie porównywania serwera UPnP AV, z innym oprogramowaniem pełniącym rolę serwera UPnP AV (DLNA). Umożliwiło to określenie możliwości odtwarzaczy oraz w przypadku problemów z odtwarzaniem materiałów udostępnianych przez aplikację **UPNP**, wykrycie różnic, które powodują, iż inny serwer udostępnia materiały poprawnie.

Autor przygotował następujące pliki graficzne (rozszerzenie jest równoznaczne z kodowaniem pliku):

- plik z rozszerzeniem \*.JPEG,
- plik z rozszerzeniem \*.JPG,
- plik z rozszerzeniem \*.PNG,
- plik z rozszerzeniem \*.GIF,
- plik z rozszerzeniem \*.BMP.

Na potrzeby testów zostały przygotowane następujące pliki audio:

- plik z rozszerzeniem \*.MP3,
- plik z rozszerzeniem \*.WAV,
- plik z rozszerzeniem \*.WMA,
- plik z rozszerzeniem \*.AAC,
- plik z rozszerzeniem \*.FLAC,
- plik z rozszerzeniem \*.M4A.

Ostatnimi przygotowanymi plikami były pliki wideo, o następujących parametrach:

- plik MATROSKA, 4GB

- jedna ścieżka audio AC3,
- plik wideo **dkwijwefjw**,
- napisy wewnętrzne kontenera,
- napisy zewnętrzne (test odbył się z napisami i bez),
- plik MATROSKA 16GB,
  - dwie ścieżki audio: stereo oraz AC3,
  - plik wideo w formacie 3D SBS (ang. *3D Side by Side*, co oznacza film, który ma 2 obrazy obok siebie, konieczne do zinterpretowania przez odtwarzacz, celem odtworzenia go w wersji 3D),
  - napisy zewnętrzne.
- plik AVI
  - ścieżka audio mp3,
  - wideo mpeg...
- plik WMV
- plik MP4.

#### 4.1.4 Polecenia UPnP AV i DLNA

Do poprawnego przetestowania oprogramowania stworzonego na potrzeby niniejszej pracy magisterskiej konieczne było zapoznanie się z poleceniami protokołami UPnP AV / DLNA oraz sposobu w jaki oprogramowanie DLNA ów polecenia wykonuje.

Jedną z niewrażliwych operacji jest przeglądanie zawartości serwera multimedialnego. Usługa *Content Directory* [37] umożliwia klientom przeglądanie zawartości oraz opisuje sposób w jaki mogą ją odczytać. Usługa musi odpowiadać na kilka poleceń wysyłanych przez klienta. Są to takie właściwości jak: sposób przeszukiwania (polecenie *GetSearchCapabilities*), sposób sortowania (polecenie *GetSortCapabilities*), numer identyfikacyjny systemu (*GetSystemUpdateID*), przeglądanie (polecenie *Browse*). Istnieje kilka poleceń opcjonalnych (według specyfikacji UPnP AV oraz DLNA), jednakże jedno z poleceń (polecenie przeszukiwania *Search*) musi być udostępniane, gdyż niektóre z odtwarzaczy bez tego rodzaju usługi, nie wykrywają wówczas serwera w sposób poprawny.

## 4.2 Napotkane problemy

mkv trik

udostępnianie napisy - sec, attachments

## 4.3 Wyniki testów

Niniejszy podrozdział zawiera opisane wyniki poszczególnych testów, których procedura wykonania została opisana we wcześniejszym podrozdziale 4.1.

### 4.3.1 Test wyszukiwania urządzenia

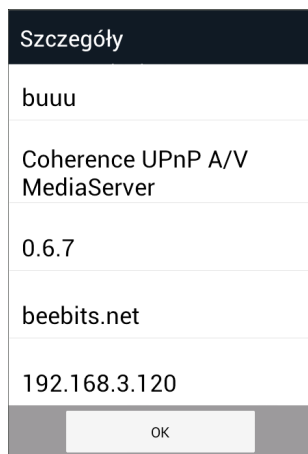
Po uruchomieniu napisanego oprogramowania, wysyła ono pakiety informujące działające już odtwarzacze o swoim istnieniu. Najnowsze oprogramowanie Samsung AllShare w wersji na każde urządzenie momentalnie wyświetlało istnienie serwera. W przypadku innych odtwarzaczy konieczne było wykorzystanie polecenia odświeżającego zgromadzoną listę serwerów. Wszystkie testowane



urządzenia poprawnie wykryły serwer oraz prawidłowo wyświetlały jego nazwę (również z wykorzystaniem polskich znaków). Miniatura na urządzeniach innych niż mobilne, nie była wyświetlana w ogóle (nie posiadają one funkcji wyświetlania miniatur). Aby miniatura serwera, „logo”, było poprawnie wyświetlane musi być w wymiarach 50 x 50 pikseli.

Wszystkie przetestowane odtwarzacze, bez dodatkowych problemów, wyświetlały szczegóły urządzenia, taką jak nazwę oprogramowania serwera (użytkownik może nadać swoją nazwę dla serwera udostępniającego, jednakże nazwa serwer **UPNP** pozostaje bez zmian). Zrzut ekranu z aplikacji mobilnej Samsung AllShare widoczny jest na rysunku 4.1.

**Rys. 4.1:** Karta Szczegóły serwera DLNA w odtwarzaczu DLNA



### 4.3.2 Test przeglądania zawartości

W większości przetestowanych modeli odtwarzaczy DLNA przeglądanie zawartości odbywa się w sposób podobny do wyboru plików w eksploratorach plików. Aby znaleźć pożądaną plik wybieramy odpowiedni katalog (który również służy za rodzaj materiału) np. Muzyka i wyszukujemy plik. Opisywana funkcjonalność została w zupełnie inny sposób zaimplementowana w najnowszych telewizorach firmy Samsung. W tym sprzęcie oprogramowanie wysyła żądanie od serwera o udostępnienie listy wszystkich filmów, zdjęć i muzyki i wyświetla wszystkie elementy w jednej liście zgodnie z kategoriami (Wideo, Muzyka, Obrazy), nie zważając na strukturę katalogów, którą udostępnia serwer.

Udostępniana zawartość od początkowych wersji aplikacji była w sposób poprawny wyświetlana na wszystkich sprawdzanych urządzeniach oraz oprogramowaniu dla komputerów klasy PC z systemami operacyjnymi Microsoft Windows oraz Ubuntu 11.10.

Po dodaniu zawartości pakiet, informujący o konieczności aktualizacji listy plików, został spreparowany poprawnie i wszystkie zasubskrybowane urządzenia go otrzymały. Jednakże niektóre programy klienckie UPnP AV (np. Samsung AllShare) nie reagują żądaniem odświeżenia listy plików, przez co lista plików nie aktualizuje się w sposób zautomatyzowany bez ingerencji użytkownika.

### 4.3.3 Test odtwarzania materiałów

Opisywany test skupił się na najważniejszej funkcji zaprojektowanej aplikacji, czyli odtwarzaniu treści multimedialnych. Poza główną funkcją, w trakcie testu sprawdzane zostały takie parametry jak:

1. możliwość przewijania materiału,
2. podgląd miniatur,

3. przegląd atrybutów udostępnianego pliku,
4. umożliwienie rozszerzania widoku (głównie działanie funkcji pełnego ekranu),
5. odtwarzanie napisów,
6. wybór ścieżki dźwiękowej,
7. zapamiętanie i wznowienie odtwarzania od ostatniej pozycji.

Ponadto dla wszystkich plików zostało sprawdzone obciążenie procesora urządzenia (na którym działa serwer) w trakcie odtwarzania,

Nie wszystkie odtwarzacze posiadają każdą z opisywanych funkcji. W celu zobrazowania wyników testów powstała tabela zestawiająca wszystkie funkcje dla każdego odtwarzacza. Kolumna A określa możliwość odtwarzania plików audio, O - plików graficznych, zaś W - plików wideo.

**Tabela 4.1:** Możliwości testowanych odtwarzaczy

| Nazwa odtwarzacza | A | O | W | 1 | 2 | 3 | 4  | 5  | 6  | 7 |
|-------------------|---|---|---|---|---|---|----|----|----|---|
| Samsung TV        | + | + | + | + | + | + | +  | +  | +  | + |
| Samsung I9000     | + | + | + | - | + | + | ?  | -  | -  | - |
| Samsung I9100     | + | + | + | + | + | + | ?  | -  | -  | - |
| AllShare PC       | + | + | + | - | + | + | +  | +  | +  | - |
| Phillips TV       | + | + | + | + | - | - | -  | -  | -  | - |
| LG BD390          | + | + | + | + | + | + | +  | +  | +  | - |
| XBOX              | + | + | + | + | - | + | +  | -  | -  | - |
| WMP               | + | + | + | + | - | + | -  | -  | -  | - |
| Rhytmbox          | + | - | - | + | - | + | nd | nd | nd | - |
| Totem             | + | - | + | + | + | + | +  | -  | -  | - |
| VLC               | + | - | + | + | + | + | +  | -  | -  | - |
|                   |   |   |   |   |   |   |    |    |    |   |

'+' - funkcja istnieje,

'-' - brak funkcji,

**? - it's complicated**

nd - nie dotyczy

Dokładne objaśnienie kwestii technicznych (w tym błędów) dotyczących odtwarzaczy została przedstawiona w rozdziale 5.

**wymyślić inaczej**

# Przegląd oprogramowania UPnP AV i DLNA

Oprogramowanie UPnP AV oraz DLNA umożliwia w bardzo prosty sposób odtwarzać materiały multimedialne w sieci lokalnej. Serwerów multimedialnych w sieci internetowej można znaleźć stosunkowo dużo. Natomiast odtwarzacze UPnP AV / DLNA są zazwyczaj wbudowane w większość standardowych odtwarzaczy dostarczanych wraz z systemem operacyjnym urządzenia.

W niniejszym rozdziale zostały zestawione, przetestowane przez autora, programy zgodne ze standardem UPnP AV lub DLNA. W kolejnych podrozdziałach wyróżnione zostały możliwości serwerów multimedialnych, a następnie odtwarzaczy multimedialnych. Drugi z podrozdziałów zawiera zestawienie zarówno oprogramowania dla komputerów typu PC, jak i odtwarzaczy sprawdzanych w sprzęcie telewizyjnym, tj. telewizorach i odtwarzaczach Blu-ray.

## 5.1 Przegląd oprogramowania serwerów UPnP AV i DLNA

Ilość aplikacji udostępniających multimedia jest bardzo dużo. Prawdopodobnie zbyt duża, aby zwykły użytkownik był w stanie podjąć prosty i zarazem dobry wybór. Ich ilość jest spowodowana jakością wykonania oraz nie spełnieniem warunku, którym jest odtwarzanie treści na urządzeniach różnych producentów. Warunkiem prawdopodobnie niemożliwym do spełnienia. W przypadku, gdy sieć domowa posiada odtwarzacze tylko jednego producenta (czyli teoretycznie powinny być zgodne), wystarczy odnaleźć serwer, który udostępni zawartość, a odtwarzacz będzie ją poprawnie prezentował. W kolejnych sekcjach niniejszego podrozdziału znajduje się krótki opis kilku wybranych serwerów multimedialnych.

### 5.1.1 Oprogramowanie Samsung AllShare

W przypadku oprogramowania firmy Samsung do udostępniania multimediiów wykorzystywane są dwa urządzenia. Komputery typu PC z systemem operacyjnym Windows (XP/Vista/7) oraz smartfony tej firmy.

Aplikacja udostępnia wszystkie rodzaje multimediiów ujęte w specyfikacji DLNA, czyli obrazy, muzykę oraz filmy. Ma możliwość kodowania materiałów w locie, udostępniania napisów w postaci SubRip to jest z rozszerzeniem \*.srt. Napisy muszą mieć taką samą nazwę plików jak film.

Program, w wersji dla systemu Microsoft Windows, umożliwia dodanie następujących plików wideo do listy odtwarzania: \*.3GP, \*.ASF, \*.AVI, \*.DIVX, \*.F4V, \*.FLV, \*.M1V, \*.M2TS, \*.M4V, \*.MKV, \*.MOV, \*.MP4, \*.MPEG, \*.MPG, \*.MPGV, \*.MPV, \*.MTS, \*.PS, \*.QT, \*.RM, \*.RMVB, \*.SVI, \*.TP, \*.TRP, \*.TS, \*.VOB, \*.VRO, \*.WMV. Pliki zdjęć mogą posiadać rozszerzenia: \*.BMP, \*.GIF, \*.JPEG, \*.JPG, \*.MPO, \*.PNG. Ostatnie pliki, audio, mogą zostać dodane spełniające

następujące parametry rozszerzenia: \*.3GA, \*.AAC, \*.AC3, \*.AU, \*.AMR, \*.APE, \*.DAT, \*.FLAC, \*.LPCM, \*.M4A, \*.MP2, \*.MP3, \*.MPA, \*.OGG, \*.PCM, \*.RA, \*.VOX, \*.WAV, \*.WMA. Przedstawione listy są bardzo długie i wydaje się, że spełniają wszystkie możliwe wymagania, jednakże pomimo iż tak dużą ilość rodzajów plików można dodać, to większość z nich jest udostępniana w taki sposób, że odtwarzacz nie umie ich zaprezentować użytkownikowi (pojawia się komunikat niepoprawny plik). W przypadku napisów autor potrafił uruchomić napisy jedynie do filmów z rozszerzeniem \*.AVI oraz \*.DIVX. W pozostałych przypadkach plik z napisami nie był wykryty przez odtwarzacz.

Oprogramowanie mobilne, dla systemu Android, nie ma możliwości przekazywania napisów dla urządzeń innych niż Samsung. Autor nie odnalazł listy z rodzajami plików do udostępnienia, lecz jest ona prawdopodobnie taka sama jak dla wersji PC serwera. Jednakże serwer mobilny, pliki które współdzieli, udostępnia w wyraźnie inny, lepszy sposób niż ma to miejsce w przypadku wersji komputerowej. Wszystkie pliki, które zostały dodane do udostępniania były poprawnie otwierane przez odtwarzacze DLNA, zarówno firmy Samsung, jak i innych producentów.

### 5.1.2 Nero MediaHome 4

Nero MediaHome 4 jest to płatne oprogramowanie (do testów wykorzystywana była wersja próbna) udostępniające multimedia, podobnie jak większość aplikacji, takie które są wyspecyfikowane w technologii DLNA. W pliku pomocy programu zostały przedstawione obsługiwane rozszerzenia. Dla plików audio są to: \*.AIFF, \*.AIF, \*.MP1, \*.MP2, \*.MP3, \*.MPA, \*.MP4, \*.AC3, \*.AAC, \*.M4A, \*.MP4, \*.WAV, \*.WAVE, \*.WMA, \*.OGG, OGM, \*.LPCM, \*.PCM, \*.L16. Dla plików obrazów: \*.BMP, \*.GIF, \*.JPEG, \*.JPG, \*.JPE, \*.PNG, \*.TIFF, \*.TIF, \*.WMF. Dla plików wideo: \*.3GP, \*.AVI, \*.DIVX, \*.ASF, \*.DV, \*.MPEG, \*.MPG, \*.MPE, \*.M1V, \*.DAT, \*.MPEG2, \*.MPG2, \*.M2V, \*.PVA, \*.MOD, \*.TOD, \*.VOB, \*.MP2P, \*.PS, \*.MP2T, \*.TS, MP4, \*.M4P, \*.MOV, \*.QT, \*.WMV, \*.DVR-MS, \*.FLV, \*.MP4. Wszystkie z wymienionych formatów mogą być transkodowane do innych formatów znajdujących się na liście. Lista plików dość długa, jednakże w trakcie sprawdzania programu autor wykrył fakt, że w rzeczywistości odtwarzanie materiału zależy od pakietu kodeków zainstalowanych w komputerze. Program prawdopodobnie dostarcza kodeki, które odtwarzają wspomniane wyżej formaty, jednakże jeżeli są zainstalowane kodeki, które odtwarzają inne formaty, wówczas aplikacja również je wspiera. Jest to zachowanie o tyle dziwne, że w przypadku braku transkodowania, dekodery formatu multimedialnego nie jest wykorzystywany.

W żadnym materiale dostarczonym wraz z Nero MediaHome 4 nie jest uwzględnione wsparcie dla napisów do filmów. Jest ono jednak zaimplementowane, a napisy muszą być w jednym z najpopularniejszych formatów, którymi w tym wypadku są SubRip lub MicroDVD.

Materiały udostępniane przez opisywany serwer poprawnie odtwarzają wszystkie przetestowane urządzenia. Serwer ten potrafi również strumieniować materiał telewizyjny z tunera DVB-T na żywo do innych urządzeń (funkcja nie została przetestowana). Można stwierdzić, że jest to kompletny odtwarzacz multimedialny, z jedną wadą, którą jest brak wieloplatformowości. Drugą cechą, która może odstraszyć przeciętnego użytkownika jest cena oprogramowania. Biorąc pod uwagę, że większość odtwarzaczy jest darmowych, jest to niewątpliwie wada.

### 5.1.3 MiniDLNA

Oprogramowanie MiniDLNA jest stworzony z myślą o platformie Linux, napisane w języku C++, udostępniane na publicznej licencji GPL ([40]). Zostało stworzone na potrzeby urządzenia *ReadyNAS* firmy *Netgear*, dzięki czemu ma bardzo małe wymagania sprzętowe. Jest bardzo popularnym serwerem multimedialnym, celem jego autora jest pełna współpraca ze wszystkimi urządzeniami UPnP AV oraz DLNA. MiniDLNA nie posiada dokumentacji technicznej, z tego powodu niemożliwe jest podanie możliwych do udostępniania plików. W trakcie testów, autor stwierdził, że wszystkie pliki muzyczne oraz obrazów były poprawnie odtwarzane na używanym sprzęcie. W przypadku filmów,

na każdym z urządzeń poprawnie udostępniane były pliki \*.avi. W przypadku innych rozszerzeń, takich jak \*.mp4 oraz \*.mkv, prezentacji treści multimedialnych możliwa była tylko w przypadku oprogramowania dla komputerów stacjonarnych. Odtwarzacze w urządzeniach RTV oraz mobilne nie potrafiły odtworzyć zawartości. MiniDLNA poprawnie przesyła napisy jedynie dla urządzeń firmy Samsung. Muszą one być w formacie SubRIP. W pozostałych urządzeniach autor nie potrafił odtworzyć filmu wraz z funkcją wyświetlania napisów. MiniDLNA konfigurowany jest poprzez modyfikację tekstowych plików modyfikacyjnych, nie posiada dedykowanego interfejsu do obsługi. Jest wbudowany w wiele dystrybucjach oprogramowania alternatywnego dla sprzętu sieciowego, wówczas posiada interfejs strony internetowej stworzony przez deweloperów.

### 5.1.4 Wild Media Server

*Wild Media Server* jest płatną aplikacją udostępniającą multimedia w sieci. Użytkownik rozpoczyna udostępnianie multimediiów od wyboru profilu urządzenia. Program ma wiele wbudowanych profili np. DLNA wersja 1.5, odtwarzacz blu-ray LG, telewizor LG, telewizor Samsung, telewizor Phillips i wiele innych. Użytkownik może stworzyć swój własny profil. Program umożliwia transkodowanie zawartości oraz wbudowywanie napisów do materiału w locie. Pomimo dostępności tych opcji odtwarzacze, które autor wykorzystywał nie potrafiły w prosty sposób wywołać transkodowania (operację tą udało się wykonać, jednakże sposób był bardzo nieprzyjazny dla użytkownika i nie zostanie tu opisany). Pliki \*.mkv, które sprawiają prawdopodobnie najwięcej problemów z odtworzeniem są przedstawiane przez serwer jako pliki \*.mpeg. Jednakże nawet ten unik (autor w swoim oprogramowaniu zastosował podobny) nie umożliwia odtwarzania materiałów na urządzeniach mobilnych. Ponadto serwer udostępnia nieprawidłowe informacje o udostępnianych plikach (błędny rozmiar oraz czas trwania materiału wideo), co może wpłynąć na obsługę (lub jej brak) funkcji przewijania.

### 5.1.5 PS3 Media Server

*PS3 Media Server* jest serwerem multimedialnym stworzonym z myślą o konsoli PlayStation 3 firmy Sony i zarówno zdaniem użytkowników, jak i autora programu, w pełni z nią współpracuje (autor niniejszej pracy nie testował aplikacji z konsolą). Serwer podobnie, jak pozostałe potrafi udostępniać obrazy, muzykę oraz filmy. Ma możliwość transkodowania materiałów audio i wideo. Umożliwia odtwarzanie napisów w postaci MicroDVD oraz SubRIP. Testowane przez autora pracy urządzenia nie potrafiły odtworzyć materiałów kodowanych przy pomocy H.264. Pozytywną cechą niniejszego serwera jest fakt, iż w takim przypadku automatycznie transkodował materiał do innego obsługiwanego, zarówno przez odtwarzacz, jak i serwer formatu. Urządzenia w przypadku innych odtwarzaczy (w tym aplikacji stworzonej na potrzeby niniejszej pracy magisterskiej) potrafiły odtwarzać materiał H.264.

### 5.1.6 Coherence

*Coherence* jest framework’iem wykorzystywanym do działania w sieci UPnP AV i DLNA. Funkcjonuje jako warstwa tworząca komunikaty UPnP. Może być również uruchomiony jako samodzielny program wykorzystujący wtyczki do zdefiniowania celu działania. Jedną z wtyczek jest udostępnianie multimediiów w sieci lokalnej, zgodnie z protokołem UPnP AV. Pożądane parametry spełnia wtyczka o nazwie *FSSStore*, która przeszukuje wyznaczone katalogi i udostępnia pliki multimedialne. Umożliwia ona udostępnianie najpopularniejsze formaty obrazów, audio oraz wideo. *Coherence* w wersji 0.6.2 ma możliwość dołączania napisów, w przypadku gdy mają one tę samą nazwę co plik wideo oraz posiadają rozszerzenie \*.srt. Do programu nie została napisana instrukcja obsługi, co w znacznym stopniu utrudnia jego konfigurację. Odbывается się ona poprzez stworzenie pliku tekstowego z odpowiednimi parametrami. Autorowi udało się uruchomić jedynie starą wersję programu *Coherence*. Dostarczała ona podstawowe funkcje tzn. wyszukiwała ona katalogi z multimediami,

a następnie mapowała je dla odtwarzacza. Nie posiada funkcji przydzielania plików do kategorii zgodnych z typem zawartości (np. Muzyka) oraz nie posiada specjalnie przygotowanych profili, aby współpracować z możliwie dużą ilością urządzeń. Wyjątek stanowi specjalne reguły dla konsoli do gier Microsoft Xbox 360, wobec którego autorzy wtyczki zrobili wyjątek i przygotowali specjalne komunikaty celem udostępniania treści dla wymienionego produktu.

## 5.2 Przegląd oprogramowania odtwarzaczy DLNA i UPnP AV

W sieci internetowej dostępnych jest bardzo wiele odtwarzaczy multimedialnych w sieciach lokalnych, zgodnych ze standardem DLNA i/lub UPnP AV. Odtwarzacze są zawarte jako programy w urządzeniach RTV, takich jak telewizory, odtwarzacze DVD, odtwarzacze Blu-ray i inne. Inną grupą odtwarzaczy są przeznaczone do instalacji w systemie operacyjnym. Spośród tych większość jest przeznaczona dla systemu operacyjnego Microsoft Windows, ewentualnie dla systemu MacOS X. Na szczęście dla użytkowników systemów rodziny Linuks, społeczność wolnego oprogramowania, jest na tyle rozwinięta, że i dla tych systemów nie jest większym problemem odnalezienie programu obsługującego opisywane standardy.

### 5.2.1 Oprogramowanie dla komputerów klasy PC

Odtwarzaczem DLNA, który autor może z całą pewnością polecić jest program XBMC Media Center, który został stworzony jako alternatywny interfejs graficzny konsoli Microsoft XBOX 360, a w chwili obecnej jest oprogramowaniem również dla systemów operacyjnych Windows, MacOS X oraz Linux. Jego interfejs jest bardzo przyjazny i starannie zaprojektowany, w taki sposób aby można było nim sterować każdym urządzeniem wskazującym np. pilotem telewizora. Jedynym mankamentem jaki odnalazł autor w niniejszym programie jest brak odtwarzania napisów w materiałach filmowych. Odtwarzacz bardzo dobrze prezentuje obrazy, muzykę oraz filmy.

Dla systemu Windows warto wymienić odtwarzacze Windows Media Player oraz Samsung AllShare, które potrafią wyszukać serwery DLNA lub UPnP AV i odtworzyć ich zawartość. W przypadku pierwszego niewątpliwą wadą jest brak obsługi napisów. Pozytywnymi cechami drugiego z wymienionych programów jest bardzo duża ilość materiałów możliwych do odtworzenia, obsługa napisów do filmów, niestety w przypadku komputera, na którym był testowany (uproszczona specyfikacja: procesor Intel Core 2 Duo 2.4 GHz, 8 GB Ram, dysk twardy 500 GB) obraz nie był płynnie odtwarzany. Zajmował przy tym około 80% procent zasobów procesora.

System operacyjny z rodziny Linux, kontrolowany za pomocą środowiska graficznego Gnome, może mieć zainstalowane programy do odtwarzania muzyki, wideo, obrazów. Po zainstalowaniu odpowiednich wtyczek do wyszukiwania muzyki w sieci lokalnej można wykorzystywać programy Rhythmbox lub Banshee, zaś do materiałów wideo program Totem. Podobnego oprogramowania specjalizującego się w wyświetlaniu obrazów oraz wyświetlającego pokaz slajdów z obrazów autor nie odnalazł.

### 5.2.2 Oprogramowanie w sprzęcie audio-wideo

W przypadku odtwarzaczy dla komputerów stacjonarnych użytkownik nie jest zmuszony do używania jednego rodzaju programu i może odnaleźć taki, który mu bardziej odpowiada. Natomiast w sytuacji odtwarzania materiałów na telewizorze użytkownik jest zobowiązany do wykorzystania programu, który dostarczył producent sprzętu. Autor zdecydował się zamieścić w niniejszym podrozdziale również krótką opinię na temat mobilnego odtwarzacza AllShare (dla smartfonów firmy Samsung), z tego względu, iż oprogramowanie to jest preinstalowane przez producenta.

Telewizor firmy Samsung, model UE46D6750WS, posiada oprogramowanie Samsung AllShare. Po włączeniu programu wyszukiwane są serwery multimedialnych. Po wybraniu serwera multimedialnych na pierwszy rzut oka widoczna jest różnica, pomiędzy innymi implementacjami tego typu

odtwarzaczy. Pozwala on na przeglądanie materiałów według trzech kategorii - Wideo, Muzyka, Obrazy. Widok ten nazywany jest widokiem prostym (ang. *Basic View*) i jeżeli serwer multimedialny nie udostępnia takiego widoku lub zwraca go błędnie, wówczas AllShare zgłasza błąd i uniemożliwia wyszukanie treści multimedialnej. Warto zwrócić uwagę, że odświeżanie automatyczne (które jest uwzględnione w specyfikacjach UPnP AV oraz DLNA) nie działa w taki sposób jak powinno, odtwarzacz nie potrafi przeglądać zawartości serwera bezpośrednio po jego dołączeniu bez ówczesnego odświeżenia listy serwerów. Testowany sprzęt posiada wbudowany sprzętowy dekodery H.264, a mimo to nie potrafił odtworzyć zawartości kontenera multimedialnego Matroska (\*.mkv). W przypadku pliku \*.mkv odtwarzanego z pamięci USB problem nie występował. Jak zostało wcześniej wspomniane autor w swoim programie zastosował obejście problemu, którym jest przedstawianie innego typu MIME dla tego rodzaju plików, to jest takiego jak dla plików \*.avi - „video/avi” lub „video/ms-video”.

Oprogramowanie Samsung AllShare dla urządzeń mobilnych wyświetla zawartość serwera poprzez przeglądanie katalogów (nie stosuje widoku prostego jak telewizory). Odświeżanie automatyczne listy serwerów zostało zaimplementowane poprawnie i działa zgodnie z oczekiwaniami. Program mobilny nie odtwarza poprawnie napisów do filmów (pobiera je, jednakże bez zgłoszenia żadnego błędu nie odtwarza). W filmach oraz obrazach, które były odtwarzane na urządzeniu występowały problem z dostosowaniem obrazu do wielkości ekranu. Podobnie jak w przypadku telewizorów konieczna jest zmiana typu MIME kontenera wideo.

Inna sytuacja występuje w przypadku odtwarzacza firmy LG. Nie trzeba stosować żadnych sztuczek, aby odtworzyć dowolny materiał. W przypadku formatu kontenera Matroska wyświetla jedynie komunikat o możliwych problemach z prezentacją filmu (autor jednak żadnych problemów w odtwarzaniu nie odnalazł). Odtwarzacz LG, dzięki temu komunikatowi odtwarza bardzo dużo formatów wideo oraz audio. Wadami oprogramowania, na które warto zwrócić uwagę to brak automatycznego odświeżania listy serwerów oraz zawartości multimedialnej serwera. Innymi mankamentami, które zostały wykryte to brak opisu obsługi napisów. Oznacza to, że napisy mogą być odtwarzane (również przy pomocy zawartości udostępnianej przez serwer multimedialny stworzony przez autora pracy), jednakże z niewiadomych względów niekiedy odtwarzacz nie potrafi wczytać napisów. Jedną z sytuacji jest, gdy film posiada dwie wersje napisów (np. napisy w dwóch językach, czyli dwa pliki z napisami).

Pliki MKV w opisywanym urządzeniu firmy LG są odtwarzane, dzięki certyfikatowi DivX HD, co powinno oznaczać pełne wsparcie dla rozdziałów, napisów oraz wielu ścieżek audio. Niestety wymagania certyfikatu nie zostały w pełni spełnione w odtwarzaniu filmów poprzez DLNA. Podobnie jak w wielu urządzeniach odtwarzacz Blu-ray nie potrafi poprawnie skalować wyświetlanych obrazów. Ma to jednak miejsce nie tylko w odtwarzaczu multimedialnym zgodnym z DLNA, ale również w „tradycyjnym” odtwarzaczu, który wczytuje pliki graficzne z nośnika danych np. pamięci USB lub płyty CD.

### 5.2.3 Obsługa standardu

Technologia DLNA została opracowana, aby umożliwić współdzielenie multimedialnych. Jej poprzedniczką jest UPnP AV, którą można określić jako bardziej liberalną. DLNA jest bardziej restrykcyjna, jednakże mimo bardziej ścisłego określenia współpracy różnice w implementacjach oraz funkcjonalność odtwarzaczy są dość znaczne. Utrudnia to prace zarówno programistom serwerów multimedialnych, a co istotniejsze zwykłym użytkownikom, którzy w prosty sposób nie mogą prezentować na każdym urządzeniu wszystkich rodzajów materiałów multimedialnych.

Różnice można podzielić na według grup, które mogą być współdzielone. W przypadku plików muzycznych, na podstawie przeprowadzonych badań, można stwierdzić, że ich obsługa jest najbardziej zaawansowana. Autor nie odnalazł żadnych negatywnych cech, które by utrudniały udostępnianie tego typu danych. Jako mankament, bardziej zauważalny w przypadku filmów, jest brak zagwarantowania w standardzie obsługi ścieżek dźwiękowych DTS oraz HD, które zapewniają

lepszą jakość dźwięku niż obsługiwane rodzaje ścieżek (np. AC3 lub MP3).[41] Można wyobrazić sobie sytuację, gdy użytkownik chce udostępnić audiobooka lub materiał muzyczny z kilkoma ścieżkami dźwiękowymi, jednakże w chwili obecnej funkcje te (zawarte w specyfikacji DLNA) nie są wykorzystywane, zarówno na poziomie odtwarzaczy, jak i serwerów.

Obrazy sprawiają wielu odtwarzaczom problem, gdy nie posiadają dostosowanej rozdzielczości do ekranu urządzenia. Odtwarzacz może źle dopasować plik graficzny do ekranu lub powiększyć go w taki sposób, że obraz jest przycięty (nawet gdy nie jest to konieczne). Spowodowane to jest z góry zdefiniowanymi przez DLNA profilami, które nie uwzględniają niektórych rozdzielczości obrazów. Problemy występują również w sposobie odtwarzania pokazu slajdów. Niektóre z odtwarzaczy nie buforują kolejnego zdjęcia, co powoduje iż pliki graficzne większych rozmiarów są dłużej ładowane na urządzeniu (użytkownik musi na nie oczekiwać). Innym problemem w przypadku obrazów jest ograniczenie tylko do dwóch formatów JPEG oraz PNG. Każdy inny, powołując się na standard DLNA, nie musi być obsługiwany. [41]

Standard DLNA nie wymusza na producentach obsługi wszystkich rodzajów kodowania materiałów wideo, obsługi napisów, obsługi wielu ścieżek dźwiękowych i innych. Niestety dla użytkowników, w specyfikacji zabrakło wielu w chwili obecnej wręcz niezbędnych formatów. Obsługiwane kontenery przez DLNA to MP4 oraz ASF. Są to kontenery multimedialne bardzo rzadko spotykane, posiadające wiele ograniczeń. Bardzo odczuwalnym w erze filmów HD jest maknament konetenera MP4, który przyjmuje rozdzielczość 720 x 576 pikseli jako maksymalną. W takim przypadku o film w rozdzielczości full HD, zgodnie ze standardem, nie może zostać przesłany. Podobna sytuacja występuje w coraz powszechniejszych filmach 3D. W standardzie pominięto bardzo popularne kontenery multimedialne, którymi są MKV oraz AVI. [41]

Z powyższych względów każdy z producentów tworzy własną wersję implementacyjną odtwarzacza DLNA, odbiegającą od standardu, z myślą o użytkownikach. Większość producentów nie tworzy oprogramowania serwerów multimedialnych na komputery klasy PC. Jeżeli jednak takie powstają, są one zgodne w pełni jedynie z urządzeniami tegoż producenta. Testy przedstawione w niniejszej pracy przeczą jednak teorii pełnej zgodności (przykład aplikacji Samsung AllShare) serwera i odtwarzacza tej samej firmy. Powodów takiej sytuacji, w artykułach oraz prasie (zarówno polskiej, jak i zagranicznej), nie sposób odnaleźć.



# Podsumowanie

# Bibliografia

- [1] UPnP Forum, UPnP Device Architecture 1.0, 2008.
- [2] Digital Living Network Alliance Home Networked Device Interoperability Guidelines Version: 1.0
- [3] Digital Living Network Alliance Home Networked Device Interoperability Guidelines Version: 1.0 Errata Version 2.0
- [4] <http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service-20110224.pdf>
- [5] [http://www.google.com/url?q=http%3A%2F%2Fwww.pervasive.jku.at%2FResearch%2FPublications%2F\\_Documents%2F2011\\_Online%2520UPnP%2520AV%2520Device%2520Database%2520for%2520Quick%2520and%2520Easy%2520Capability%2520Checking\\_Riener.pdf&sa=D&sntz=1&usg=AFQjCNEMLsCjydhx6Qy1L30d4wZ5HgUVA](http://www.google.com/url?q=http%3A%2F%2Fwww.pervasive.jku.at%2FResearch%2FPublications%2F_Documents%2F2011_Online%2520UPnP%2520AV%2520Device%2520Database%2520for%2520Quick%2520and%2520Easy%2520Capability%2520Checking_Riener.pdf&sa=D&sntz=1&usg=AFQjCNEMLsCjydhx6Qy1L30d4wZ5HgUVA)
- [6] [http://www.google.com/url?q=http%3A%2F%2Fwww.territorioscuola.com%2Fwikipedia%2Fen.wikipedia.php%3Ftitle%3DTalk%3ADigital\\_Living\\_Network\\_Alliance&sa=D&sntz=1&usg=AFQjCNFB10Qj3jNXfAqwV\\_NNQaRIBOk5zg](http://www.google.com/url?q=http%3A%2F%2Fwww.territorioscuola.com%2Fwikipedia%2Fen.wikipedia.php%3Ftitle%3DTalk%3ADigital_Living_Network_Alliance&sa=D&sntz=1&usg=AFQjCNFB10Qj3jNXfAqwV_NNQaRIBOk5zg)
- [7] [http://www.google.com/url?q=http%3A%2F%2Fsuperuser.com%2Fquestions%2F226636%2Fwhat-is-the-difference-between-dlna-and-upnp&sa=D&sntz=1&usg=AFQjCNHUjIpiufrZlgM\\_YLSNohaL355J\\_g](http://www.google.com/url?q=http%3A%2F%2Fsuperuser.com%2Fquestions%2F226636%2Fwhat-is-the-difference-between-dlna-and-upnp&sa=D&sntz=1&usg=AFQjCNHUjIpiufrZlgM_YLSNohaL355J_g)
- [8] [http://www.google.com/url?q=http%3A%2F%2Fjan.newmarch.name%2Fpublications%2Fjini\\_upnp.isng05.pdf&sa=D&sntz=1&usg=AFQjCNEZCczU2XK8hGsmHe7wU1mAbQpeKA](http://www.google.com/url?q=http%3A%2F%2Fjan.newmarch.name%2Fpublications%2Fjini_upnp.isng05.pdf&sa=D&sntz=1&usg=AFQjCNEZCczU2XK8hGsmHe7wU1mAbQpeKA)
- [9] Abul Ahsan Md. Mahmudul Haque, UPnP Networking: Architecture and Security Issues, 2007
- [10] UPnP Device Architecture V1.0 Annex A – IP Version 6 Support
- [11] UPnP Device Template
- [12] Networked Digital Media Standards A UPnP / DLNA Overview, Allegro Software Development Corporation, 2006.
- [13] RFC 2616 Hypertext Transfer Protocol – HTTP/1.1, Internet Engineering Task Force, 1999
- [14] SOAP Simple Object Access Protocol, W3C, 2007

- [15] Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C, 2008
- [16] Overview of UPnP AV Architecture, Intel Corporation, 2003
- [17] Abusing Universal Plug and Play, Armijn Hemel, 2008
- [18] Design and Implementation of home network systems using UPnP middleware for networked appliances, Dong-Sung Kim, Jae-Min Lee, Wook Hyun Kwon, Kwan You, 2002
- [19] UPnP Design by Example: A Software Developers's Guide to Universal Plug and Play, Michael Jeromino, Jack Weast, 2003
- [20] UPnP Forum UPnP AV Architecture:2, 2010
- [21] Network Media Content Aggregator for DLNA Server, Jan Kubový, 2011
- [22] ContentDirectory service:2 Service Template Version 1.01 – Document Version 1.00, 2006
- [23] DLNA Overview and Vision Whitepaper 2007
- [24] [http://brage.bibsys.no/hia/bitstream/URN:NBN:no-bibsys\\_brage\\_9695/1/andreas-haber-phone-based-upnp-control.pdf](http://brage.bibsys.no/hia/bitstream/URN:NBN:no-bibsys_brage_9695/1/andreas-haber-phone-based-upnp-control.pdf)
- [25] Wikipedia, [http://pl.wikipedia.org/wiki/Kontener\\_multimedialny](http://pl.wikipedia.org/wiki/Kontener_multimedialny), 2011
- [26] ISO/IEC 10918, JPEG
- [27] Cling Framework, <http://4thline.org/projects/cling>
- [28] Cyberlink for Java, <http://www.cybergarage.org/twiki/bin/view/Main/CyberLinkForJava>
- [29] MiniUPnP, <http://miniupnp.free.fr/>
- [30] MiniDLNA, <http://sourceforge.net/projects/minidlna/>
- [31] Coherence <http://coherence.beebits.net/wiki/WikiStart#CurrentState>
- [32] Licencja MIT, <http://www.opensource.org/licenses/mit-license.php>
- [33] Test Baz danych Open Source, SDJ Extra nr 17, PHP Starter Kit, <http://www.scribd.com/doc/51087611/2/Test-baz-danych-Open-Source>
- [34] Typy multimedialne MIME, <http://tools.ietf.org/html/rfc2046>
- [35] MPEG-4 ISO/IEC 14496-10
- [36] SQLAlchemy <http://www.sqlalchemy.org/>
- [37] ContentDirectory:1, Service Template Version 1.01, 2002, <http://upnp.org/specs/av/UPnP-av-ContentDirectory-v1-Service.pdf>
- [38] AllShare
- [39] JSON, <http://www.json.org/>
- [40] Licencja GPL, <http://www.gnu.org/licenses/gpl.html>
- [41] <http://www.digitalversus.com/tv-television/dlna-standard-real-mess-a971.html>