**Whaley DLNA&Cling协议用法详解**

1. **DLNA**
2. **文件功能介绍**

**1.0.1、HeliosReceiver：**

监听LMP消息ACTION\_DMP\_UPDATE\_REQUEST，启动HeliosCoreService

监听本机消息ACTION\_LOCAL\_DEVICE\_NAME，启动HeliosCoreService

监听网络状态CONNECTIVITY\_ACTION，启动PolicyService和CoreService

**1.0.2、HeliosCoreService：**

注册启动upnpService，注册监听器DefaultRegistryListener，通过MSCP监听网络内的DMS设备

处理intent消息ACTION\_DMP\_UPDATE\_REQUEST，提取DMS设备列表信息，通过广播ACTION\_DMP\_RESPONSE\_UPDATEALL发送到LMP。

处理intent消息ACTION\_LOCAL\_DEVICE\_NAME，更新DMR的FriendlyName，并广播到局域网

**1.0.3、HeliosPolicyService：**

ACTION\_UPDATE\_DEVICE，重新搜索DMS设备，注册DMR设备并广播到局域网。

**1.0.4、AndroidUpnpServiceImpl：**

提供UpnpServiceImpl实例upnpService，设置DMR监听接口，createRouter。

**1.0.5、HeliosDLNAApplication：**

维护全局资源：AndroidUpnpServiceImpl单实例、HeliosMediaRenderer单实例等。

* 1. **Application**

1.1.1、HeliosDLNAApplication，提供全局资源——UpnpService单例、MediaRenderer单例、ImageLoader、设备信息数据库。

1.1.2、启动HeliosPolicyService。

**1.2、AndroidUpnpServiceImpl**

android:name=org.fourthline.cling.android.AndroidUpnpServiceImpl

启动cling协议内部的service，协议层探测、监听等。

**1.3、HeliosReceiver**

HeliosReceiver监听网络状态（CONNECTIVITY\_ACTION），网络连接成功后启动HeliosPolicyService和HeliosCoreService两个service。

HeliosReceiver收到com.helios.action.dmp.request

和

com.helios.link.FRIENDLY\_NAME\_CHANGE广播后启动HeliosCoreService，intent内容是com.helios.action.dmp.request

和

com.helios.link.FRIENDLY\_NAME\_CHANGE，刷新DMS设备信息和修改DMR设备名。

**1.4、AndroidUpnpServiceImpl：**

文件功能是：UpnpService（UpnpServiceImpl实例）。构建时传入配置参数createConfiguration，设置一个握手的超时时间（30000）。重写了createRouter(创建一个AndroidRouter实例)和shutdown（退出时一些必要的release）

createRouter中直接构建AndroidRouter实例，AndroidRouter功能：后面章节的Cling部分详细描述。

定制DMR监听端口：2869

**1.5、HeliosCoreService：**

注册监听器registryListener到upnpService，通过MSCP监测局域网内DMS设备更新情况，在add和remove等接口中更新DeviceContainer（存储DMS设备列表信息）。

App LMP发送com.helios.action.dmp.request广播，HeliosReceiver收到此广播并启动HeliosCoreService，HeliosCoreService把deviceContainer内容填充到Intent中并广播（com.helios.action.dmp.list)给LMP，LMP监听到设备更新完成信息：com.helios.action.dmp.list后，提取DMS设备列表信息并显示。

registryListener.remoteDeviceAdded，注册需要监听哪些DMS设备。

收到com.helios.link.FRIENDLY\_NAME\_CHANGE intent后更新DMR friendly name并广播出去advertiseLocalDevices给到DMC。

**1.6、HeliosPolicyService：**

被HeliosDLNAApplication启动，网络连接成功后被HeliosReceiver启动。

**1.6.1、**清除cling注册表中的所有设备信息

**1.6.2、**在handler中启动cling去search特定的设备（InternetGatewayDevice和ContentDirectory），有设备更新时，利用callback机制，更新DeviceContainer数据，供LMP访问HeliosCoreService时提取DMS设备列表信息。

**1.6.3、**构建DMR设备并广播出去advertiseLocalDevices到DMC

1. **DMR**

**2.0、文件功能介绍**

HeliosPolicyService：创建注册DMR设备并广播通知DMC设备。

HeliosMediaRenderer：DMR设备类，创建一个LocalDevice，含有三个LocalService

HeliosMediaPlayerStore：创建并存储MagicMediaPlayer实例在一个map中

MagicMediaPlayer：播控类。

封装HeliosAudioPlayer/HeliosImageViewer/HeliosVideoPlayer，根据DMR的URI类型创建player实例。各个player对MediaPlayer或image封装。

HeliosConnectionManagerService：设置protocols category，指定支持的媒体格式和传输协议。

HeliosAVTransportService：DMR播放控制（seek/play等），封装MagicMediaPlayer

HeliosAudioRenderingControl：音量控制mute/setVolume，封装MagicMediaPlayer

**2.1、初始化：**

HeliosPolicyService在启动UpnpService的searchDMS之前，注册创建HeliosMediaRenderer实例：

LocalDevice renderer = mApplication.getMediaRenderer().getDevice()。

并广播到局域网：

getUpnpService().getRegistry().advertiseLocalDevices()

**2.2、HeliosMediaRenderer类**

**2.2.1、HeliosMediaPlayerStore(继承自HashMap)**

HeliosMediaRenderer类中构造函数首先创建一个HeliosMediaPlayerStore实例，HeliosMediaPlayerStore创建并存储一个MagicMediaPlayer实例，以Map形式存储，value是MagicMediaPlayer实例，key是UnsignedIntegerFourBytes。

Cling部分：

？？AVTransportLastChangeParser、RenderingControlLastChangeParser待分析？？

**2.2.2、MagicMediaPlayer**

DMR的各种控制指令通过此类启动各player activity并实现对其的封装。Player activity实现类：HeliosAudioPlayer/HeliosImageViewer/HeliosVideoPlayer。与DMP共用。

**三个LocalService：**

DMR设备（LocalDevice）嵌入三个service（LocalService）

**2.2.3、HeliosConnectionManagerService（ConnectionManagerService）**

初始化时传入protocols category，指名本DMR设备具体支持哪些媒体类型：audio、video及image的codec和封装格式、传输协议等。

**2.2.4、HeliosAVTransportService（AbstractAVTransportService）**

注册后Cling内部回调。实现并覆盖DMR播控（pause/seek/play等）方法。取得PlayerStore中存储的MagicMediaPlayer实例。HeliosAVTransportService直接封装MagicMediaPlayer中的播控方法（pause/seek/play等），MagicMediaPlayer直接封装MediaPlayer（由HeliosVideoPlayer等创建）的播控方法。

2.2.4.1、setAVTransportURI

解析Cling接收并处理过的DMC指令——URI，得到mimeType。调用MagicMediaPlayer.setURI，得到具体的item（ImageItem/AudioItem/VideoItem）。

2.2.4.2、Play（item）

根据item类型选择启动哪一个player的activity：

HeliosAudioPlayer/HeliosImageViewer/HeliosVideoPlayer等。

把item加入对应player的mPlayList列表中，播放时使用。

2.2.4.3、启动activity时填充intent extra参数：

putExtra("SOURCE", "DLNA");

putExtra("TYPE", "picture");

putExtra("DMR", true);

HeliosAudioPlayer等player类根据extra值区分DMP/DMR，若为DMR则调用getMediaRenderer().registPlayer(mMediaPlayer)，传入MediaPlayer的实例到HeliosMediaRenderer、HeliosAVTransportService、MagicMediaPlayer。

**2.2.5、HeliosAudioRenderingControl（AbstractAudioRenderingControl）**

注册后Cling内部回调。DMR音量控制（getVolume/ setVolume/ setMute），封装MagicMediaPlayer中的音量控制方法。

MagicMediaPlayer通过AudioManager控制电视音量。

(AudioManager) mApplication.getSystemService(Context.AUDIO\_SERVICE)。

**2.2.6、LocalDevice**

DMR设备实例，参数：ID、UDADeviceType（“MediaRenderer”）、内嵌三个LocalService

启动一个线程执行如下监听：

mAVTransport.fireLastChange()

mRenderingControl.fireLastChange()

1. **DMP**
2. **Cling**

主要有4个模块：Cling Core、Cling Support、Cling Workbench(app侧)、Cling MediaRender（app侧）

Whaley DLNA使用了Core和Support两个library。

Cling协议版本：2.0-alpha2。

官网：http://4thline.org/projects/cling/和DLNA spechttps://www.dlna.org/