**流媒体服务器详细设计**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 内容 | 作者 | 时间 |
| 1.0 | 流媒体方案初稿 | 李纯 | 2014/11/18 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1 引言 3](#_Toc404010963)

[2 需求 3](#_Toc404010964)

[2.1 应用场景拓扑结构图 3](#_Toc404010965)

[2.2 框架结构图 3](#_Toc404010966)

[2.3 P2P 通信交互图 4](#_Toc404010969)

[2.4 流媒体转发服务器交互图 5](#_Toc404010973)

[2.5 模块功能列表 5](#_Toc404010974)

[2.6 负载均衡策略 6](#_Toc404010975)

[3 数据库表定义 6](#_Toc404010976)

[3.4 数据库实体图 6](#_Toc404010977)

[3.5 mysql具体表描述 6](#_Toc404010978)

[3.6 Hbase 7](#_Toc404010979)

[4 功能交互图 8](#_Toc404010980)

[4.4 设备和接入服务器交互处理图 8](#_Toc404010981)

[4.2 RTSP会话请求流程 9](#_Toc404010984)

[4.3 设备SDK交互流程图 10](#_Toc404010988)

[4.7 P2P实时播放请求交互图 11](#_Toc404010991)

[4.5 流媒体转发实时播放请求交互图 12](#_Toc404010997)

[5 APP接入网关接口定义 12](#_Toc404010998)

[5.4 激活设备接口 12](#_Toc404010999)

[5.5 返回用户绑定的所有设备接口 13](#_Toc404011000)

[5.6 查看设备在线状态接口 14](#_Toc404011001)

[5.7 请求播放地址接口 14](#_Toc404011002)

[6 接入服务器功能和接口定义 14](#_Toc404011003)

[6.4 设备发送心跳接口 14](#_Toc404011004)

[7 设备SDK交互接口定义 15](#_Toc404011005)

[7.1 设备激活 15](#_Toc404011006)

[7.2 设备心跳信息 15](#_Toc404011009)

[7.3 设备受控响应 16](#_Toc404011013)

[8 分发服务器功能和接口定义 16](#_Toc404011014)

[8.1 播放接口 16](#_Toc404011015)

[8.2 暂停接口 16](#_Toc404011016)

[8.3 停止接口 17](#_Toc404011017)

[9 管理服务器功能和接口定义 17](#_Toc404011018)

[9.1 厂商发布产品功能 17](#_Toc404011019)

[9.2 在线用户管理功能 17](#_Toc404011020)

[9.3 厂商管理功能 17](#_Toc404011021)

# 引言

作为新一代[互联网](http://baike.baidu.com/view/6825.htm)应用的标志，[流媒体](http://baike.baidu.com/view/794.htm)技术在近几年得到了飞速的发展。而[流媒体](http://baike.baidu.com/view/794.htm)服务器又是流媒体应用的[核心](http://baike.baidu.com/subview/22680/8050316.htm)系统，是[运营商](http://baike.baidu.com/view/969221.htm)向用户提供视频服务的关键[平台](http://baike.baidu.com/view/58664.htm)。其主要功能是对[媒体](http://baike.baidu.com/view/7072.htm)内容进行[采集](http://baike.baidu.com/view/267501.htm)、[缓存](http://baike.baidu.com/view/907.htm)、[调度](http://baike.baidu.com/view/708911.htm)和[传输](http://baike.baidu.com/view/389471.htm)播放，[流媒体](http://baike.baidu.com/view/794.htm)应用系统的主要性能体现都取决于[媒体服务器](http://baike.baidu.com/view/1656360.htm)的[性能](http://baike.baidu.com/view/651600.htm)和[服务质量](http://baike.baidu.com/view/522662.htm)。因此，[流媒体服务器](http://baike.baidu.com/view/1295757.htm)是[流媒体](http://baike.baidu.com/view/794.htm)应用系统的基础，也是最主要的组成部分。作为智能硬件服务端。特别是需要接入视频设备，建设自己的流媒体服务器是刻不容缓的。

# 需求

## 应用场景拓扑结构图



## 框架结构图





## P2P 通信交互图



以下为NAT 类型点对点通信示意表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NAT | full cone | restricted cone | port restricted cone | symmetric nat |
| full cone | Direct | Reversal | Reversal | Reversal |
| restricted cone | Direct | Hole Punch | Hole Punch | Hole Punch |
| port restricted cone | Direct | Hole Punch | Hole Punch | Relaying |
| symmetric nat | Direct | Hole Punch | Relaying | Relaying |

注释：上表左边栏为请求发起者，右上边栏位接收者



## 流媒体转发服务器交互图



## 模块功能列表

1. 分发服务器  
   支持标准RTSP协议，支持H264/AAC RTP封包，支持分布式部署，支持用户链路数控制功能，将输入和输出通过内外网卡分离保障输入和分发的独立性和可靠性，
2. P2P服务器  
   只是点对点的实时穿透在UDP基础上进行数据传输，并不包含P2P其他业务。需要在公网上部署stun服务器做为一个协议协商服务，和smart后台进行通信获取当前任务，监测服务是否正常运行，有smart完成负载均衡并分派stun服务器地址和端口给设备和app
3. 接入服务器  
   支持分发服务器的实际地址分配，支持设备长连接，支持给设备下发控制指令功能
4. 管理服务器  
   支持厂商和用户管理，支持查看历史记录。

## 负载均衡策略

* + 1. 接入服务器

按照智能家居框架设计，接入服务器匹配VIP规则，以轮询条件做为选择接入服务器的原则

* + 1. 分发服务器

所有分发服务器启动接入zookeeper, 并实时写入当前连接数。zookeeper自动管理接入的分发服务器，smart根据分发服务实时连接数量来选择最少的分发服务器做为当前转发

# 数据库表定义

## 数据库实体图



## mysql具体表描述

1. 产品表（增加类别5来区分是否是视频设备）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 是否为空 | 备注 |
| id | bigint | 20 | 否 | 主键产品ＩＤ |
| name | varchar | 64 | 否 | 产品名称 |
| secret | varchar | 64 | 否 | 产品密码 |
| description | text |  |  | 描述 |
| private | tinyint | 1 | 否 | 是否公开，0不公开，1公开 |
| pro\_type | int | 10 | 否 |  |
| user\_pin | varchar | 64 | 否 | 创建人,京东pin |
| release\_state | tinyint | 1 | 否 | 发布状态，0未发布，1已发布 |
| img\_url | varchar | 255 |  |  |
| gmt\_create | timestamp |  | 否 |  |
| gmt\_modify | timestamp |  | 否 |  |
| public\_flag | int | 10 | 否 | 默认为私人设备-0 公开设备-1 |
| config\_type | int | 10 | 否 |  |
| pro\_uuid | varchar | 200 |  |  |
| callback\_url | varchar | 200 |  |  |
| callback\_types | varchar | 100 | 否 |  |
| share\_flag | int | 10 |  |  |

1. 设备表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 是否为空 | 备注 |
| id | bigint | 20 | 否 | 主键feed\_id |
| device\_id | varchar | 64 | 否 | 设备ID,厂商设备定义，如mac,ismi |
| product\_id | bigint | 11 | 否 | 产品ID,关联产品表 |
| access\_key | varchar | 64 | 否 | 访问KEY |
| pro\_type | int | 11 | 否 | 类别 1能源管理 2环境检测 3安防设备 4智能电器　**5　视频设备** |
| private | tinyint | 1 | 否 | 是否公开，0不公开，1公开 |
| gmt\_create | timestamp |  | 否 | 创建时间 |
| gmt\_modify | timestamp |  | 否 | 最后更新时间 |
| public\_flag | int | 10 | 否 |  |
| device\_version | float | 10,2 | 否 |  |

1. 用户对应设备表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 是否为空 | 备注 |
| id | bigint | 20 |  | 主键id |
| user\_pin | varchar | 64 |  | 京东pin |
| feed\_id | bigint | 20 |  | 关联设备表ID |
| device\_name | varchar | 64 |  | 我的设备名称 |
| gmt\_create | timestamp |  |  |  |
| own\_flag | int | 10 |  |  |

## Hbase

使用两张表存储：jdtsdb-uid与jdtsdb。

jdtsdb用于存储元数据：tagk、tagv的值与其编码的应用。如下图所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| rowkey | column family:name | | column family:id | |
| tagk | tagv | tagk | tagv |
| [0,0,0,1] | stream\_id | temperature |  |  |
| [0,0,2,1] |  | switch |  |  |
| stream\_id |  |  | [0,0,0,1] |  |
| temperature |  |  |  | [0,0,0,1] |
| switch |  |  |  | [0,0,2,1] |

表1 jdtsdb-uid表

Jdtsdb用于存储真正的时间序列数据（datapoint）。

rowkey设计

metric编码（8字节）+ 时间戳（4字节）+ stream\_id编码（1字节 对应播放，暂停，下载）+tagv编码（4字节），总共17字节组成。

时间戳计算方式：timestamp = int.Max - (now(标准unix时间)– start(2010.1.1 00:00:00))

时间戳优点：a、距离现在越近，timestamp越小，方便查询current值；

b、减去start(2010.1.1 00:00:00))，可以延长以后的存储时间（可以使用63年）

存储表设计

如下表所示：

|  |  |
| --- | --- |
| row key | column family:t |
| v |
| xxx | 36 |

表2：jdtsdb表

Column qualifier：使用2个byte存储，第一个byte表示此column qualifier对应的cell值的类型（0：int，1：float，2：string），后面1个字节为列名v。

# 功能交互图

## 设备和接入服务器交互处理图





## RTSP会话请求流程





## 设备SDK交互流程图





## P2P实时播放请求交互图

上述P2P通道打通后连接通道为UDP传输通道。

设备发送SDP信息到客户端，格式为，

{

"cmd" : 50201,

"sdp" : ”xxxxxxxxxxxxxx”

}

客户端接收SDP后，请求发送流数据格式为，

{

"cmd" : 50202

}

视频流关闭需要发送到服务器，同时客户端需要发送关闭命令道设备，格式为，

{

"cmd" : 50203

}



## 流媒体转发服务器交互图（12.3号加）

（特别说明，手机APP请求播放的验证，上图中的处理是跟在URL后面以参数的形式进行传输，这样的好处就是APP端对于验证来说不需要做什么处理，对于服务器端，还需要验证这种方式是否可行。如果不可行，需要APP端把TOKEN放入到RTSP协议的中来进行传输）

上述流程图是按照先走P2P方式，P2P连接失败后按照上图进行交互。设备端按照darwin系统的连接方式进行SDP的传输和流的发送。

# APP接入网关接口定义

## 激活设备接口

|  |  |
| --- | --- |
| 请求方法 | Post |
| 请求URL | <http://gw.smart.jd.com/f/service/activate> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) | {  "product\_uuid": "MCQEVS",  "device\_id": “3333”,  "feed\_id":0  } |
| 返回JSON | {  "status": 0,  "error": null,  "result": {  "server\_ip": [  "10.28.171.130:9101"  ],  "server\_time": "2014-09-09T09:22:13+0800",  "access\_key": "bb0dbd6f869520b22323a406f8e05819",  "feed\_id": 141022573368801980  }  } |

## 获取设备快照信息

|  |  |
| --- | --- |
| 请求方法 | post |
| 请求URL | <http://gw.smart.jd.com/f/service/getStreams> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) | {  "feed\_id": 214739  } |
| 返回JSON | {"status":0,  "error":null,  "result":"{  \"device\":{  \"access\_key\":\"3dc13543c48163b5d3fa8d7a582d545e\",  \"active\_time\":\"2014-06-19 20:01:27\",  \"device\_id\":\"12345678\",  \"device\_name\":\"jk3\_195403\",  \"feed\_id\":214733,  \"status\":\"0\"  },  \"product\":{  \"p\_description\":\"jk3\",  \"p\_img\_url\":\"\",  \"pro\_type\":-2,  \"product\_id\":12835,  \"product\_name\":\"jk3\",  \"type\_desc\":\"健康类\",  \"type\_name\":\"健康类\"},  \"streams\":[{  \"at\":\"\",  \"current\_value\":\"\",  \"master\_flag\":1,  \"stream\_id\":\"sport\",  \"stream\_name\":\"运动数据\",  \"stream\_type\":1,  \"units\":\"\",  \"value\_des\":\"\"  }  ]  }"  } |

## 获取设备列表

|  |  |
| --- | --- |
| 请求方法 | get |
| 请求URL | <http://gw.smart.jd.com/f/service/getDeviceList> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) |  |
| 返回JSON |  |

## 音视频请求(暂时保留不做)

|  |  |
| --- | --- |
| 请求方法 | post |
| 请求URL | <https://gw.jd.com/f/service/realplay> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) | {  "feed\_id":214739  } |
| 返回JSON | {  "status": 0,  "error": null,  "result": null  } |

## 获取stun服务器

|  |  |
| --- | --- |
| 请求方法 | get |
| 请求URL | <https://gw.jd.com/f/service/getStunServer> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) |  |
| 返回JSON | {  "status": 0,  "error": null,  "result": {“stunip”:”211.23.4.5”, “stunport”:”3478”}  } |

## Stun交互信息上传

|  |  |
| --- | --- |
| 请求方法 | post |
| 请求URL | <https://gw.jd.com/f/service/stunResult> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) | {  "mapaddr": “213.214.21.27”,  "mapport": 6300,  “localaddr”: “192.168.1.34”,  “localport”: “6754”,  "natype": 1,  “netype”: “udp”  } |
| 返回JSON | {  "status": 0,  "error": null,  "result": null  } |

## 获取是否支持p2p并获取设备stun信息

|  |  |
| --- | --- |
| 请求方法 | get |
| 请求URL | <https://gw.jd.com/f/service/stunResult> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) | {  "feed\_id":214739  } |
| 返回JSON | {  "status": 0,  "error": null,  "result": {“p2psupoort”:0, "mapaddr": “213.214.21.27”, "mapport": 6300, “localaddr”: “192.168.1.34”, “localport”: “6754”, "natype": 1, “netype”: “udp”}  }  p2psupport: 0表示支持p2p, -1表示不支持p2p |

## 获取播放地址接口

|  |  |
| --- | --- |
| 请求方法 | get |
| 请求URL | <https://gw.jd.com/f/service/rtsp> |
| Cookie | \_jsmart\_=U6LN5GTF2Y6CRAM7DKME7KEQFREO5JUGC5VZC4I; Path=/ |
| 请求参数  (JSON) | {  "feed\_id":214739  } |
| 返回JSON | {  "status": 0,  "error": null,  "result": {"url":"rtsp://ip:port/realplay/FEED\_ID“}  } |

# 设备SDK和接入服务器交互接口定义

## 设备心跳信息

|  |  |
| --- | --- |
| 请求方式 | TCP |
| 请求URL |  |
| 请求头 |  |
| 请求参数  (JSON) | {  "code": 101,  "device": {  "feed\_id": "A5FFE67YU",  "access\_key": "xxxxxxx"  }  } |
| 返回状态码 |  |
| 返回 | {  "code" : 1001,  "result" : 0,  "time" : "yyyy-MM-dd hh:mm:ss"  } |



## 设备受控响应

|  |  |
| --- | --- |
| 请求方式 | TCP |
| 请求URL |  |
| 请求头 |  |
| 请求参数  (JSON) | {  "code": 1002,  "control": "xxxxxxxxx",//透传的控制命令  "feed\_id":xxxxx,  "attribute":xxxx//本次会话属性(josn字符串，设备不需要解析)，设备响应时，需要原样返回  } |
| 返回状态码 |  |
| 返回 | {  "code" : 102,  "result" : 0,  "control\_resp" : "xxxxxxx" ,//设备响应服务器  "attribute":xxxx,//服务器发送控制命令时，传送的本次会话属性  "device": {  "feed\_id": "A5FFE67YU",  "access\_key": "xxxxxxx"  }  } |

注：设备视频请求格式同上，code码为1009

设备准备rtsp服务格式同上，code码为1015

## Stun服务器分配

|  |  |
| --- | --- |
| 请求方式 | TCP |
| 请求URL |  |
| 请求头 |  |
| 请求参数  (JSON) | {  "code": 1010,  "control": "xxxxxxxxx",//透传的控制命令  "feed\_id":xxxxx,  "attribute":xxxxx,//本次会话属性(josn字符串，设备不需要解析)，设备响应时，需要原样返回  “stunip”:”211.23.4.5”,  “stunport”:”3478”  } |
| 返回状态码 |  |
| 返回 | {  "code" : 1010,  "result" : 0,  "control\_resp" : "xxxxxxx" ,//设备响应服务器  "attribute":xxxxx,//服务器发送控制命令时，传送的本次会话属性  "device": {  "feed\_id": "A5FFE67YU",  "access\_key": "xxxxxxx"  }  } |

## Stun交互信息上传服务器

|  |  |
| --- | --- |
| 请求方式 | TCP |
| 请求URL |  |
| 请求头 |  |
| 请求参数  (JSON) | {  "code": 1011,  "device": {  "feed\_id": "A5FFE67YU",  "access\_key": "xxxxxxx"  },  "mapaddr": “213.214.21.27”,  "mapport": 6300,  “localaddr”: “192.168.1.34”,  “localport”: “6754”,  "natype": 1,  “netype”: “udp”  } |
| 返回状态码 |  |
| 返回 | {  "code" : 1011,  "result" : 0,  "time" : "yyyy-MM-dd hh:mm:ss"  } |

## p2p是否支持消息及推送APP stun信息

|  |  |
| --- | --- |
| 请求方式 | TCP |
| 请求URL |  |
| 请求头 |  |
| 请求参数  (JSON) | {  "code": 1012,  "control": "xxxxxxxxx",//透传的控制命令  "feed\_id":xxxxx,  "attribute":xxxxx,//本次会话属性(josn字符串，设备不需要解析)，设备响应时，需要原样返回  “p2presult”: 0 //0表示支持p2p, -1表示不支持p2p  "mapaddr": “213.214.21.27”,  "mapport": 6300,  “localaddr”: “192.168.1.34”,  “localport”: “6754”,  "natype": 1,  “netype”: “udp”  } |
| 返回状态码 |  |
| 返回 | {  "code" : 1012,  "result" : 0,  "control\_resp" : "xxxxxxx" ,//设备响应服务器  "attribute":xxxxx,//服务器发送控制命令时，传送的本次会话属性  "device": {  "feed\_id": "A5FFE67YU",  "access\_key": "xxxxxxx"  }  } |

## p2p通道是否打通消息

|  |  |
| --- | --- |
| 请求方式 | TCP |
| 请求URL |  |
| 请求头 |  |
| 请求参数  (JSON) | {  "code": 1014,  "device": {  "feed\_id": "A5FFE67YU",  "access\_key": "xxxxxxx"  },  "p2pconnect": 0 // 0表示穿透成功,-1表示穿透失败  } |
| 返回状态码 |  |
| 返回 | {  "code" : 1014,  "result" : 0,  "time" : "yyyy-MM-dd hh:mm:ss"  } |

## 转发服务器地址端口及token推送到设备

|  |  |
| --- | --- |
| 请求方式 | TCP |
| 请求URL |  |
| 请求头 |  |
| 请求参数  (JSON) | {  "code": 1015,  "device": {  "feed\_id": "A5FFE67YU",  "access\_key": "xxxxxxx"  },  "relayIP": “xxxx”,  “port”:5666,  “token”: “xxxxxxx”  } |
| 返回状态码 |  |
| 返回 | {  "code" : 1015,  "result" : 0,  "time" : "yyyy-MM-dd hh:mm:ss"  } |

# 分发服务器功能和接口定义

## 播放接口

|  |  |
| --- | --- |
| 请求方式 | rtsp |
| 请求URL | Rtsp://x.x.x.x:port/realplay/{FEED\_ID} |
| 请求头 |  |
| 请求参数  (JSON) | Play |
| 返回状态码 | 按照rtsp协议规范执行 |
| 返回 | 视频流 |

## 暂停接口

|  |  |
| --- | --- |
| 请求方式 | Rtsp |
| 请求URL | Rtsp://x.x.x.x:port/realplay/{FEED\_ID} |
| 请求头 |  |
| 请求参数  (JSON) | Pause |
| 返回状态码 | 按照rtsp协议规范执行 |
| 返回 |  |

## 停止接口

|  |  |
| --- | --- |
| 请求方式 | Rtsp |
| 请求URL | Rtsp://x.x.x.x:port/realplay/{FEED\_ID} |
| 请求头 |  |
| 请求参数  (JSON) | Terminated |
| 返回状态码 | 按照rtsp协议规范执行 |
| 返回 |  |

# 管理服务器功能和接口定义

## 厂商发布产品功能

可以和现在smart合用，用于厂商发布产品和审核

## 在线用户管理功能

查看用户和相应设备。在线用户和正在使用的设备

## 厂商管理功能

可以和现在smart合用，用于厂商和厂商产品的管理