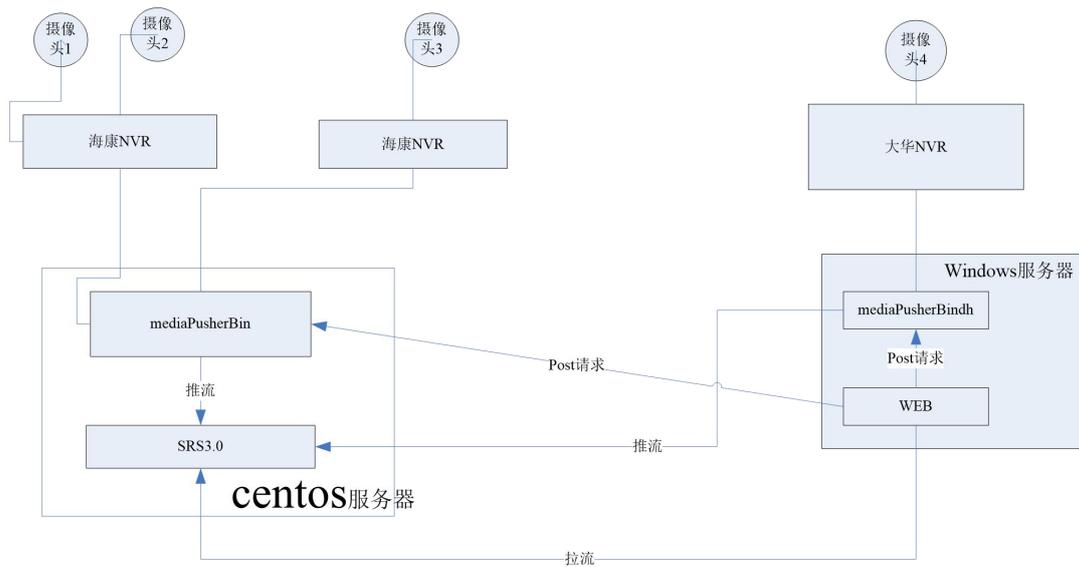


推流服务器

一、框架部署图

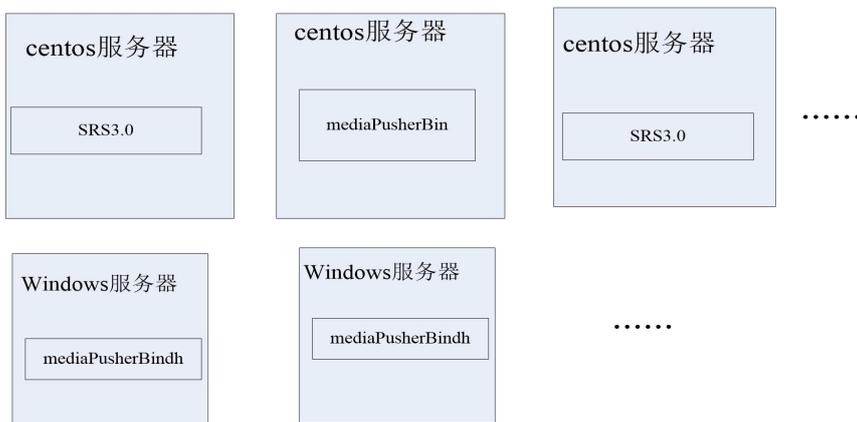
以新余视频为例部署如下图：



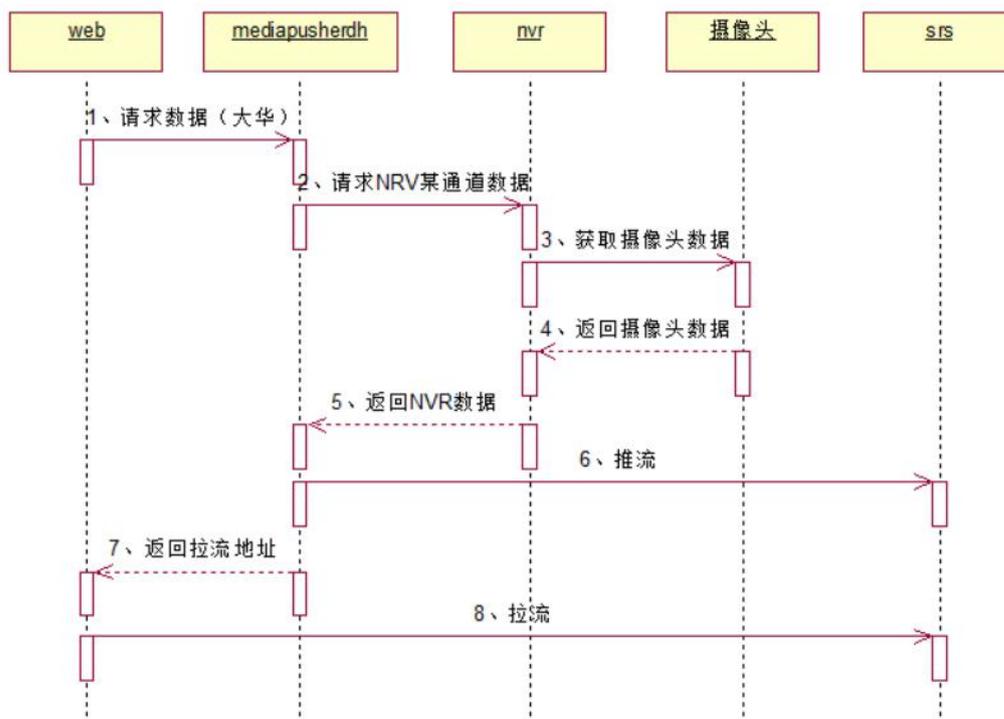
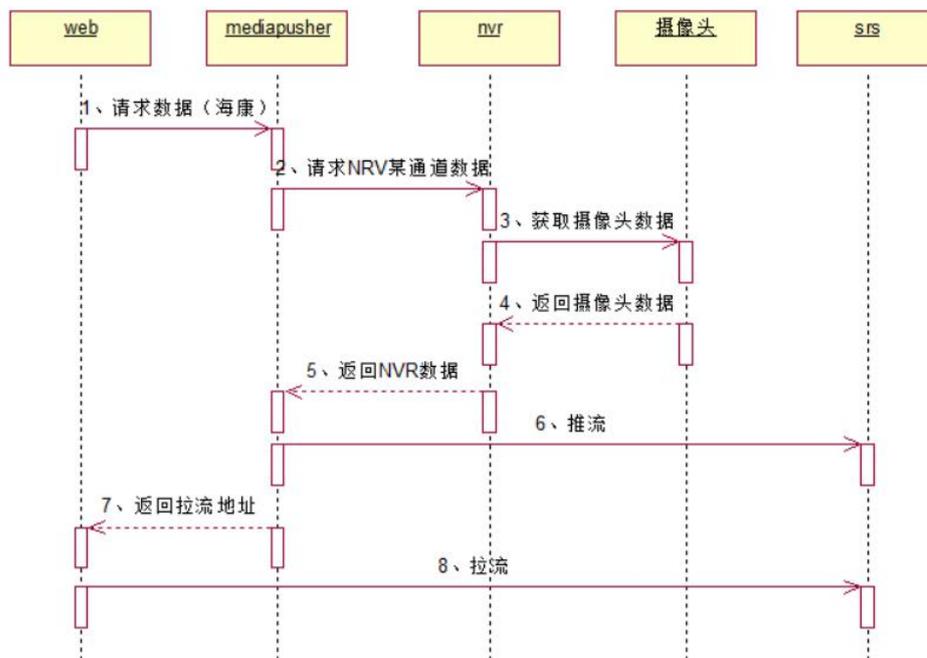
- 1、mediaPusherBin：原海康和大华的推流服务程序，基于 centos 系统运行。
- 2、mediaPusherBindh：大华推流服务程序，基于 windows 系统运行。
- 3、SRS3.0：是一个简单高效的实时视频服务器，支持 RTMP/WebRTC/HLS/HTTP-FLV/SRT/GB28181/RTSP。

二、扩展部署

如果一台 centos 或一台 windows 不能满足使用需求，可以增加推流服务器。



三、时序图



四、推流服务器框架



API 接口：用于监听处理 WEB 端各项业务的 http 请求

任务调度：用于管理多个任务

大华/海康 sdk：用于通过 SDK 与 NVR 数据进行数据交换

推流：用于将交换的数据推送到 SRS 服务。

1、海康推流

- 1.1 基于 centos 平台.
- 1.2 采用开源 restbed 作为 web api 库提供服务.
- 1.3 集成海康的取流 sdk
- 1.4 实现 PS 流向 RTMP 流封装，使用 ffmpeg 推流
- 1.5 RTMP Server 采用开源 simple rtmp server3.0

2、大华推流

- 2.1 基于 windows 平台.
- 2.2 采用开源 JQHttpServer 作为 web api 库提供服务.
- 2.3 集成大华的取流 sdk
- 2.4 实现 rtsp 流向 RTMP 流封装，使用 ffmpeg 推流 4.3.1
- 2.5 RTMP Server 采用开源 Simple Realtime Server3.0

五、Web API 接口说明

```
#define ERR_NO_ERROR 200
#define ERR_PARAMETER 600 //输入参数错误
#define ERR_ANA_STREAM 601 //帧分时失败
#define ERR_CON_RTMPSEVER 602 //连接 rtmp server 错误
#define ERR_PLAY_REAL 603 //实时播放失败
#define ERR_PLAY_BACK 604 //回放播放失败
#define ERR_NO_PUSHER_FOUND 605 //未找到播放项
#define ERR_NVR_LOGIN 606 //NVR 登录失败
#define ERR_NO_FILE_FOUND 607 //未找到视频文件
#define ERR_PLAY_CONTROL 608 //播放控制失败
#define ERR_PTZ_CONTROL 609 //云台控制失败
```

接口说明	实时流推送接口
接口地址	http://ip:port /REAL_STREAM
接口方法	POST
请求 Header	Content-type: "application/json"
请求 Body	{ "nvr_category":0, "nvr_addr":"10.8.30.109", "nvr_port":8000, "nvr_user":"admin", "nvr_pwd":"123456", "nvr_channel":30 }
响应 Body	{ "status_code":200, "status_message":"OK", "rtmp_url": "rtmp://10.8.30.109:8554/142595818.sdp", "play_handle":12681 }
说明	用于监控点的实时视频推流 nvr_category: NVR 厂家, 0: 海康 1: 大华 nvr_channel: 监控点所在通道, 可能不是从 0 号起, nvr 配置决定。 play_handle: 本次播放的句柄, 调用方应该记录此句柄。 后续调用停止播放和心跳接口时需要传递此值。

接口说明	历史流推送接口
接口地址	http://ip:port/BACK_STREAM
接口方法	POST
请求 Header	Content-type: "application/json"
请求 Body	<pre>{ "nvr_category":0, "nvr_addr":"10.8.30.109", "nvr_port":8000, "nvr_user":"admin", "nvr_pwd":"123456", "nvr_channel":30, "startTime" : "2016-07-20 12:00:00", "endTime" : "2016-07-20 13:00:00" }</pre>
响应 Body	<pre>{ "status_code":200, "status_message":"OK", "rtmp_url": "rtmp://10.8.30.109:8554/142595818.sdp", "play_handle":13579 "start_time" : "2016-07-20 12:00:00", "stop_time" : "2016-07-20 13:30:00", "record_files":[{ "file_name": "ch0010_01000001613000000", "file_size": 900417012, "start_time": "2016-07-20 11:50:00", "stop_time": "2016-07-20 12:50:00" }, { "file_name": "ch0010_01000001613000100", "file_size": 161928736, "start_time": "2016-07-20 12:50:00", "stop_time": "2016-07-20 13:30:00" }] }</pre>
说明	<p>用于监控点的回放视频推流</p> <p>nvr_category: NVR 厂家, 0: 海康 1: 大华</p> <p>nvr_channel: 监控点所在通道, 可能不是从 0 号起, nvr 配置决定。</p> <p>play_handle: 本次播放的句柄, 调用方应该记录此句柄。 后续调用停止播放和心跳接口时需要传递此值。</p> <p>Start_time: 本次播放实际的开始时间</p> <p>Stop_time: 本次播放实际的停止时间</p> <p>Record_files: 录像文件数组</p>

接口说明	流推送时心跳接口
接口地址	http://ip:port /STREAM/play_handle /BEATS
接口方法	GET
请求 Header	
请求 Body	
响应 Body	{ "status_code":200, "status_message": "OK" }
说明	用于通知推流端仍在播放, 在播放期间每 20 秒调用一次。 前端调用推流接口成功后(包含实时和回放), 在结束播放前, 需要按频率调用此接口保持推流动作, 否则推流端可能会终止推流过程.

接口说明	播放流定位接口(历史流)
接口地址	http://ip:port / STREAM/play_handle/POSITION
接口方法	POST
请求 Header	Content-type: "application/json"
请求 Body	{ "pos_time": "2016-07-20 12:35:00" }
响应 Body	{ "status_code":200, "status_message": "OK", }
说明	用于快速定位播放流到指定时间位置 pos_time: 定位时间点, 必须在请求历史流的时间范围内。

接口说明	实时播放时云台控制接口(仅实时流)
接口地址	http://ip:port / STREAM/play_handle/PTZControl
接口方法	POST
请求 Header	Content-type: "application/json"
请求 Body	{ "command": 0, "duration":200 }
响应 Body	{ "status_code":200, "status_message":"OK", }
说明	<p>用于对相机外接的云台进行控制（如果有云台的话）</p> <p>command: 控制命令类型，可用列表如下：</p> <ul style="list-style-type: none"> 0: 焦距变大(倍率变大) 1: 焦距变小(倍率变小) 2: 焦点前调 3: 焦点后调 4: 光圈扩大 5: 光圈缩小 6: 云台上仰 7: 云台下俯 8: 云台左转 9 云台右转 10: 云台上仰和左转 11: 云台上仰和右转 12: 云台下俯和左转 13: 云台下俯和右转 <p>duration: 控制时长,单位毫秒，取值范围 100--1000。为空时默认 200ms.</p>

接口说明	停止流推送接口
接口地址	http://ip:port / STREAM/play_handle/STOP
接口方法	GET
请求 Header	
请求 Body	
响应 Body	{ "status_code":200, "status_message":"OK" }
说明	<p>播放结束后调用此接口释放推流通道。</p> <p>play_handle 为取流时获得的播放句柄。</p> <p>.</p>

