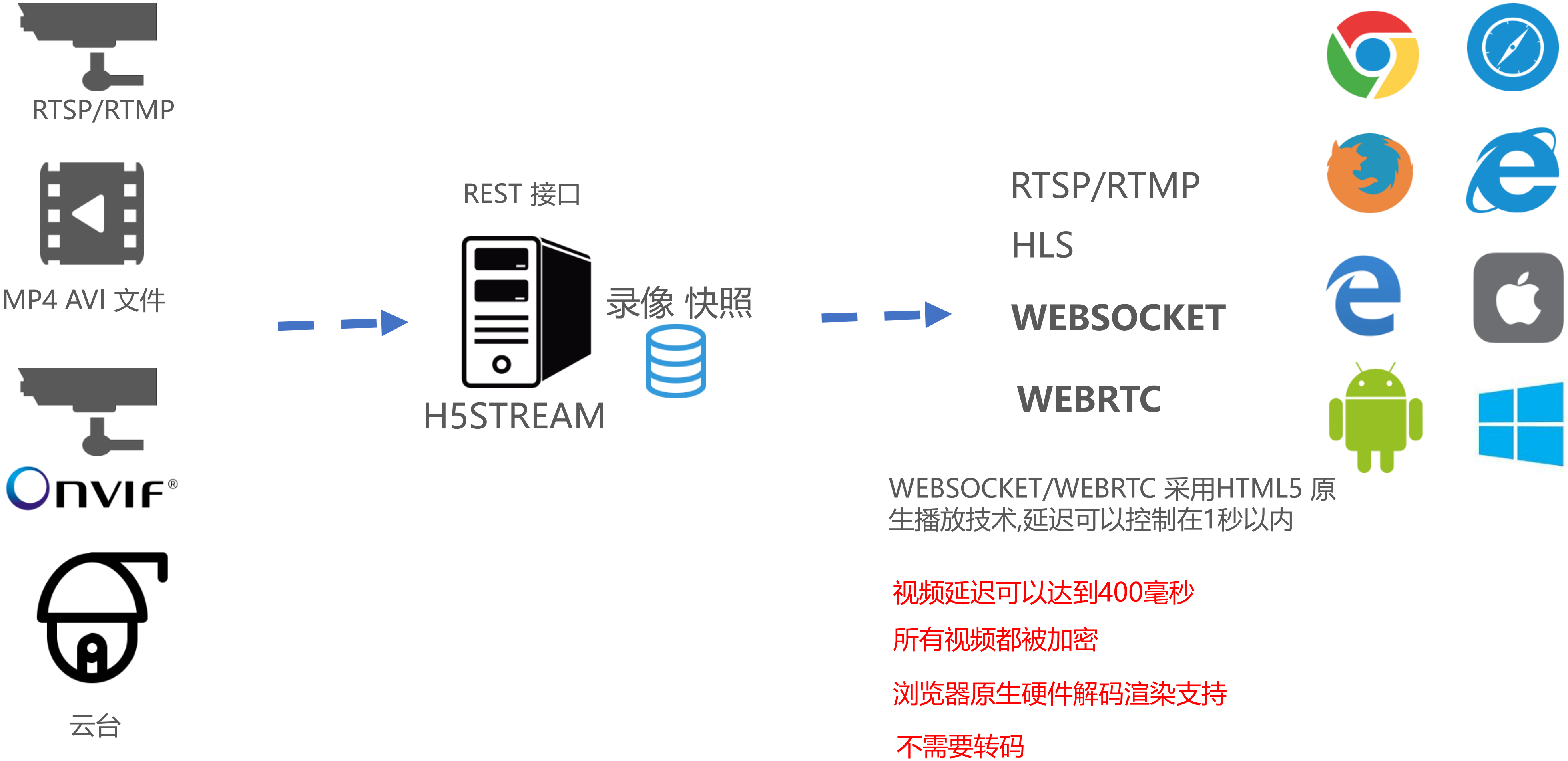
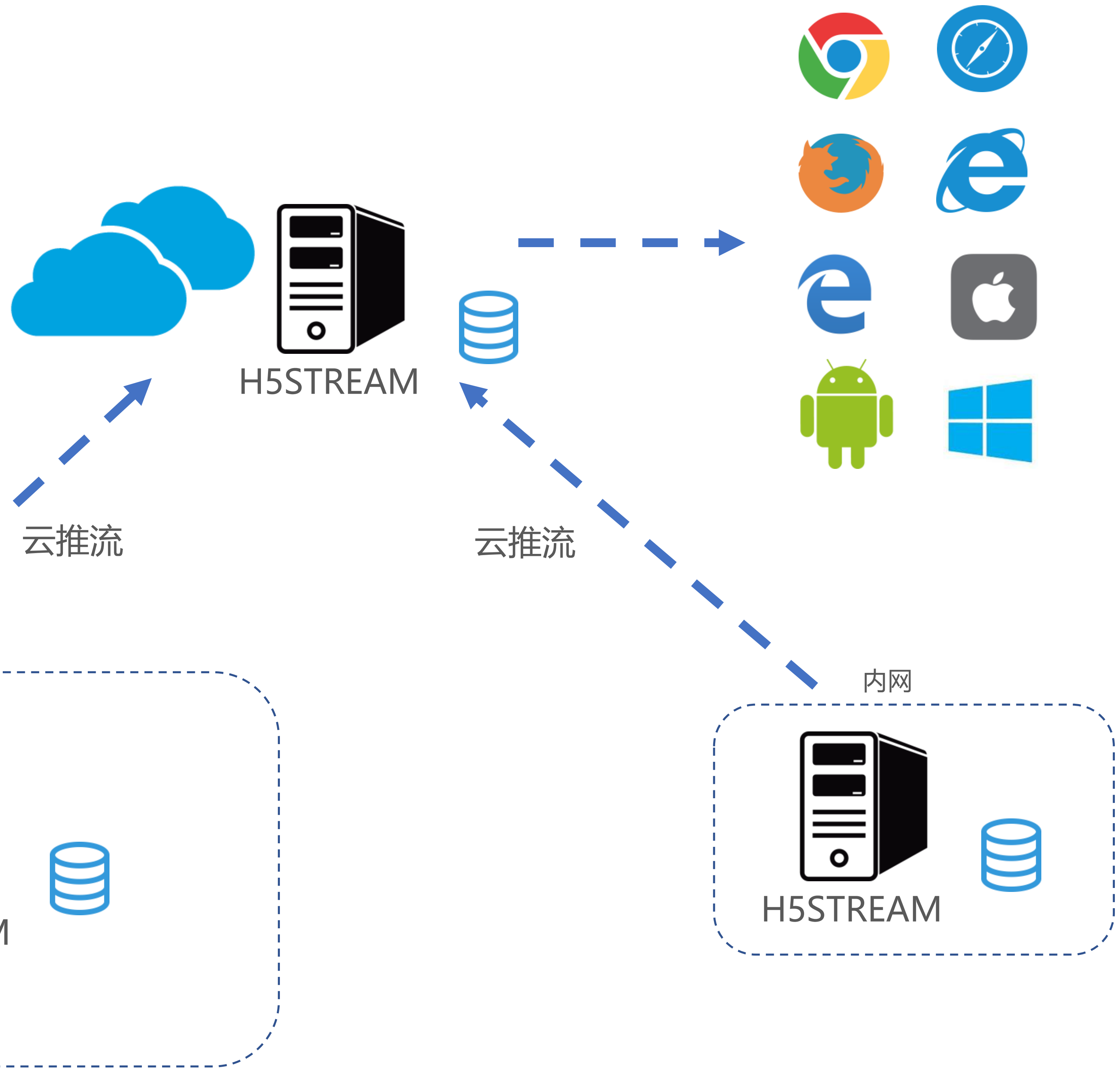
H5STREAM

linkingvison

内网直播


RTSP/RTMP
MP4 AVI 文件
ONVIF®
云台
REST 接口
H5STREAM
录像 快照
RTSP/RTMP
HLS
WEBSOCKET
WEBRTC
WEBSOCKET/WEBRTC 采用HTML5 原生播放技术,延迟可以控制在1秒以内
视频延迟可以达到400毫秒
所有视频都被加密
浏览器原生硬件解码渲染支持
不需要转码

云直播



RESTFUL 接口

系统

Login

Logout

Keepalive

视频源管理

GetSrc

GetImage

AddSrcFile

AddSrcRTSP

AddSrcONVIF

DelSrc

Ptz

ONVIF

OnvifSearch

OnvifProbe

录像和快照

Record

Snapshot

Search

JAVASCRIPT 接口

WEBSOCKET

H5sPlayerWS

```
/**
* Interface with h5s websocket player API
* @constructor
*/

function H5sPlayerWS(conf)

H5sPlayerWS.prototype.connect
H5sPlayerWS.prototype.disconnect
```

WEBRTC

H5sPlayerRTC

```
/**
* Interface with h5s WebRTC player API
* @constructor
* @param {string} videoId - id of the
video element tag
*/
function H5sPlayerRTC(conf)

H5sPlayerRTC.prototype.connect
H5sPlayerRTC.prototype.disconnect
```

RTMP

VideoJS

www/rtmp.html

HLS

H5sPlayerHls

```
/**
* Interface with h5s websocket player API
* @constructor
*/
function H5sPlayerHls(conf)
H5sPlayerHls.prototype.connect
H5sPlayerHls.prototype.disconnect
```

```
/**
 @param
 var conf = {
          videoid:'h5sVideo1', //{string} - id of the video element tag
          videodom: h5svideodom1, //{object} - video dom. if there has videoid, just use the videoid
          protocol: window.location.protocol, // {string} - http: or https:
          host: window.location.host, //{string} - localhost:8080
          rootpath:window.location.pathname, // {string} - path of the app running
          token:'token1', // {string} - token of stream
          hlsver:'v1’, //{string} -  v1 is for ts, v2 is for fmp4
          session:'c1782caf-b670-42d8-ba90-2244d0b0ee83' //{string} - session got from login
*/
```

浏览器兼容性

	Chrome	Firefox	IE11	Edge	Safari	WeChat
WIN7	WEBRTC WEBSOCKET	WEBRTC WEBSOCKET	RTMP	-	-	-
WIN 8/10	WEBRTC WEBSOCKET	WEBRTC WEBSOCKET	RTMP WEBSCOKET	WEBRTC	-	-
macOS	WEBRTC WEBSOCKET	WEBRTC WEBSOCKET	-	-	WEBRTC WEBSOCKET	-
iOS 11	HLS WEBRTC	HLS WEBRTC	-	-	HLS WEBRTC	HLS WEBRTC
iOS 8-10	HLS	HLS	-	-	HLS	HLS
Android	WEBSOCKET WEBRTC	WEBRTC WEBSOCKET	-	-	-	HLS

单协议客户端

http://localhost:8080/ws.html?token=token1&autoplay=1
http://localhost:8080/rtc.html?token=token1&autoplay=1

http://localhost:8080/rtmp.html?token=token2
http://localhost:8080/hls.html?token=token2

Websocket	WebRTC	RTMP	HLS
ws.html	rtc.html	rtmp.html	hls.html

Chrome support this http://localhost:8080/rtc.html?token=token1&autoplay=1
but when change to websocket, chrome does't allow
Uncaught (in promise) DOMException: play() failed because the user didn't interact with the document first. https://goo.gl/xX8pDD

配置文件

conf/h5ss.conf	
HTTP	HTTP HTTPS 服务器配置
RTSP	RTSP 服务器配置, SSL 代表 RTSP over TCP/TLS
RTMP	RTMP服务器配置, SSL 代表 RTMP over TCP/TLS
FLV	FLV服务器配置, SSL 代表 FLV over HTTPS
HLS	HLS服务器配置，包括HLS版本及参数配置
WEBRTC	WEBRTC 配置
SYSTEM	H5stream系统配置，包括日志和线程池配置
USER	用户管理配置
SOURCE	视频源配置，包括文件 RTSP/RTMP/ONVIF

音频开启

h5stream 从5.4开始支持AAC 音频，已经测试过HIKVISION的网络摄像机。
从6.0 开始，h5stream 将会对G711/PCM 音频进行转码成AAC以便支持音频

1.在配置文件中开启音频支持


```
"strPasswd": "admin12345",
"bPasswdEncryptComment": "Password Encrypted",
"bPasswdEncrypt": false,
"bEnableAudioComment": "Enable Audio",
"bEnableAudio": true,
"nConnectTypeComment": "H5_ONDEMAND/H5_ALWAYS/H5_AUTO",
"nConnectType": "H5_AUTO",
"nRTSPTypeComment": "RTSP Connect protocol H5_RTSP_TCP/H5_
"nRTSPType": "H5_RTSP_AUTO",
"strSrcIpAddressComment": "Ip Address for the device",
"strSrcIpAddress": "192.168.100.173",
```


2.把网络摄像机的音频格式修改为AAC