Live555介绍：

是一个为流媒体提供解决方案的跨平台的C++开源项目，它实现了对标准流媒体传输是一个为流媒体提供解决方案的跨平台的C++开源项目，它实现了对标准流媒体传输协议如RTP/RTCP、RTSP、SIP等的支持。Live555实现了对多种音视频编码格式的音视频数据的流化、接收和处理等支持，包括MPEG、H.263+、DV、JPEG视频和多种音频编码。同时由于良好的设计，Live555非常容易扩展对其他格式的支持。目前，Live555已经被用于多款播放器的流媒体播放功能的实现，如VLC(VideoLan)、MPlayer。

Live555主要分为5大模块，分别简绍如下：

UsageEnvironment

The "[UsageEnvironment](http://www.live555.com/liveMedia/doxygen/html/classUsageEnvironment.html)" and "[TaskScheduler](http://www.live555.com/liveMedia/doxygen/html/classTaskScheduler.html)" classes are used for scheduling deferred events, for assigning handlers for asynchronous read events, and for outputting error/warning messages. Also, the "[HashTable](http://www.live555.com/liveMedia/doxygen/html/classHashTable.html)" class defines the interface to a generic hash table, used by the rest of the code.

These are all abstract base classes; they must be subclassed for use in an implementation. These subclasses can exploit the particular properties of the environment in which the program will run - e.g., its GUI and/or scripting environment.

BasicUsageEnvironment模块是UsageEnvironment的一个控制台应用的实现。它针对控制台的输入输出和信号响应进行具体实现。

groupsock

The classes in this library encapsulate network interfaces and sockets. In particular, the "[Groupsock](http://www.live555.com/liveMedia/doxygen/html/classGroupsock.html)" class encapsulates a socket for sending (and/or receiving) multicast datagrams.

liveMedia

This library defines a class hierarchy - rooted in the "[Medium](http://www.live555.com/liveMedia/doxygen/html/classMedium.html)" class - for a variety of streaming media types and codecs.

 LiveMedia模块是Live555最重要的模块。该模块声明了一个[抽象类](http://baike.baidu.com/view/262290.htm)Medium，其他所有类都派生自该类 ，下面简要介绍这些类：

Ø RTSPClient：该类实现RTSP请求的发送和响应的解析，同时根据解析的结果创建对应的RTP会话。

Ø MediaSession：用于表示一个RTP会话，一个MediaSession可能包含多个子会话(MediaSubSession)，子会话可以是音频子会话、视频子会话等。

Ø RTCPInstance：该类实现RTCP协议的通信。

Ø Source和Sink：这两个概念类似DirectShow中的Filter。Source抽象了数据源，比如通过RTP读取数据。Sink是数据消费者的抽象，比如把接收到数据存储到文件，该文件就是一个Sink。数据的流动可能经过多个Source和Sink。MediaSink是各种类型的Sink的基类，MediaSource是各种类型Source的基类，各种类型的流媒体格式和编码的支持即是通过对这两个类的派生实现的。Source和Sink通过RTP子会话(MediaSubSession)联系在一起。

BasicUsageEnvironment

This library defines [one concrete implementation](http://www.live555.com/liveMedia/doxygen/html/classBasicUsageEnvironment.html) (i.e., subclasses) of the "UsageEnvironment" classes, for use in simple, console applications. Read events and delayed operations are handled using a select() loop.

testProgs

This directory implements some simple programs that use "BasicUsageEnvironment" to demonstrate how to develop applications using these libraries.

RTSP client

**testRTSPClient** is a command-line program that shows you how to open and receive media streams that are specified by a RTSP URL - i.e., an URL that begins with *rtsp://*

In this demonstration application, nothing is done with the received audio/video data. You could, however, use and adapt this code in your own application to (for example) decode and play the received data.

[**openRTSP**](http://www.live555.com/openRTSP/) is similar to "testRTSPClient", but has many more features. It is a command-line program that - unlike "testRTSPClient" - is intended to be used as a complete, full-featured application (rather than having its code used within other applications). For more information about "openRTSP" - including its many command-line options - see [the online documentation](http://www.live555.com/openRTSP/).

RTSP server

**testOnDemandRTSPServer** creates a RTSP server that can stream, via RTP unicast, from various types of media file, on demand. (Supported media types include: MPEG-1 or 2 audio or video (elementary stream), including MP3 audio; MPEG-4 video (elementary stream); H.264 video (elementary stream); H.265 video (elementary stream); MPEG Program or Transport streams, including VOB files; DV video; AMR audio; WAV (PCM) audio.) The server can also stream from a [Matroska](http://matroska.org/) or [WebM](http://www.webmproject.org/) file (by demultiplexing and streaming the tracks within the file). MPEG Transport Streams can also be streamed over raw UDP, if requested - e.g., by a set-top box.

This server application also demonstrates how to deliver - via RTSP - a MPEG Transport Stream that arrived at the server as a UDP (raw-UDP or RTP/UDP) multicast or unicast stream. In particular, it is set up, by default, to accept input from the "testMPEG2TransportStreamer" demo application.

SIP client

[**playSIP**](http://www.live555.com/playSIP/) is a command-line program (similar to "openRTSP") that makes a call to a SIP session (using a *sip:* URL), and then (optionally) records the incoming media stream into a file.

Live555机制介绍

只要能理解live555的工作机制和流程就能大致掌握live555的核心，本小节主要内容就是介绍任务调度和工作流程。

任务调度

Live555中有三种任务可供调度处理：分别是：

socket handler

event handler

delay task

响应的调度事件分别为：

socket handler：当对应的socket属性为可读时调度；

event handler：（待研究）；

delay task ：当延时时间到时候就会调用。

1、socket handler 保存在队列BasicTaskScheduler0::HandlerSet\* fHandlers中;

2、event handler保存在数组BasicTaskScheduler0::TaskFunc \* fTriggeredEventHandlers[MAX\_NUM\_EVENT\_TRIGGERS] 中;

3、delay task 保存在队列BasicTaskScheduler0::DelayQueue fDelayQueue 中。

下面看一下三种任务的执行函数的定义：

* socket handler 为

typedef void BackgroundHandlerProc (void\* clientData, int mask);

* event handler为

typedef void TaskFunc(void\* clientData);

* delay task  为

typedef void TaskFunc(void\* clientData);// 跟event handler一样。

再看一下向任务调度对象添加三种任务的函数的样子：

* socket handler 为

void setBackgroundHandling(int socketNum, int conditionSet   ,BackgroundHandlerProc\* handlerProc, void\* clientData)

* event handler为

EventTriggerId createEventTrigger(TaskFunc\* eventHandlerProc)

* delay task 为

TaskToken scheduleDelayedTask(int64\_t  microseconds, TaskFunc\* proc,void\* clientData)

工作流程

1. 创建rtspServer以端口554创建监听；
2. 若收到相机准确可用的上报信息则创建ServerMediaSession并将其加入RTSPServer中；
3. 创建RTSPClient并发送describe指令；
4. 根据SDP创建ProxyServerSubSession，该session用来充当服务器，代理指定摄像机的视频流；
5. 监听端口554判断是否有连接请求，如果有则调用对应的handler去处理若为连接请求则创建clientSession；
6. 监听clientSession的监听端口判断是否有RTSP指令，如果有请求则调用相应的handler处理。