RTP 协议分析

一. RTP 协议背景 2
二. RTP 协议原理及工作机制 2
2．1 RTP 协议原理 3
2．1．1 RTP 协议原理 3
2．1．2 RTCP 协议原理 3
2．2 RTP 数据包格式 4
2．2．1 RTP 数据包格式 4
2．2．2 RTCP 数据包格式 6
2．3 RTP 工作机制 9
2．3．1 RTP 工作机制 9
2．3．2 RTCP 工作机制 9
三. RTP 协议关键技术指标 10
3．1 时间戳 10
3．2 时延 10
3．3 抖动 11
3．4 丢包率 11
3．5 会话和流两级分用 11
3．6 多种流同步控制 12
四. RTP 协议应用方案 12
4．1 RTP 协议应用方案之单播 12
4．2 RTP 协议应用方案之广播 12
4．3 RTP 协议应用方案之组播 13
4．3.1 RTP 协议组播方案总体概述 13
4．3.2 RTP 协议组播方案服务器端实现 14
4．3. 3RTP 协议组播方案客户端实现 14
4．3. 4RTP 协议视频帧率和质量调整策略 15
五. RTP 协议移植计划 16
六. RTP 协议安全方面考虑 16

一． RTP 协议背景

流（Streaming）是近年在 Internet 上出现的新概念，其定义非常广泛，主要是指通过网络传输多媒体数据的技术总称。流媒体包含广义和狭义两种内涵：广义上的流媒体指的是使音频和视频形成稳定和连续的传输流和回放流的一系列技术、方法和协议的总称，即流媒体技术；狭义上的流媒体是相对于传统的下载-回放方式而言的，指的是一种从 Internet 上获取音频和视频等多媒体数据的新方法，它能够支持多媒体数据流的实时传输和实时播放。通过运用流媒体技术，服务器能够向客户机发送稳定和连续的多媒体数据流，客户机在接收数据的同时以一个稳定的速率回放，而不用等数据全部下载完之后再进行回放。

流式传输有顺序流式传输(Progressive Streaming)和实时流式传输(Realtime Streaming)两种方式。实时流式传输是实时传送，特别适合现场事件，实时流式传输必须匹配连接带宽，这意味着图像质量会因网络速度降低而变差，以减少对传输带宽的需求。“实时”的概念是指在一个应用中数据的交付必须与数据的产生保持精确的时间关系，这需要相应的协议支持，这样 RTP 和 RTCP 就相应的出现了。

实时传输协议 RTP（Realtime Transport Protocol）：是针对 Internet 上多媒体数据流的一个传输协议，由 IETF 作为 RFC1889 发布，现在最新的为 RFC3550。RTP 被定义为在一对一或一对多的传输情况下工作，其目的是提供时间信息和实现流同步。RTP 的典型应用建立在 UDP 上，但也可以在 TCP 等其他协议之上工作。RTP 本身只保证实时数据的传输，并不能为按顺序传送数据包提供可靠的传送机制，也不提供流量控制或拥塞控制，它依靠 RTCP 提供这些服务。

实时传输控制协议 RTCP（Realtime Transport Control Protocol）：负责管理传输质量，在当前应用进程之间交换控制信息，提供流量控制和拥塞控制服务。在 RTP 会话期间，各参与者周期性地传送 RTCP 包，包中含有已发送的数据包的数量、丢失的数据包的数量等统计资料，因此，服务器可以利用这些信息动态地改变传输速率，甚至改变有效载荷类型。RTP 和 RTCP 配合使用，能以有效的反馈和最小的开销使传输效率最佳化，故特别适合传送网上的实时数据。

二． RTP 协议原理及工作机制

让我们先看一下 RTP 和 RTCP 在网络层次中的位置，以便我们更加清楚的了解该协议，如下图 1-1 所示：

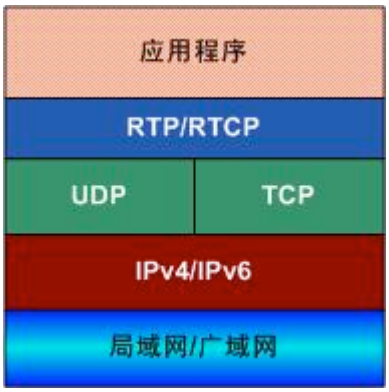


图 1-1 RTP&RTCP 网络层次关系图

下面我们就从 RTP 以及 RTCP 的协议原理，数据包格式，工作机制三个方面来对该协议做一个基本的认识和了解：

2．1 RTP 协议原理

2．1．1 RTP 协议原理

RTP 协议原理比较简单，负责对流媒体数据进行封包并实现媒体流的实时传输，即它按照 RPT 数据包格式来封装流媒体数据，并利用与它绑定的协议进行数据包的传输，具体见本文 2.2.1RTP 数据格式；RTP 本身只保证实时数据的传输，并不能为按顺序传送数据包提供可靠的传送机制，也不提供流量控制或拥塞控制，它依靠 RTCP 提供这些服务.

2．1．2 RTCP 协议原理

RTCP 原理是向会话中的所有成员周期性地发送控制包来实现的，应用程序通过接收这些控制数据包，从中获取会话参与者的相关资料，以及网络状况、分组丢失概率等反馈信息，从而能够对服务质量进行控制或者对网络状况进行诊断.

RTCP 协议的功能是通过不同的 RTCP 数据报文(具体描述的见 2.2.2RTCP 数据包格式)来实现的，主要有如下几种类型：

- SR(Sender Report) 发送端报告，所谓发送端是指发出 RTP 数据报的应用程序或者终端，发送端同时也可以是接收端。
- RR(Receiver Report) 接收端报告，所谓接收端是指仅接收但不发送 RTP 数据报的应用程序或者终端。
- SDES 源描述，主要功能是作为会话成员有关标识信息的载体，如用户名、邮件地址、电话号码等，此外还具有向会话成员传达会话控制信息的功能。
- BYE 通知离开，主要功能是指示某一个或者几个源不再有效，即通知会话中的其他成员自己将退出会话。
- APP 由应用程序自己定义，解决了 RTCP 的扩展性问题，并且为协议的实现者提供了很大的灵活性。

RTCP 数据报携带有服务质量监控的必要信息，能够对服务质量进行动态的调整，并能够对网络拥塞进行有效的控制。由于 RTCP 数据报采用的是组播方式，因此会话中的所有成员都可以通过 RTCP 数据报返回的控制信息，来了解其他参与者的当前情况。

例如在流媒体应用场合下，发送媒体流的应用程序将周期性地产生发送端报告 SR，该 RTCP 数据报含有不同媒体流间的同步信息，以及已经发送的数据报和字节的计数，接收端根据这些信息可以估计出实际的数据传输速率。另一方面，接收端会向所有已知的发送端发送接收端报告 RR，该 RTCP 数据报含有已接收数据报的最大序列号、丢失的数据报数目、延时抖动和时间戳等重要信息，发送端应用根据这些信息可以估计出往返时延，并且可以根据数据报丢失概率和时延抖动情况动态调整发送速率，以改善网络拥塞状况，或者根据网络状况平滑地调整应用程序的服务质量。

RTCP 具有以下四个功能：

1、基本功能是提供数据传输质量的反馈.这是 RTP 作为一种传输协议的主要作用,它与其他协议的流量和阻塞控制相关.反馈可能对自适应编码有直接作用,但是 IP 组播的实验表明它对于从接收机得到反馈信息以诊断传输故障也有决定性作用.向所有成员发送接收反馈可以使"观察员"评估这些问题是局部的还是全局的.利用类似多点广播的传输机制,可以使某些实体,诸如没有加入会议的网络网络业务观察员,接收到反馈信息并作为第三类监视员来诊断网络故障.反馈功能通过 RTCP 发射机和接收机报告实现.

2、RTCP 为每个 RTP 源传输一个固定的识别符,称为标称名或 CNAME.由于当发生冲突或程序重启时 SSRC 可能改变,接收机要用 CNAME 来跟踪每个成员.接收机还要用 CNAME 来关联一系列相关 RTP 会话期中来自同一个成员的多个数据流,例如同步语音和图象.

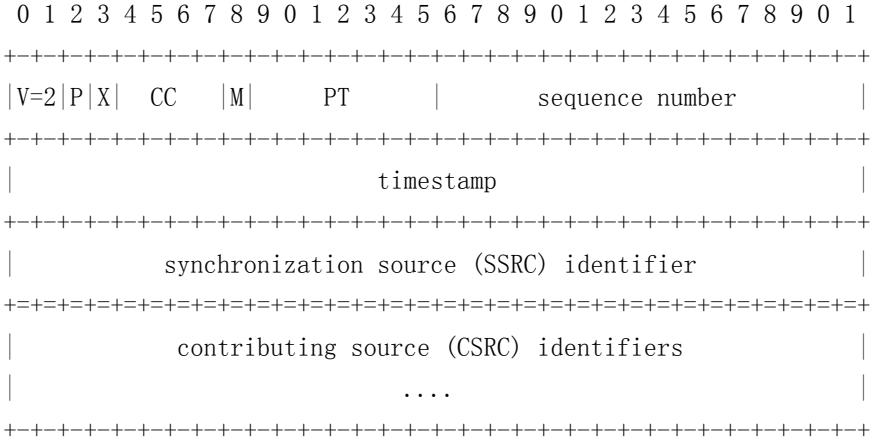
3、前两个功能要求所有成员都发送 RTCP 包,因此必须控制速率以使 RTP 成员数可以逐级增长.通过让每个成员向所有成员发送控制包,各个成员都可以独立地观察会议中所有成员的数目.

4、可选的功能是传输最少的会议控制信息,例如在用户接口中显示的成员识别.这最可能在"松散控制"的会议中起作用,在"松散控制"会议里,成员可以不经过资格控制和参数协商而加入或退出会议.RTCP 作为一个延伸到所有成员的方便通路,必须要支持具体应用所需的所有控制信息通信.

2．2 RTP 数据包格式

2．2．1 RTP 数据包格式

RTP 报文头格式（见 RFC3550 Page12）：



以上域具体意义如下：
版本(V):2 比特 此域定义了 RTP 的版本.此协议定义的版本是 2.(值 1 被 RTP 草案版本使用,值 0 用在最初"vat"语音工具使用的协议中.)
填料(P):1 比特 若填料比特被设置,此包包含一到多个附加在末端的填充比特,不是负载的一部分.填料的最后一个字节包含可以忽略多少个填充比特.填料可能用于某些具有固定长度的加密算法,或者在底层数据单元中传输多个 RTP 包.
扩展(X):1 比特 若设置扩展比特,固定头(仅)后面跟随一个头扩展.
CSRC 计数(CC):4 比特 CSRC 计数包含了跟在固定头后面 CSRC 识别符的数目.
标志(M):1 比特 标志的解释由具体协议规定.它用来允许在比特流中标记重要的事件,如帧范围.规定该标志在静音后的第一个语音包时置位.

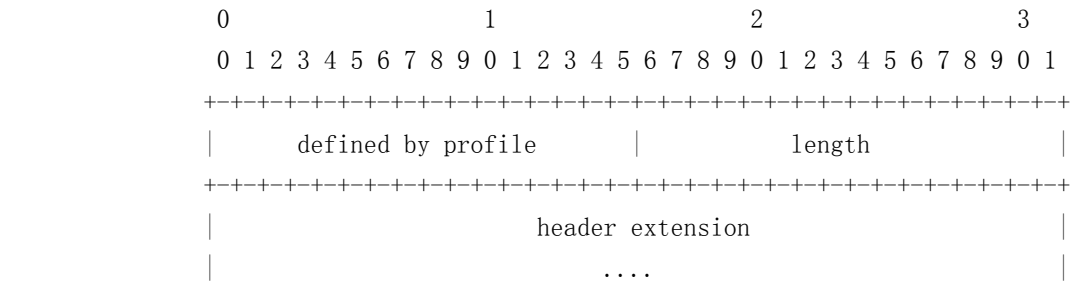
负载类型(PT):7 比特 此域定义了负载的格式,由具体应用决定其解释.协议可以规定负载类型码和负载格式之间一个默认的匹配.其他的负载类型码可以通过非 RTP 方法动态定义.RTP 发射机在任意给定时间发出一个单独的 RTP 负载类型;此域不用来复用不同的媒体流.
序列号（sequence number）:16 比特 每发送一个 RTP 数据包,序列号加一,接收机可以据此检测包损和重建包序列.序列号的初始值是随机的(不可预测),以使即便在源本身不加密时(有时包要通过翻译器,它会这样做),对加密算法泛知的普通文本攻击也会更加困难.

时间标志（timestamp）:32 比特 时间标志反映了 RTP 数据包中第一个比特的抽样瞬间.抽样瞬间必须由随时间单调和线形增长的时钟得到,以进行同步和抖动计算.时钟的分辨率必须满足要求的同步准确度,足以进行包到达抖动测量.时钟频率与作为负载传输的数据格式独立,在协议中或定义此格式的负载类型说明中静态定义,也可以在通过非 RTP 方法定义的负载格式中动态说明.若 RTP 包周期性生成,可以使用由抽样时钟确定的额定抽样瞬间,而不是读系统时钟.例如,对于固定速率语音,时间标志钟可以每个抽样周期加 1.若语音设备从输入设备读取覆盖 160 个抽样周期的数据块,对于每个这样的数据块,时间标志增加 160,无论此块被发送还是被静音压缩.
时间标志的起始值是随机的,如同序列号.多个连续的 RTP 包可能由同样的时间标志,若他们在逻辑上同时产生.如属于同一个图象帧.若数据没有按照抽样的
顺序发送,连续的 RTP 包可以包含不单调的时间标志,如 MPEG 交织图象帧.

同步源（SSRC）:32 比特 SSRC 域用以识别同步源.标识符被随机生成,以使在同一个 RTP 会话期中没有任何两个同步源有相同的 SSRC 识别符.尽管多个源选择同一个 SSRC 识别符的概率很低,所有 RTP 实现工具都必须准备检测和解决冲突.若一个源改变本身的源传输地址,必须选择新的 SSRC 识别符,以避免被当作一个环路源.

有贡献源（CSRC）列表:0 到 15 项,每项 32 比特 CSRC 列表识别在此包中负载的有贡献源.识别符的数目在 CC 域中给定.若有贡献源多于 15 个,仅识别 15 个.CSRC 识别符由混合器插入,用有贡献源的 SSRC 识别符.例如语音包,混合产生新包的所有源的 SSRC 标识符都被陈列,以期在接收机处正确指示交谈者.
注意：前 12 个字节出现在每个 RTP 包中,仅仅在被混合器插入时,才出现 CSRC 识别符列表.

RTP 报文扩展头格式（见 RFC3550 Page18）：



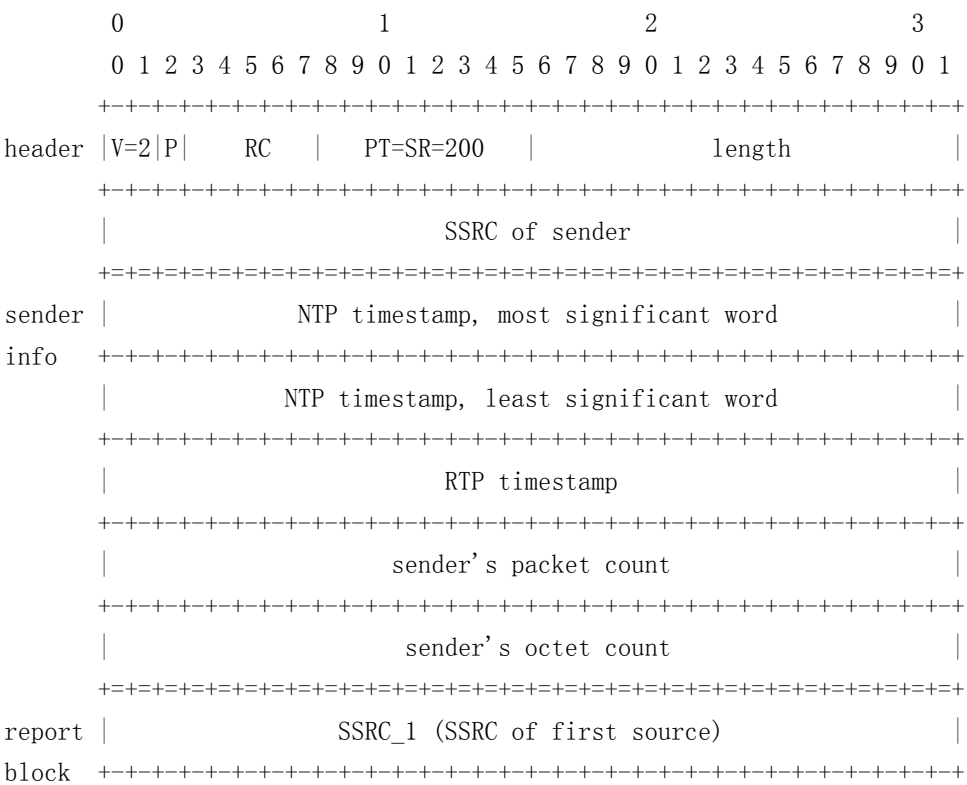
若 RTP 头中的扩展比特位 X 置 1,则一个长度可变的头扩展部分被加到 RTP 固定头之后,.头扩展包含 16 比特的长度域,指示扩展项中 32 比特字的个数,不包括 4 个字节扩展头(因此零是有效值).RTP 固定头之后只允许有一个头扩展.为允许多个互操作实现独立生成不同的头扩展,或某种特定实现有多种不同的头扩展,扩展项的前 16 比特用以识别标识符或参数.这 16 比特的格式由具体实现的上层协议定义.基本的 RTP 说明并不定义任何头扩展本身。

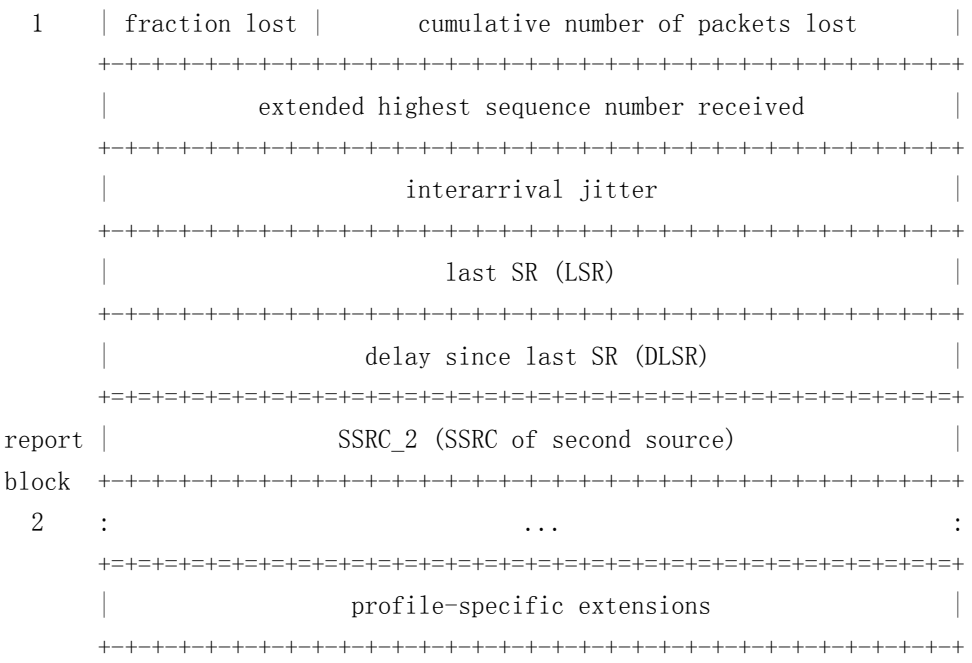
2．2．2 RTCP 数据包格式

RTCP 包括五种数据包类型（RFC3550 Page69）：

abbrev.	name	value(该值RTCP头格式中的PT类型字段)
SR	sender report	200
RR	receiver report	201
SDES	source description	202
BYE	goodbye	203
APP	application-defined	204

现在我们就 SR 报文为例详细描述一下 RTCP 报文格式（RFC3550 Page35）：





每个 RTCP 包的开始部分是与 RTP 数据包相类似的固定部分,随后是一块结构化单元,它随负载类型不同长度发生变化,但是总以 32 比特终止.

发射机报告包由 3 部分组成,若定义,可能跟随第 4 个面向协议的扩展部分.

第一部分:

8 字节长.该域有以下意义:

版本(V):2 比特 RTP 版本识别符,在 RTCP 包内的意义与 RTP 包中的相同.此协议中定义的版本号为 2.

填料(P):1 比特 若设置填料比特,该 RTCP 包在末端包含一些附加填料比特,并不是控制信息的基本部分.填料的最后一个比特统计了多少个字节必须被忽略.某些有固定块大小的加密算法可能需要填料比特.在复合 RTCP 包中,复合包作为一个整体加密,填料比特只能加在最后一个单包的后面.

接收报告块计数(RC):5 比特 该包中所含接收报告块的数目.零值有效.

包类型(PT):8 比特 包含常数 200,用以识别这个为 RTCP SR 包.

长度:16 比特 以 32 比特字为单位,该 RTCP 包的长度减一,包括头和任何填料.(偏移量 1 保证零值有效,避免了在扫描 RTCP 包长度时可能发生的无限循环,同时以 32 比特为单位避免了对以 4 为倍数的有效性检测.)

SSRC:32 比特 SR 包发起者的同步源标识符.

第二部分:

发射机信息,20 比特长,在每个发射机报告包中出现.它概括了从此发射机发出的数据传输情况.此域有以下意义:

NTP 时间标志:64 比特 指示了此报告发送时的壁钟时刻,它可以与从其它接收机返回的接收报告块中的时间标志结合起来,测量到这些接收机的环路时沿.接收机必须期望此时间标志的准确度远低于 NTP 时间标志的分辨率.测量的不确定度不可知,因此也无需指示.某个发射机,能够跟踪逝去时间但是无法跟踪壁钟时间,可以用加入会议后的逝去时间代替.假定该值小于 68 年,则最高比特为零.允许用抽样时钟估计逝去壁钟时间.无法用壁钟时间或逝去时间的可以设置此项为零.

RTP 时间标志:32 比特 与以上的 NTP 时间标志对应同一时刻,但是与数据包中的 RTP 时间标志具有相同的单位和偏移量.这个一致性可以用来让 NTP 时间标志已经同步的源间进行媒体内/间同步,还可以让与媒体无关的接收机估计标称 RTP 时钟频率.注意在大多数情况下此时间标志不等于任何临近的 RTP 包中的时间标志.然而,通过"RTP 时间标志计数器"和"由在抽样点上周期性检测壁钟时间得到的实际时间"两者之间的关系,可以通过相应的 NTP 时间标志计算得到此 RTP 时间标志.

发送的报文数:32 比特 从开始传输到此 SR 包产生时该发射机发送的 RTP 数据包总数.若发射机改变 SSRC 识别符,该计数器重设.

发送的字节文数:32 比特 从开始传输到此 SR 包产生时该发射机在 RTP 数据包发送的字节总数(不包括头和填料).若发射机改变 SSRC 识别符,该计数器重设.此域可以用来估计平均负载类型数据速率.

第三部分:

零到多个接收报告块,块数等于从上一个报告以来该发射机收听到的其它源的数目.每个接收报告块传输关于从某个同步源来的数据包的接收统计信息.若某个源因冲突而改变其SSRC识别符,接收机并不延续统计数字.这些统计数字是:

SSRC_n(源识别符):32比特 在此接收报告块中信息所属源的SSRC识别符.

丢包率:8比特 自从前一SR包或RR包发射以来,从SSRC_n传来的RTP数据包的损失比例,以固定点小数的形式表示,小数点在此域的左侧,等于将损失比例乘256后取整数部分.该值定义为损失包数被期望接收的包数除,在下一段中定义.若由于复制而导致包损为负值,损失比例值设为零.注意在收到上一个包后,接收机无法告之以后的包是否丢失,若在上一个接收报告间隔内从某个源发出的所有数据包都丢失,那么将不为此源发送接收报告块.

累计包丢失数:24比特 从开始接收到现在,从源SSRC_n发到本源的RTP数据包的丢包总数.该值定义为期望接收的包数减去实际接收的包数,接收的包括复制的或迟到的.由于迟到的包不算作损失,在发生复制时包损可能为负值.期望接收的包数定义为扩展的上一接收序号(随后定义)减去最初接收序号.

接收到的扩展的最高序列号:32比特 低16比特包含从源SSRC_n来的最高接收序列号,高16比特用相应的序列号周期计数器扩展该序列号.注意在同一会议中的不同接收机,若启动时间明显不同,将产生不同的扩展项.

到达间隔抖动:32比特 RTP数据包到达时刻统计方差的估计值,以时间标志为单位测量,用无符号整数表达.到达时刻抖动J定义为一对包中接收机相对发射机的时间跨度差值的平均偏差(平滑后的绝对值).如以下等式所示,该值等于两个包相对传输时间的差值,相对传输时间是指包的RTP时间标志和到达时刻接收机时钟,以同一单位的差值.若Si是包i的RTP时间标志,Ri是包i以RTP时间标志单位的到达时刻值,对于两个包i和j,D可以表达为

D(i,j) = (Rj - Ri) - (Sj - Si) = (Rj - Sj) - (Ri - Si)

到达时刻抖动可以在收到从源SSRC_n来的每个数据包i后连续计算,利用该包和前一包i-1的偏差D(按到达顺序,而非序号顺序),根据公式J(i) = J(i-1) + (|D(i-1,i)| - J(i-1))/16计算.无论何时发送接收报告,都用当前的J值.

此处描述的抖动计算允许与协议独立的监视器对来自不同实现的报告进行有效的解释.

上一SR报文 (LSR):32比特 接收到的来自源SSRC_n的最新RTCP发射机报告(SR)的64位NTP时间标志的中间32位.若还没有接收到SR,该域值为零.

自上一SR的时间延时(DLSR):32比特 是从收到来自SSRC_n的SR包到发送此接收报告块之间的延时,以1/65536秒为单位.若还未收到来自SSRC_n的SR包,该域值为零.

假设SSRC_r为发出此接收报告块的接收机.源SSRC_n可以通过记录收到此接收报告块的时刻

A来计算到SSRC_r的环路传输时延.可以利用最新的SR时间标志(LSR)域计算整个环路时间A-LSR,然后减去此DLSR域得到环路传播时延.
可以用此来近似测量到一族接收机的距离,尽管有些连接可能有非常不对称的时延.

接收机报告包(RR)与发射机报告包基本相同,除了包类型域包含常数 201 和没有发射机信息的 5 个字(NTP 和 RTP 时间标志和发射机包和字节计数).余下区域与 SR 包意义相同.若没有发送和接收据报告,在 RTCP 复合包头部加入空的 RR 包(RC=0)。

其它三种报文的格式由于比较简单，不再具体描述；

2．3 RTP 工作机制

2．3．1 RTP 工作机制

RTP 根据应用程序的要求将流媒体数据包封装成 RTP 数据包并进行发送；它靠上层的调用以及依赖网络层发送来实现；

工作时，RTP 协议从上层接收流媒体信息码流（如 H.263），装配成 RTP 数据包发送给下层，下层协议提供 RTP 和 RTCP 的分流。如在 UDP 中，RTP 使用一个偶数号端口，则相应的 RTCP 使用其后的奇数号端口。RTP 数据包没有长度限制，它的最大包长只受下层协议的限制。

2．3．2 RTCP 工作机制

RTCP 报文不封装音视频数据，而是封装发送端或者接收端的统计报表信息；

在 RTP 会话期间，每个参与者周期性的向其它参与者发送 RTCP 控制信息包，如下图 1-2 所示：

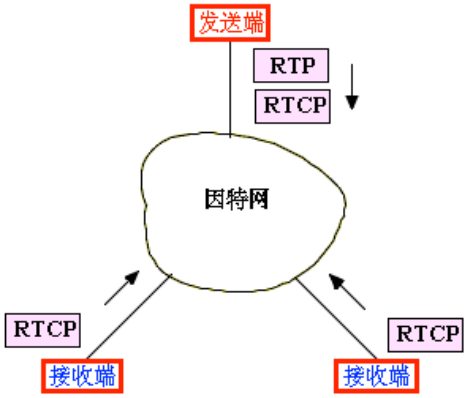


图 1-2 RTCP 工作示意图

因为网络的情况很不稳定，如果网络情况好我们可以减少语音的延迟时间，也可以增大视频的发送帧率或质量。若网络状况不好我们可以增大语音延迟时间以保证语音连续，也可减少视频的发送帧率或质量，以减少网络的阻塞。
RTCP 包的发送率根据与会者的数量来调整.

三． RTP 协议关键技术指标

3．1 时间戳

时间戳字段是 RTP 首部中说明数据包时间的同步信息，是数据能以正确的时间顺序恢复的关键。时间戳的值给出了分组中数据的第一个字节的采样时间(Sampling Instant)，要求发送方时间戳的时钟是连续、单调增长的，即使在没有数据输入或发送数据时也是如此。在静默时，发送方不必发送数据，保持时间戳的增长，在接收端，由于接收到的数据分组的序号没有丢失，就知道没有发生数据丢失，而且只要比较前后分组的时间戳的差异，就可以确定输出的时间间隔。

RTP 规定一次会话的初始时间戳必须随机选择，但协议没有规定时间戳的单位，也没有规定该值的精确解释，而是由负载类型来确定时钟的颗粒，这样各种应用类型可以根据需要选择合适的输出计时精度。

在 RTP 传输音频数据时，一般选定逻辑时间戳速率与采样速率相同，但是在传输视频数据时，必须使时间戳速率大于每帧的一个滴答。如果数据是在同一时刻采样的，协议标准还允许多个分组具有相同的时间戳值，如多个分组属于同一画像。

RTCP 中的 SR（Sender Report 发送端报告）控制分组包含 NTP（网络时间，是以 1900-1-1 零时为起点的系统绝对时间）时间戳和 RTP 时间戳（封装数据时候打上的时间戳与媒体帧上打上的时间戳不同）可用于同步音视频媒体流。其实现机制如下：
RTP 时间戳是依据邻近的 RTP 数据包中的时间戳结合 NTP 时间差得到的，用公式表达为：
RTP_tsi = tsi + NTPi-NTP'i
其中：
RTP_tsi 表示 RTCP 中的 RTP 时间戳；tsi 表示邻近的 RTP 包中的时间戳；NTPi 表示 RTCP 的网络时间戳；NTP'i 表示邻近的 RTP 包对应的网络时间戳；下标表示第 i 个源。
RTP_tsj=tsj+NTPj—NTP'j 表示第 j 个源的 RTP 时间戳；
因此，i 和源 j 之间的相对时差可以表示为：
（RTP_tsi – tsi） -(RTP_tsj - tsj) = (NTPi –NTP'i) - （NTPj—NTP'j）；
由于 NTP 同步，差值可以反映出两个源的相对时差。因为要同步不同来源的媒体流，必须使得同步他们的绝对时间基准，而 NTP 时间戳正是这样的绝对时间基准[4]。而对于同一来源的媒体流，应用 RTP 的时间戳来保证其同步。

3．2 时延

影响时延的因素有多个方面：编解码、网络、防抖动缓冲、报文队列等都影响时延，其中有些是固定时延，如编解码网络速率等；有些是变化的，如防抖动缓冲和队列调度等，固定的时延可以通过改变编解码方式和提高网络速率来改变，而变化的时延通常采用提高转发效率来提高；

假设 SSRC_r 为发出一个接收报告块的接收机. 源 SSRC_n 可以通过记录收到接收报告块的时刻 A 来计算到 SSRC_r 的环路传输时延. 可以利用最新的 SR 时间标志(LSR)域计算整个环路

时间 A-LSR,然后减去此 DLSR 域得到环路传播时延.

3．3 抖动

在视频电话中，语音、视频数据都是使用 UDP 协议传送的，但这种协议传输的数据包在网络层不能保证其发送顺序，需要应用层进行排序。在网络的传输中都会有延时，且随着网络负载的变化，延时的长短也不相同，对于语音数据，如果接收方收到后立即播放，很容易造成语音的抖动。

RTP 数据包到达时刻统计方差的估计值,以时间标志为单位测量,用无符号整数表达
到达时刻抖动J定义为一对包中接收机相对发射机的时间跨度差值的平均偏差(平滑后的绝对值).如以下等式所示,该值等于两个包相对传输时间的差值,相对传输时间是指包的RTP时间标志和到达时刻接收机时钟,以同一单位的差值.若Si是包i的RTP时间标志,Ri是包i以RTP时间标志单位的到达时刻值,对于两个包i和j,D可以表达为

$$D(i,j) = (Rj - Ri) - (Sj - Si) = (Rj - Sj) - (Ri - Si)$$

到达时刻抖动可以在收到从源SSRC_n来的每个数据包i后连续计算,利用该包和前一包i-1的偏差D(按到达顺序,而非序号顺序),根据公式

$$J(i) = J(i-1) + (|D(i-1,i)| - J(i-1))/16$$

计算.无论何时发送接收报告,都用当前的J值.

为了更好的解决抖动的问题，最好能实现抖动缓存（原理比较简单，在此不做详细描述），一是保证语通道读取数据包的顺序正确，二是控制接收方按照采集的时间顺序播放语音，减少语音的抖动；另外提供QoS和资源预留使语音数据获得优先发送和获得固定的带宽也是解决抖动问题的主要手段。

3．4 丢包率

丢包率是通过计算接收包数量和发送包数量的比率得到的，丢包率获得的整个流程是：发送方每间隔一定时间读取每个发送通道的发包数量和数据长度，组成一个此通道的 RTCP 报文发送给接收方，同时将发送数据包计数清零；接收方收到 RTCP 包后，读取接收通道接收到的包数量，并计算出丢包率，通过一个 RTCP 接收汇报包发送给发送方，同时对接收数据包计数清零。

自从前一 SR 包或 RR 包发射以来,从 SSRC_n 传来的 RTP 数据包的损失比例,以固定点小数的形式表示,定义为损失包数被期望接收的包数除,在下一段中定义.若由于复制而导致包损为负值,损失比例值设为零.注意在收到上一个包后,接收机无法告之以后的包是否丢失,若在上一个接收报告间隔内从某个源发出的所有数据包都丢失,那么将不为此源发送接收报告块.

3．5 会话和流两级分用

一个 RTP 会话(Session)包括传给某个指定目的地对(Destination Pair)的所有通信量，发送方

可能包括多个。而从同一个同步源发出的 RTP 分组序列称为流(Stream),一个 RTP 会话可能包含多个 RTP 流。一个 RTP 分组在服务器端发送出去的时候总是要指定属于哪个会话和流，在接收时也需要进行两级分用，即会话分用和流分用。只有当 RTP 使用同步源标识(SSRC)和分组类型(PTYPE)把同一个流中的分组组合起来，才能够使用序列号(Sequence Number)和时间戳(Timestamp)对分组进行排序和正确回放。

3．6 多种流同步控制

RTCP 的一个关键作用就是能让接收方同步多个 RTP 流，例如：当音频与视频一起传输的时候，由于编码的不同，RTP 使用两个流分别进行传输，这样两个流的时间戳以不同的速率运行，接收方必须同步两个流，以保证声音与影像的一致。为能进行流同步，RTCP 要求发送方给每个传送一个唯一的标识数据源的规范名(Canonical Name)，尽管由一个数据源发出的不同的流具有不同的同步源标识(SSRC)，但具有相同的规范名，这样接收方就知道哪些流是有关联的。而发送方报告报文所包含的信息可被接收方用于协调两个流中的时间戳值。发送方报告中含有一个以网络时间协议 NTP(Network Time Protocol)格式表示的绝对时间值，接着 RTCP 报告中给出一个 RTP 时间戳值，产生该值的时钟就是产生 RTP 分组中的 TimeStamp 字段的那个时钟。由于发送方发出的所有流和发送方报告都使用同一个绝对时钟，接收方就可以比较来自同一数据源的两个流的绝对时间，从而确定如何将一个流中的时间戳值映射为另一个流中的时间戳值。

四． RTP 协议应用方案

4．1 RTP 协议应用方案之单播

在客户端与媒体服务器之间建立一个单独的数据通道，从一台服务器送出的每个数据包只能传送给一个客户端，这种传送方式称为单播。

优点：便于控制和管理；
缺点：每个用户必须分别对媒体服务器发送单独的查询，而媒体服务器必须向每个用户发送所申请的数据包拷贝。这种巨大冗余造成服务器负担沉重，响应需要很长时间.

4．2 RTP 协议应用方案之广播

广播指的是用户被动地接收流。在广播过程中，数据包的单独一个拷贝将发送给网络上的所有用户，客户端接收流，但不能控制流；广播方式中资料包的单独一个拷贝将发送给网络上的所有用户，而不管用户是否需要，会非常浪费网络带宽。

优点：简单
缺点：浪费网络带宽

4．3 RTP 协议应用方案之组播

组播技术构建的网络，允许路由器一次将数据包复制到多个通道上。采用组播方式，媒体服务器只需要发送一个信息包，所有发出请求的客户端即可同时收到连续数据流而无延时。这就大大减少了网络上传输的信息包的总量，组播吸收了单播和广播两种发送方式的长处，克服了上述两种发送方式的弱点，将资料包的单独一个拷贝发送给需要的那些客户。组播不会复制资料包的多个拷贝传输到网络上，也不会将资料包发送给不需要它的那些客户，保证了网络上多媒体应用占用网络的最小带宽。

优点：减少网络上传输的信息包的总量。网络利用效率大大提高，成本大为下降；
缺点：当不同的用户同时点播同一个节目时，由于点播总有先后顺序，后点播的用户并不是从开始播放，而是依照网络中同时点播此节目的其它用户的播放进度，这就造成当前用户极有可能从节目的中间开始看起, 很难做到个性化。

下面我们就组播方案进行详细的分析：

4．3.1 RTP 协议组播方案总体概述

组播方案中包括由服务器端、客户端和组播网络载体三部分组成，由于组播网络对于客户端和服务器来说可以是屏蔽的，因此本文仅对服务器和客户端做详细描述，组播整体方案示意图如下图 1-3 所示：

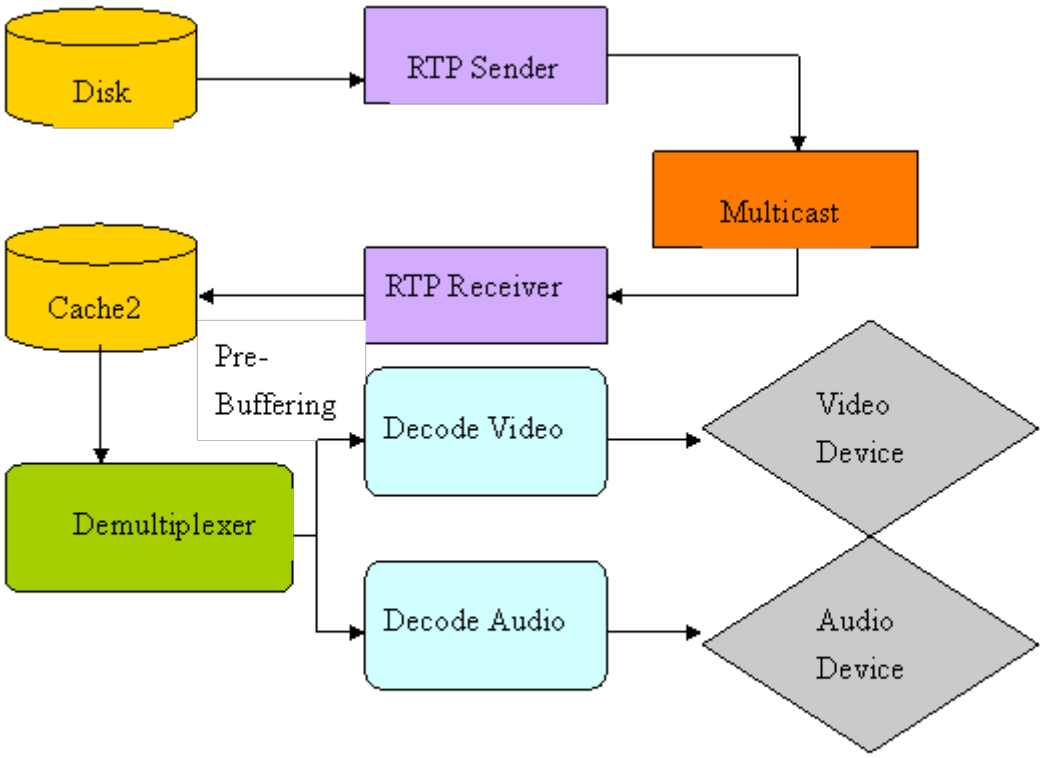


图 1-3 组播整体方案示意图

如图 1-3 所示：

RTPSender 为服务器端；

RTPReceiver 为客户端；

Multicast 为组播网络；

流数据包由 RTPSender 发送，经过 Multicast 网络，到达 RTPReceiver 客户端；

4．3.2 RTP 协议组播方案服务器端实现

服务器端的算法为：

1、打开设备，分配资源。当设备准备好时，创建一个 RTP 实时服务线程和一个 RTCP 实时服务线程，端口的选择是 RTP 数据在偶数 UDP 端口传输，RTCP 包在下一个高奇数端口传输；

2、创建一个 UDP 套接字并将其绑定到所提供服务的地址之上。

3、反复调用接收模块，接收来自客户的 RTCP 报告，根据其类型做出响应。

服务器端将音视频流封装成 RTP 数据包通过组播网络发送，如图 1-4：

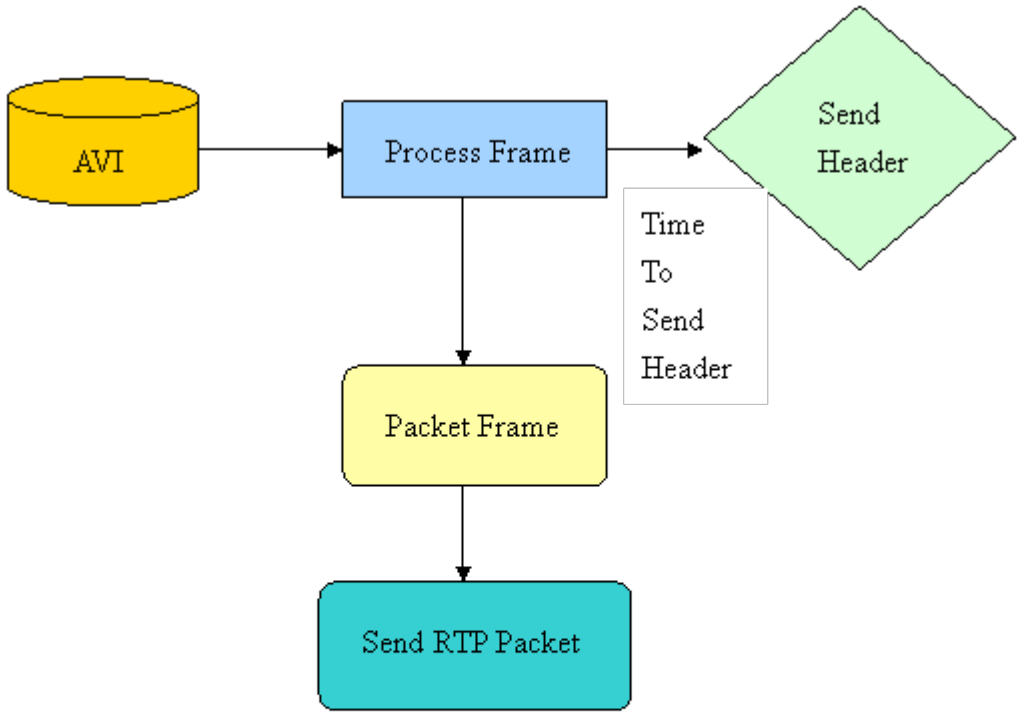
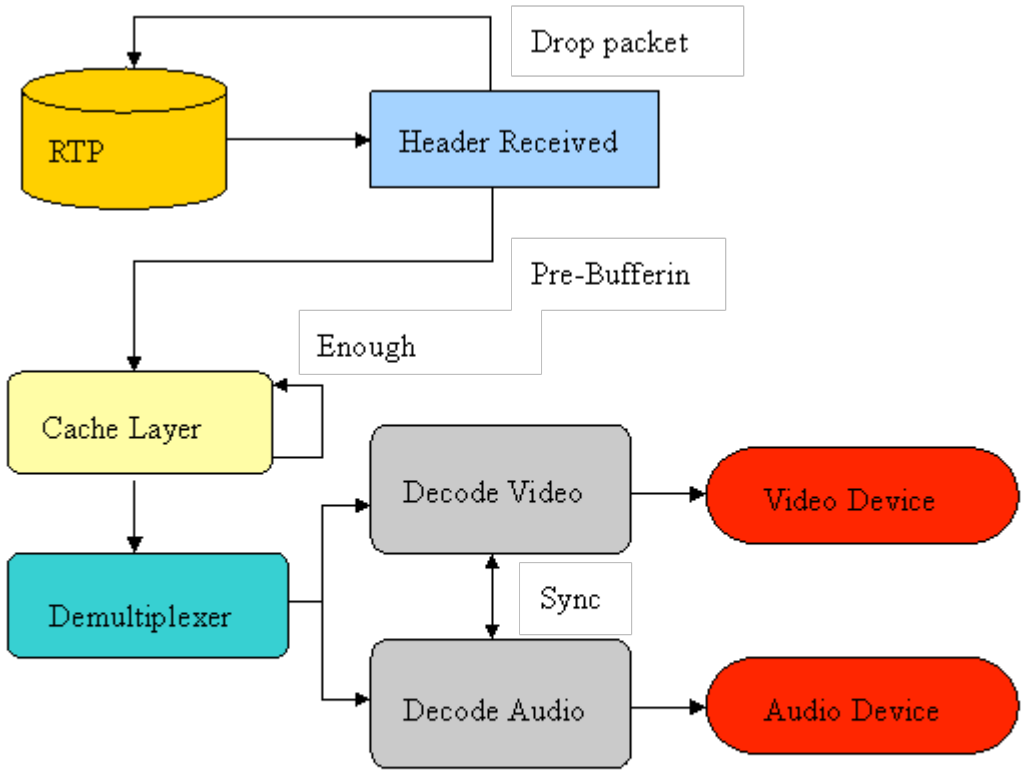


图 1-4 服务器发送示意图

4．3. 3RTP 协议组播方案客户端实现

客户端加入组播群组后，就可以接收音视频流，如图 1-5 所示：



如图 1-5　客户端接收示意图

4．3. 4RTP 协议视频帧率和质量调整策略

视频帧率和质量的动态调整的目的是在保证语音质量的情况下，不需要用户干预，而尽网络带宽的可能提高视频的帧率和质量；同时，在网络可用带宽下降时，自适应网络的带宽动态调整视频的帧率和质量，尽可能不出现马赛克，不影响音视频的传输。

判断网络带宽一般有两种方法：一是通过丢包率；二是通过 RTCP 包循环一圈的时间。丢包率的计算流程见本文 3.4 丢包率中的描述，但丢包率由于以下原因往往不能正确反映实现带宽的情况：

一是由于 UDP 包的接收顺序不能保证，接收到的数据包可能多于发送的数据包例如复制数据包，造成计算的丢包率不能正确反映网络带宽；另一个问题是 RTCP 的数据包可能丢失，如果发送方 RTCP 发送汇报包丢失，则接收方接收数据包计数不能正确清零，如接收方 RTCP 接收汇报包丢失，则发送方不能得到丢包率，因此也不能正确反映网络带宽。

因此一般采用 RTCP 包循环一圈的时间来判断网络带宽。但是 RTCP 包循环一圈的时间不能准确反映网络带宽，我们通过一些措施使之尽量准确：一是使用多个 RTCP 包的循环时间的平均值判断网络带宽，尽量避免网络的抖动情况；二是设置最大循环时间为 400ms，解决丢失 RTCP 包或网络突然发生抖动时对计算循环时间的平均值造成的影响。

视频调整的频率：首先使用一个比较低的帧率和质量的动态调整，然后根据网络带宽的情况逐步调整帧率和质量。一般在开始阶段（如 1 分钟以内）调整的频繁一些（比如 10 秒钟调整一次），这样调整的目的是使视频的质量和帧率尽快的适应带宽；然后保持一个固定的间隔调整一次（比如 1 分钟），使视频帧率和质量随着网络的变化不断调整。

RTP 现在还没有一个标准的视频帧率和质量调整算法，服务器根据自身的情况进行相应的调整，一般来说根据 RTCP 包循环时间的平均值计算出一个加权值，然后根据这个值所在的区间，并经过一定的处理后，对视频的帧率和质量分别进行相应的调整以满足需要。要得到一个明确的算法，需要我们做相应的测试实验来验证我们的算法；

五． RTP 协议移植计划

代码 Jrtplib-3.3.0 基本上实现了 RTP 协议的一个封装，我们后续要做的是第一对该代码进行一次全面的验证，以确保该代码的质量；

第二通过做测试实验来得到一个稳定可用的视频帧率和质量调整算法以满足网络实时传输的要求；

计划是先在 PC 上完成代码和算法的验证，然后再移植到 DVR 上；

六． RTP 协议安全方面考虑

RTP 安全方面的考虑：攻击者可能通过伪造源地址,目的地址,修改报头等方式来攻击 RTP,如果 RTP 是通过组播方式发送,那么发送者无法控制接收者的行为,即无法对接收者进行管理以达到安全上的要求,目前也有一些策略如加密方法来解决安全方面的问题,但还没有完全解决该问题,这里不再详细描述；