

0

史丹利部落格

倘若您認為「本網站」內容、圖案、超連結等侵害他人之著作權或不願意被連結，請留言通知站長，站長看到留言後會立即移除。

 搜尋

2007年4月4日 星期三

RTSP RTP RTCP

底下都是轉載的資料

RTSP：即時流協議（RTSP：Real Time Streaming Protocol）

即時流協議（RTSP）建立並控制一個或幾個時間同步的連續流媒體，如音頻和視頻。儘管連續媒體流與控制流交叉是可能的，RTSP 本身並不發送連續流。換言之，RTSP 充當多媒體伺服器的網路遠端控制。RTSP 提供了一個可擴展框架，實現即時資料（如音頻與視頻）的受控、按需傳送。資料源包括實況資料與存儲的剪輯。RTSP 用於控制多個資料發送會話，提供了選擇發送通道（如 UDP、組播 UDP 與 TCP 等）的方式，並提供了選擇基於 RTP 的發送機制的方法。

目前還沒有 RTSP 連接的概念；伺服器維護由識別符標識的會話。RTSP 會話不會綁定到傳輸層連接，如 TCP。在 RTSP 會話期間，RTSP 用戶端可打開或關閉多個對伺服器的可靠傳輸連接以發出 RTSP 請求。它也可選擇使用無連接傳輸協定，如 UDP。

RTSP 控制的流可能用到 RTP，但 RTSP 操作並不依賴用於傳輸連續媒體的傳輸機制。RTSP 在語法和操作上與 HTTP/1.1 類似，因此 HTTP 的擴展機制在多數情況下可加入 RTSP。然而，在很多重要方面 RTSP 仍不同於 HTTP：

- RTSP 引入了大量新方法並具有一個不同的協議識別字：
- 在大多數情況下，RTSP 伺服器需要保持缺省狀態，與 HTTP 的無狀態相對；
- RTSP 中用戶端和伺服器都可以發出請求；
- 在多數情況下，資料由不同的協定傳輸；
- RTSP 使用 ISO 10646（UTF-8）而並非 ISO 8859-1，與當前的國際標準 HTML 相一致；
- URI 請求總是包含絕對 URI。為了與過去的錯誤相互相容，HTTP/1.1 只在請求過程中傳送絕對路徑並將主機名置於另外的頭欄位。

該協定支援如下操作：

- 從媒體伺服器上檢索媒體：用戶可通過 HTTP 或其他方法提交一個演示描述請求；
- 媒體伺服器邀請進入會議：媒體伺服器可被邀請參加正進行的會議，或重播媒體，或記錄部分或全部演示；
- 將新媒體加到現有演示中：如伺服器能告訴用戶端接下來可用的媒體內容，對現場直播顯得尤其有用。

協定結構

RTSP 是一種文本協定，採用 UTF-8 編碼中的 ISO 10646 字元集。一行可通過 CRLF 終止，但接收端需要做好解釋 CR 和 LF 作為一行終止符的準備。關於頭欄位概述如下：

Header Type Support Methods

Accept R opt. entity

BloggerAd

光泉調味乳
開盒立即送電影票

開盒即送馬達加斯加
電影套票，登入發票
加坡環球影城三天兩

讚

1.5

ADSENSE

Accept-Encoding R opt. entity
 Accept-Language R opt. all
 Allow R opt. all
 Authorization R opt. all
 Bandwidth R opt. all
 Blocksize R opt. All but OPTIONS, TEARDOWN
 Cache-Control G opt. SETUP
 Conference R opt. SETUP
 Connection G req. all
 Content-Base E opt. entity
 Content-Encoding E req. SET_PARAMETER
 Content-Encoding E req. DESCRIBE, ANNOUNCE
 Content-Language E req. DESCRIBE, ANNOUNCE
 Content-Length E req. SET_PARAMETER, ANNOUNCE
 Content-Length E req. entity
 Content-Location E opt. entity
 Content-Type E req. SET_PARAMETER, ANNOUNCE
 Content-Type R req. entity
 CSeq G req. all
 Date G opt. all
 Expires E opt. DESCRIBE, ANNOUNCE
 From R opt. all
 If-Modified-Since R opt. DESCRIBE, SETUP
 Last-Modified E opt. entity
 Proxy-Authenticate
 Proxy-Require R req. all
 Public R opt. all
 Range R opt. PLAY, PAUSE, RECORD
 Range R opt. PLAY, PAUSE, RECORD
 Referer R opt. all
 Require R req. all
 Retry-After R opt. all
 RTP-Info R req. PLAY
 Scale Rr opt. PLAY, RECORD
 Session Rr req. All but SETUP, OPTIONS
 Server R opt. all
 Speed Rr opt. PLAY
 Transport Rr req. SETUP
 Unsupported R req. all
 User-Agent R opt. all
 Via G opt. all
 WWW-Authenticate R opt. all

類型 "g" 表示請求和回應中的通用請求頭；類型 "R" 表示請求頭；類型 "r" 表示回應頭；類型 "e" 表示實體頭欄位。在 "support" 一欄中 標有 "req." 的欄位 必須由接收者以特殊的方法實現；而 "opt." 的欄位是可選的。注意，不是所有 "req." 欄位在該類型的每個請求中都會被發送。

"req." 只表示客戶機（支持回應頭）和伺服器（支持請求頭）必須執行該欄位。最後一欄列出了關於頭欄位產生作用的方法；其中 "entity" 針對於返回一個資訊主體的所有方法。

<http://www.javvin.com/protocol/rfc2326.pdf>

頻點播通信的基礎是RTP和RTSP；RTP是下行傳輸的流協定，RTSP是針對資料的控制協定，兩者都容許在實現中建立極大的靈活性。

RTP的傳輸層是使用UDP(User Data Protocol)而非TCP。UDP在資料遞送方



請選取語言

由「Google 翻譯」技術

中原 98 碩專班

中原資工系

中原iLearning

中原e點靈(itouch)

中原圖書館

書報討論BBS

班版BBS

張耀仁-寬頻無線通訊
網路

文章分類

Android (14)

ARM (11)

BIOS (10)

bluespec (1)

面，會比TCP快速且有效率，因此可有效避免延遲現象。可是UDP缺乏回報資料遺失的機制，所以在網際網路或無線網路串流中，幾乎都會有資料遺失的情況，造成品質下降。另外，大部份公司和企業的防火牆都會檔掉UDP，所以在防火牆內是無法接收經由UDP遞送的串流。想要在有防火牆的情況下使用RTP，必須使用HTTP tunneling 技術，亦即將RTP封包包裹在HTTP封包內，以方便通過防火牆。不幸地，HTTP tunneling 會增加許多額外的資料，佔用更多的頻寬。RTP又可搭配RTCP(Real Time Control Protocol)與RTSP(Real Time Streaming Protocol)。RTCP可自動偵測現在的網路頻寬。RTSP支援伺服器與播放器雙向溝通，使用者可以透過RTSP下指令給伺服器作像是「暫停」、「快轉」、「倒帶」、「跳到下一章」等動作。

即時運輸協議 RTP (Real-time Transport Protocol)

RTP 為即時應用提供端到端的運輸，但不提供任何服務品質的保證。多媒體資料塊經壓縮編碼處理後，先送給 RTP 封裝成為 RTP 分組，再裝入運輸層的 UDP 用戶資料報，然後再交給 IP 層。
RTP 是一個協議框架，只包含了即時應用的一些共同的功能。
RTP 自己並不對多媒體資料塊做任何處理，而只是向應用層提供一些附加的資訊，讓應用層知道應當如何進行處理。

RTP 的層次

從應用開發者的角度看，RTP 應當是應用層的一部分。
在應用的發送端，開發者必須編寫用 RTP 封裝分組的程式碼，然後把 RTP 分組交給 UDP 插口介面。
在接收端，RTP 分組通過 UDP 插口介面進入應用層後，還要利用開發者編寫的程式碼從 RTP 分組中把應用資料塊提取出來。

RTP 也可看成是運輸層的一個子層

RTP 封裝了多媒體應用的資料塊。由於 RTP 向多媒體應用程式提供了服務（如時間戳和序號），因此也可以將 RTP 看成是在 UDP 之上的一個運輸層的子層。

- 運輸層
- 應用層
- IP
- 資料連結層
- 物理層
- RTP
- UDP

RTP 分組的首部格式

- 12
- 位元組
- 序 號
- 比特 0 1 3 8 16 31
- 有效載荷類型
- 版本
- P
- X
- M

- Bootloader (5)
- C 語言 (14)
- C# (1)
- CSS (2)
- DoCoMo (5)
- ESL (3)
- Google (1)
- GPL (1)
- Graphic (1)
- jpeg (1)
- kernel (16)
- LLVM (2)
- mangos (8)
- MID (1)
- MPEG (1)
- mysql (1)
- OpenSourceTool (1)
- Protocol (3)
- socket (9)
- SystemC (9)
- ubuntu (3)
- USB (3)
- verilog (11)
- WindowsTool (2)
- x86 (1)
- 定義 (6)
- 專案管理 (4)
- 平行處理 (2)
- 影音 (24)
- 感測器 (2)
- 投資 (8)
- 理財 (5)
- 生活 (6)
- 經典c程式100例 (10)
- 英文 (9)
- 計算機組織 (6)
- 論文 (1)
- 雜項 (5)
- 霹靂布袋戲 (9)

我的最愛

- 何圓圓的媽咪
- Jack Homework Blog
- 討論KGDB網站
- 查爾斯的天空
- 內核之旅
- 菠蘿麵包的家

參與源數

時間戳

同步源標識符 (SSRC)

參與源標識符 (CSRC) [0..15]

...

發送

RTP 分組

UDP 用戶資料報

IP 數據報

IP 首部 UDP 首部 RTP 首部 RTP 資料部分 (應用層資料)

即時傳輸控制協議 RTCP (RTP Control Protocol)

RTCP 是與 RTP 配合使用的協定。

RTCP 協議的主要功能是：服務品質的監視與回饋、媒體間的同步，以及多播組中成員的標識。

RTCP 分組也使用 UDP 傳送，但 RTCP 並不對聲音或視像分組進行封裝。可將多個 RTCP 分組封裝在一個 UDP 用戶資料報中。

RTCP 分組週期性地在網上传送，它帶有發送端和接收端對服務品質的統計資訊報告。

RTCP 使用的五種分組類型

結束分組 BYE 表示關閉一個資料流程。

特定應用分組 APP 使應用程式能夠定義新的分組類型。

接收端報告分組 RR 用來使接收端週期性地向所有的點用多播方式進行報告。

發送端報告分組 SR 用來使發送端週期性地向所有接收端用多播方式進行報告。

源點描述分組 SDES 給出會話中參加者的描述。

即時傳輸控制協定rtcp協定

1. rtcp協議：

rtcp (real-time transpor、control protocol) 是設計和rtp一起使用的進行流量控制和擁塞控制的服務控制協定。

2. rtcp協議如何工作：

當應用程式開始一個rtp會話時將使用兩個埠：一個給rtp，一個給rtcp。rtp本身並不能為按順序傳送資料包提供可靠的傳送機制，也不提供流量控制或擁塞控制，它依靠rtcp提供這些服務。在rtp的會話之間週期的發放一些rtcp包以用來傳監聽服務品質和交換會話用戶資訊等功能。rtcp包中含有已發送的資料包的數量、丟失的資料包的數量等統計資料。因此，伺服器可以利用這些資訊動態地改變傳輸速率，甚至改變有效載荷類型。rtp和rtcp配合使用，它們能以有效的回饋和最小的開銷使傳輸效率最佳化，因而特別適合傳送網上的即時資料。根據用戶間的資料傳輸回饋資訊，可以制定流量控制的策略，而會話用戶資訊的交互，可以制定會話控制的策略。

rtcp協定處理機根據需要定義了五種類型的報文——

rr: receiver report

sr: sender report

sdes: source description items.

bye: indicates end of participation.

app: application specific functions

翔惠葦淝Family

空中英語教室

本益比排行榜

lily的blog

ann的blog

ESL-LAB

ELLLO免費英文聽力練習

MaoYang-有話要說

南華大學-課程影音系統

民視英文新聞

圖解英文字典[only IE]

JServ 黃敬群

KATDC

Jollen網路學院

小吳寶寶家

十字佣兵

MangosProject

黑石論壇

VC知識庫在線雜誌

CSDN中國IT社區

LinuxSir.Org

Linux相關文章

中國Linux論壇

The Code Project

Codeguru

The News for Linux

台灣宏觀英語新聞台-每日更新一次

為公寫作網

LinuxJournal

Acrobat

Format My Source

Code For Blogging

中原資工ESL

網誌存檔

► 2012 (3)

► 2011 (19)

► 2010 (14)

► 2009 (30)

► 2008 (124)

▼ 2007 (45)

► 六月 (1)

► 五月 (5)

它們完成接收、分析、產生和發送控制報文的功能。

即時流式協議RTSP (Real-Time Streaming Protocol)

RTSP 協定以客戶伺服器方式工作，它是一個多媒體播放控制協定，用來使用戶在播放從網際網路下載的即時資料時能夠進行控制，如：暫停/繼續、後退、前進等。因此 RTSP 又稱為“網際網路錄影機遙控協議”。要實現 RTSP 的控制功能，我們不僅要有協議，而且要有專門的媒體播放器(media player)和媒體伺服器(media server)。

流式(streaming)音頻和視頻

媒體伺服器與媒體播放器的關係是伺服器與客戶的關係。

媒體伺服器與普通的萬維網伺服器的最大區別就是媒體伺服器支援流式音頻和視頻的傳送，因而在用戶端的媒體播放器可以邊下載邊播放（當然需要將節目存儲一小段時間）。

但從普通萬維網伺服器下載多媒體節目時，是先將整個檔下載完畢，然後再進行播放。

RTSP 與 RTP 和 RTCP 的關係

RTSP播放器

RTSP伺服器

RTSP 控制分組（TCP）

RTP 資料分組（UDP）

RTCP 分組（UDP）

客戶

伺服器

RTSP 僅僅是使媒體播放器能控制多媒體流的傳送。因此，RTSP 又稱為帶外協定，而多媒體流是使用 RTP 在帶內傳送的。

一個基於Darwin Streaming Server的linux RTSP Server,也有一個小型的RTSP Client Source

<http://folk.uio.no/meccano/reflector/>

```
mplayer -noframedrop -dumpfile out.rm -dumpstream
rtsp://url/to/file.rm
```

即時傳輸協議 RTP

RTP（Real-timeTransportProtocol）是用於Internet上針對多媒體資料流程的一種傳輸協定。RTP被定義為在一對一或一對多的傳輸情況下工作，其目的是提供時間資訊和實現流同步。RTP通常使用UDP來傳送資料，但RTP也可以在TCP或ATM等其他協議之上工作。當應用程式開始一個RTP會話時將使用兩個埠：一個給RTP，一個給RTCP。RTP本身並不能為按順序傳送資料包提供可靠的傳送機制，也不提供流量控制或擁塞控制，它依靠RTCP提供這些服務。通常RTP演算法並不作為一個獨立的網路層來實現，而是作為應用程式碼的一部分。即時傳輸控制協議RTCP。

RTCP(Real-timeTransportControlProtocol)和RTP一起提供流量控制和擁

▼ 四月 (39)

C#面試題

C/C++語言struct 深層探索

NTT DoCoMo Vision 2010 - Old School Friends

楊宗緯 人質@超 級星光大道 2007.4.6

mangos ZG技能 資料

女兒學會了"嘖"

mangos 奧妮克 希亞的AI

mangos 2.0.10 patch

時間=金錢,智識= 力量

有趣的計算

SUN公司的考試 題目

有趣的C程式

電腦開機流程

變數的資料型態 種類

深入淺出 Hello World Part

快速修復資料庫 的方法

C語言巨集定義 技巧

Linux下的即時流 媒體編程

RTSP RTP RTCP

mangos DBC File Format

編程秘笈篇：C 語言高效編程 的四大絕招

A*尋徑演算法

C語言嵌入式系 統編程修煉之 記憶體操作

C語言嵌入式系 統編程修煉之 六:性能優化

經典c程式100例

塞控制服務。在RTP會話期間，各參與者週期性地傳送RTCP包。RTCP包中含有已發送的資料包的數量、丟失的資料包的數量等統計資料，因此，伺服器可以利用這些資訊動態地改變傳輸速率，甚至改變有效載荷類型。RTP和RTCP配合使用，它們能以有效的回饋和最小的開銷使傳輸效率最佳化，因而特別適合傳送網上的即時資料。

5. RTP資料傳輸協議

RTP提供端對端網路傳輸功能，適合通過組播和點播傳送即時資料，如視頻、音頻和仿真資料。RTP沒有涉及資源預訂和品質保證等即時服務，RTCP擴充資料傳輸以允許監控資料傳送，提供最小的控制和識別功能。RTP與RTCP設計成獨立於傳輸和網路層。

5.1 RTP固定頭

RTP 頭格式如下：



開始12個八進制出現在每個RTP包中，而CSRC標識列表僅出現在混合器插入時。

5.2 複用 RTP 連接

為使協定有效運行，複用點數目應減至最小。RTP中，複用由定義RTP連接的目的傳輸位址（網路位址與埠號）提供。例如，對音頻和視頻單獨編碼的遠端會議，每個媒介被攜帶在單獨RTP連接中，具有各自的目的傳輸位址。目標不在將音頻和視頻放在單一RTP連接中，而根據SSRC段載荷類型進行多路分解。使用同一SSRC，而具有不同載荷類型的交叉包將帶來幾個問題：

如一種載荷類型在連接期間切換，沒有辦法識別新值將替換那一個舊值。

SSRC定義成用於標識單個計時和系列號空間。如媒體時鐘速率不同，而要求不同系列號空間以說明那種載荷類型有丟包，交叉複用載荷類型將需要不同計時空間。

RTCP發送和接收報告可能僅描述每個SSRC的計時和系列號空間，而不攜帶載荷類型段。

RTP混合器不能將不相容媒體流合併成一個流。

在一個RTP連接中攜帶多個媒介阻止幾件事：使用不同網路路徑或網路資源分配；接受媒介子集。

對每種媒介使用不同SSRC，但以相同RTP連接發送可避免前三個問題，但不能避免後兩個問題。

<http://man.chinaunix.net/develop/rfc/RFC2959.txt>即時傳輸協定管理資訊庫

5.3 對RTP頭特定設置profile的修改

可以認為，現用RTP資料包頭對RTP支援的所有應用類共同需要的功能

91-100

經典c程式100例
81-90

經典c程式100例
71-80

經典c程式100例
61-70

經典c程式100例
51-60

經典c程式100例
41-50

經典c程式100例
31-40

經典c程式100例
21-30

經典c程式100例
11-20

經典c程式100例
1-10

Embedded C
Language
Porblems

mangos 恐懼的
修正

mangos 解決怪
物陣營問題

mangos 傳送寶
石[整理版]

Mangos 簡介

關於我自己

史丹利

檢視我的完整簡介



集是完整的。然而，為了維持ALF設計原則，頭可通過改變或增加設置來裁剪，並仍允許設置無關監控和記錄工具起作用。

* 標記位元與載荷類型段攜帶特定設置資訊，但由於很多應用需要它們，或者要容納它們，就要增加另外32位字，故允許分配在固定頭中。包含這些段的八進制可通過設置重新定義以適應不同要求，如採用更多或更少標記位元。如有標記位元，既然設置無關監控器能觀察包丟失模式和標記位元間關係，我們就可以定位八進制中最重要的位。

* 其他特殊載荷格式（視頻編碼）所要求的資訊應該攜帶在包的載荷部分。可出現在頭，總是在載荷部分開始處，或在資料模式的保留值中指出。

* 如特殊應用類需要獨立載荷格式的附加功能，應用運行的設置應該定義附加固定段跟隨在現存固定頭SSRC之後。這些應用將能迅速而直接訪問附加段，同時，與監控器和記錄器無關設置仍能通過僅解釋開始12個八進制處理RTP包。如證實附加功能是所有設置共同需要的，新版本RTP應該對固定頭作出明確改變。

6. RTP控制協議-- RTCP

RTCP協定將控制包週期發送給所有連接者，應用與資料包相同的分發機制。低層協定提供資料與控制包的複用，如UDP分別給兩者提供單獨的埠。

RTCP執行下列四大功能：

1. 主要是提供資料分發的品質回饋。是作為RTP傳輸協議的一部分，與其他傳輸協議的流控制和阻塞控制有關。回饋對自適應編碼控制直接起作用，但IP組播經驗表明，從發送者收到回饋對診斷發送錯誤也是致關重要的。給所有參加者發送接收回饋報告，允許問題觀察者估計那些問題是局部的，還是全局的。諸如IP組播等發佈機制可能會使一個如網路服務提供商等團體接收回饋信息，充當第三方監控者來診斷網路問題。回饋功能由RTCP發送者和接收者報告執行。

2. 對應於一個RTP源，RTCP攜帶一個稱作規範名字（CNAME）的固定的傳輸層標識。既然SSRC標識可改變，所以如果發現衝突，或程式重新啟動，接收者需要用CNAME跟蹤每個參加者（即CNAME也會改變）。接收者也需要CNAME 與相關RTP連接中同一個給定的連接者的幾個資料流程進行聯繫，比如用來同步音頻流和視頻流。媒體流的同步也要求發送者把NTP和RTP的時間戳包含到RTCP包中。

3. 前兩種功能要求所有參加者發送RTCP包，因此，為了RTP連接的參加者擴展到大規模數量，速率必須受到控制。通過讓每個參加者給其他參加者發送控制包，參加者就能獨立的觀察到參加者的數量。該數量用來計算每個包發送的速率。

4. 第四個可選功能是傳送最小連接控制資訊，如參加者辨識。最可能用在"鬆散控制"連接，那裏參加者可以的自由進入或離開，沒有成員控制或參數協調，RTCP充當通往所有參加者的方便通道，但不必支援應用的所有控制通訊要求。高級連接控制協議超出本書範圍。

在IP組播場合應用RTP時，前3個功能是必須的，推薦用於所有情形。RTP應用設計人員必須避免使用僅在單播模式下工作的機制，那將導致無法擴展規模。

6.1 RTCP 包格式

下面定義幾個攜帶不同控制資訊的RTCP包類型：

SR：

發送方報告，來自當前活動發送者的發送、接收情況的統計資訊。

RR：

接收方報告，來自非活動的發送者的對接收情況的統計資訊。接收方報告與來自活動的發送者的SR相聯繫，這些發送者可以報告超過31個源（的資訊）。

SDES :

源描述項，包括**CNAME**。

BYE :

表示一個參與的結束。

APP :

特定應用函數。

類似于**RTP**資料包的開頭部分，每個**RTCP**包以固定部分開始，緊接著的是可變長結構元素，但必須以一個**32**位邊界結束。為了使**RTCP**包可堆疊，需要包含佇列要求和固定部分中的一個長度段。不需要插入任何分隔符號可以直接將多個**RTCP**包連接起來，形成一個**RTCP**組合包，然後以單一包形式在低層協議發送。由於需要低層協定提供整體長度來決定組合包的結尾，所以在組合包中沒有單個**RTCP**包顯式計數。

組合包中每個**RTCP**包可獨立處理，不需要根據包組合順序。但未了執行協議功能，強加如下約束：

- *接收統計（在**SR**或**RR**中）應該經常發送，只要帶寬允許，因此每個週期發送的組合**RTCP**包應包含報告包。

- *新接收者需要接收**CNAME**，並儘快識別源，開始聯繫媒介進行同步，因此每個包應該包含**SDES CNAME**。

- *出現在組合包前面的是包類型數量，其增長應該受到限制，以提高常數位數量，提高成功驗證**RTCP**包對含錯誤位址**RTP**資料包或其他無關包的概率。

因此，所有**RTCP**包必須以至少兩個**RTCP**包組合形式發送，推薦格式如下：

加密首碼（**Encryption prefix**）：

僅當組合包被加密，才需要加上一個**32**位元亂數用於每個組合包發送。

SR或RR :

組合包中第一個**RTCP**包必須（如附錄**A2**中）總為一個報告包，方便頭的確認。即使沒有資料發送，也沒有接收到資料，也要發送一個空**RR**，那怕組合包中**RTCP**包為**BYE**。

附加**RR :**

如果報告統計的源的數目超過**31**，那麼在初始報告包後應該有附加的**RR**包。

SDES :

包含**CNAME**項的**SDES**包必須包含在每個組合**RTCP**包中。如果特定的應用程式要求的話，其他源描述項可選，但受到帶寬限制。

BYE或APP :

其他**RTCP**包類型可以任意順序排列，除了攜帶特定的**SSRC/CSRC**的**BYE**應作為最後一個包發送，包類型出現可不止一次。

為了每個參加者的帶寬能夠被正確的估計，一個獨立的**RTP**參加者在每個報告間隔應該只發送一個組合**RTCP**，除了在**9.1**中說的對每個本分加密後的組合**RTCP**包。如果太多的源把所有必要的**RR**包都裝進一個組合包中，且沒有超過網路中的最大傳輸單元，那麼裝到一個**MTU**中的子集應該在每個間隔中都被包含進來。這些子集應該通過多個時間間隔迴圈地被選擇，這樣所有的源都會被報告到。

為了分攤包中過大部分，建議轉換器或混合器從多個源組合單個**RTCP**包，不管合不合適。如組合包整體長度超過網路路徑最大傳輸單元，可分成多個較短組合包用低層協定以單個包形式發送。這不會削弱**RTCP**帶寬的估計能力。注意，每個組合包必須以**SR**或**RR**包開始。

一個執行程式應該忽略掉攜帶未知類型的來到的RTCP包。而附加RTCP包類型可在Internet Assigned Numbers Authority (IANA)處註冊。

```

if encrypted: random 32-bit integer
|
|[------ packet -----][------ packet -----][-packet-]
|
| receiver chunk chunk
V reports item item item item
-----
R[SR #sendinfo #site1#site2][SDES #CNAME PHONE #CNAME LOC]
[BYE##why]
-----
| |
|<----- compound packet ----->|
|<----- UDP packet ----->|

#: SSRC/CSRC identifier

```

Figure 1: Example of an RTCP compound packet

6.2 RTCP傳輸間隔

RTP設計成允許應用程式自動按量決定連接數，範圍可從幾個到上千個。例如，音頻會議中，資料流程量是內在限制的，因為同一時刻只有一兩個人說話；對組播，給定連接資料率仍是常數，獨立於連接數，但控制流量不是內在限制的。如果每個參加者以固定速率發送接收報告，那麼控制流量將隨參加者數量線性增長，因此，速率必須按比例下降，通過動態的計算RTCP包傳輸的時間間隔來完成。

對每個會話來說，資料流程量受到一個叫做連接帶寬的集合的限制，在參加者之間被分塊。這個帶寬由網路保留並受網路的限制。如果沒有保留值，那麼會有其他基於環境的限制。這些限制建立合理的最大值給這個連接使用，同時成為連接的帶寬。這個連接的帶寬的選擇基於一些開銷或是這個連接對應的可以獲得的一個預先知道的網路帶寬。它有時可以獨立於多媒體編碼，但是編碼選項可能受到這個連接帶寬的限制。這個連接帶寬經常是發送者期望的名義上的併發時的帶寬。對於分層編碼，每一層是一個擁有自己連接帶寬參數的獨立RTP連接。

當這個連接發起一個多媒體應用程式的時候，這個連接參數希望被被一個連接管理應用程式使用。但是多媒體應用程式可能要為連接選擇的編碼設置一個默認的基於單個發送者的資料帶寬。這個應用程式也可能給這個帶寬強加機遇多播範圍的規則或是其他標準。所有的參加者需要使用同樣的連接帶寬值來計算相同的RTCP時間間隔。

控制流還是資料流程的帶寬的計算都要包含低層的傳輸層和網路層協定（如UDP、IP），那是正是源保留值需要知道的东西。這個應用程式也期望知道哪個協定在被使用。連路層的頭不包含在計算中，因為這個包在傳輸時用不同的連路層頭來進行封裝。

一旦確認位址有效，如後來標記成未活動，位址的狀態應仍保留，位址應繼續計入共用RTCP帶寬地址的總數中，時間要保證能掃描典型網路分區，建議為30分鐘。注意，這仍大於RTCP報告間隔最大值的五倍。

這個規範定義了除必需的CNAME外的幾個源描述項，如NAME（人名）和EMAIL（電子郵件位址）。它也為定義新的特定應用程式的RTCP包類型的途徑。給附加資訊分配控制帶寬應引起注意，因為它將降低接收報告和CNAME發送的速率而損害協定的性能。建議分配給單個參加者用於攜帶附

加資訊的RTCP帶寬不要超過20%。而且並沒有有意讓所有SDES項包含在每個應用中。

6.2.1 獲得連接成員個數

用一個表中的SSRC和CSRC來區分，是否有新的連接到來；在包的時間間隔中計算有沒新的到達，或者是有哪個變成非活動的了

6.3 RTCP包的發送和接收規則

允許在多播和多點點播環境中進行操作的的一個執行程式必須滿足6.2部分的要求。

執行這些規則，參加者要通過一些狀態來：

tp：一個包的最後時間

tn：一個包的下個調節傳輸時間

tc, pmember:已經在的參加者數目，member：估計

rtcp_bw:帶寬

we_sent:前面的二個包的報告到時為真

avg_rtcp_size:所有組合包的平均長度。

Initial:還沒發出包的標誌位元

很多規則指定都是為了統計時間間隔。

6.3.1 計算RTCP傳輸的時間間隔

要用一個組長度.T有以下幾點決定：

1. 小等於25%的成員參加連接時，T取決與這些參加者是不是發送方。是發送方， $we_send = true$, $avg_trcp_size = Constant$ 發送者數目，被 $rtcp_bw = 0.25$ 除
不是， $false$ 0.75
the 25% fraction becomes $S/(S+R)$ and the 75% fraction becomes $R/(S+R)$.

2. 一個連接還沒發送包（initial=true）時， $T_{min} = 2.5s$ 否則=5s

3. 確定值Td設置為最大

4. 估計時間0.5，1.5倍的Td

5. T除以1.21828來補償帶寬轉換時的演算法來的值

時間間隔隨機分配，時間25%分配給發送者，75%給接收者

6.3.2 初始化

$Tp=0$, $tc=0$, $sender=0$, $pmember=1$, $members=1$, $we_sent=false$,
 $rtcp_bw$ =連接帶寬的確定部分， $initial=true$, avg_rtcp_size =第一個包的可能長度， $tn=T$,

6.3.3 接收一個RTP或者一個非BYE RTCP包

接到含不在成員表格中的SSRC的RTCP或是RTP包的時候，添加到表中，並更新member的數值

不在成員表中的RTP 包時，添加，並更新senders

收到組合包時， $avg_rtcp_size = (1/16) * packet_size + (15/16) * avg_rtcp_size$

Packet_size 為剛剛收到的組合包的長度

6.3.4 接收一個RTCP BYE 包

收到RTCP BYE，先檢查成員表格中的SSRC，存在這個包，則從表中移除更新members

在檢查發送者表格的SSRC，出現成員，移除，更新senders

同時要考慮其他數值如tn tc

$tn = tc + (members/pmembers) * (tn - tc)$

$tp = tc - (members / pmembers) * (tc - tp).$

6.3.5 一個SSRC超時

另外的一些連接超時的時候，這個連接就要檢查臨時的時間間隔及
Tdeteministic, we_send=false. >Mtd=5s 為超時

6.3.6 計時器終止（滿）

* 獲得時間間隔T

* $tp + T < tc$, 表RTCP包發了，設置 $tp=tc$, 時間間隔也變

$tp + T \geq tc$, 沒有RTCP包在發， $tn=tp+T$

□ Pmembers=members

有RTCP包在傳，initial=false, $avg_rtcp_size = (1/16) * packet_size + (15/16) * avg_rtcp_size$

6.3.7 傳一個 BYE包

6.3.8 更新 we_sent

是否發RTP包

6.3.9 源描述字的帶寬分配

SDS定義NAME，EMAIL等非強制性的

6. 4 發送者與接收者報告

RTTP接收者使用RTCP報告包提供接收品質回饋，報告包根據接收者是否也是一個發送者而採用兩種格式中的一種。除包類型代碼外，發送者報告與接收者報告間唯一的差別是發送者報告包含一個20個位元組發送者資訊段。如果某位址（源）在發出最後或前一個報告的時間間隔中發送資料包，就發佈SR；否則，就發出RR。SR和RR都可以沒有或包括多個接收報告塊。對於同步源中的每個源都一樣。自從最後一個報告後，這個接收者都是接收RTP資料包發佈報告不是為列在CSRC列表上的起作用的源，每個接收報告塊提供從特殊源接收資料的統計。既然最大可有31個接收報告塊嵌入在SR 或 RR包中，丟失包累計數差別給出間隔期間丟掉的數量，而所收到擴展的最後一個系列號的差別給出間隔期間希望發送的包數量，兩者之比等於經過間隔期間包丟失百分比。如兩報告連續，比值應該等於丟失段部分；否則，就不等。每秒包丟失率可通過NTP時標差除以丟失部分得到。

從發送者資訊，第三方監控器可計算載荷平均數據速率與沒收到資料間隔的平均包速率，兩者比值給出平均載荷大小。如假設包丟失與包大小無關，那麼特殊接收者收到的包數量給出此接收者收到的表觀流量。

6. 5 SDS: 源描述RTCP包

SDS 包為三層結構，由頭與資料塊組成，資料塊可以沒有，也可有多個，組成項描述塊所表明的源。項描述如下：

版本（V）、填充（P）、長度：

如SR包中所描述。

包類型（PT）：

8位，包含常數202，識別RTCP SDS包。

源計數（SC）：

5位，包含在SDS包中的SSRC/CSRC塊數量，零值有效，但沒有意義。

源描述項內容如下：

6.5.1 CNAME: 規範終端標識SDS項

CNAME標識屬性如下：

如發生衝突或重啟程式，由於隨機分配的SSRC標識可能發生變化，需要CNAME項提供從SSRC標識到仍為常量的源標識的綁定。

象SSRC標識，CNAME標識在RTP連接的所有參加者中應是唯一的。

為了提供一套相關RTP連接中某個參加者所採用的跨多媒體工具間的綁定，**CNAME**應固定為那個參加者。

為方便第三方監控，**CNAME**應適合程式或人員定位源。

6.5.2 NAME：用戶名稱SDS項

這是用於描述源的真正的名稱，如"John Doe, Bit Recycler, Megacorp"，可是用戶想要的任意形式。對諸如會議應用，這種名稱也許是參加者列表顯示最適宜的形式，它將是除**CNAME**外發送最頻繁的項目。設置可建立這樣的優先順序別。**NAME**值至少在連接期間仍希望保持為常數。它不該成為連接的所有參加者中唯一依賴。

6.5.3 EMAIL：電子郵件位址SDS項

郵件地址格式由RFC822規定，如"John.Doe@megacorp.com"。連接期間，電子郵件仍希望保持為常數。

6.5.4 PHONE：電話號碼SDS項

電話號碼應帶有加號，代替國際接入代碼，如"+1 908 555 1212"即為美國電話號碼。

6.5.5 LOC：用戶地理位置SDS項

根據應用，此項具有不同程度的細節。對會議應用，字串如"Murray Hill, New Jersey"就足夠了。然而，對活動標記系統，字串如"Room 2A244, AT&T BL MH"也許就適用。細節留給實施或用戶，但格式和內容可用設置指示。在連接期間，除移動主機外，**LOC**值期望仍保留為常數。

6.5.6 TOOL：應用或工具名稱SDS項

是一個字串，表示產生流的應用的名稱與版本，如"videotool 1.2"。這部分資訊對調試很有用，類似於郵件或郵件系統版本**SMTP**頭。**TOOL**值在連接期間仍保持常數。

6.5.7 NOTE：通知/狀態SDS項

該項的推薦語法如下所述，但這些或其他語法可在設置中顯式定義。**NOTE** 項旨在描述源當前狀態的過渡資訊，如"on the phone, can't talk"，或在講座期間用於傳送談話的題目。它應該只用於攜帶例外資訊，而不應包含在全部參加者中，因為這將降低接收報告和**CNAME**發送的速度，因此損害協定的性能。特殊情況下，它不應作為用戶設置檔的專案，也不是自動產生。

當其為活動時，由於**NOTE**項對顯示很重要，其他非**CNAME**項（如**NAME**）傳輸速率將會降低，結果使**NOTE**項佔用RTCP部分帶寬。若過渡資訊不活躍，**NOTE**項繼續以同樣的速度重複發送幾次，但以一個串長為零的字元串通知接收者。然而，如對小倍數的重複或約20-30 RTCP間隔也沒有接收到，接收者也應該考慮**NOTE**項是不活躍的。

6.5.8 PRIV：專用擴展SDS項

該項用於定義實驗或應用特定的**SDS**擴展，它包括由長字串對組成的首碼，後跟填充該項其他部分和攜帶所需資訊的字串值。首碼長度段為8位元。首碼字串是定義**PRIV**項人員選擇的名稱，唯一對應應用接收到的其他**PRIV**項。應用實現者可選擇使用應用名稱，如有必要，外加附加子類型標識。另外，推薦其他人根據其代表的實體選擇名稱，然後，在實體內部協調名稱的使用。

注意，首碼消耗了總長為255個八進制項的一些空間，因此，首碼應盡可能的短。這個設備和受到約束的RTCP帶寬不應超載，其目的不在於滿足所有應用的全部控制通訊要求。**SDS PRIV**首碼沒在IANA處註冊。如證實某些形式的**PRIV**項具有通用性，IANA應給它分配一個正式的**SDS**項類型，這樣就不再需要首碼。這簡化了應用，並提高了傳輸的效率。

6. 6 BYE：斷開RTCP包

如混合器接收到一個**BYE**包，混合器轉發**BYE**包，而不改變**SSRC/CSRC**標識。如果混合器關閉，它也應該發出一個**BYE**包，列出它所處理的所有源，而不只是自己的**SSRC**標識。作為可選項，**BYE**包可包括一個8位八進制計數，後跟很多八進制文本，表示離開原因，如："camera

malfunction"或"RTP loop detected"。字串具有同樣的編碼，如在SDES 中所描述的。如字串填充包至下32位元邊界，字串就不以空結尾；否則，BYE包以空八進制填充。

6.7 APP：定義應用的RTCP包

APP包用於開發新應用和新特徵的實驗，不要求註冊包類型值。帶有不可識別名稱的APP包應被忽略掉。測試後，如確定應用廣泛，推薦重新定義每個APP包，而不用向IANA註冊子類型和名稱段。

6.3 RTSP協議

即時流協定（RTSP）是應用級協定，控制即時資料的發送。RTSP提供了一個可擴展框架，使即時資料，如音頻與視頻，的受控、點播成為可能。資料源包括現場資料與存儲在剪輯中資料。該協定目的在於控制多個資料發送連接，為選擇發送通道，如UDP、組播UDP與TCP，提供途徑，並為選擇基於RTP上發送機制提供方法。

6.3.1 簡介

6.3.1.1 目的

即時流協議（RTSP）建立並控制一個或幾個時間同步的連續流媒體。儘管連續媒體流與控制流交叉是可能的，通常它本身並不發送連續流。換言之，RTSP充當多媒體伺服器的網路遠端控制。RTSP連接沒有綁定到傳輸層連接，如TCP。在RTSP連接期間，RTSP用戶可打開或關閉多個對伺服器的可靠傳輸連接以發出RTSP 請求。此外，可使用無連接傳輸協定，如UDP。RTSP流控制的流可能用到RTP，但RTSP操作並不依賴用於攜帶連續媒體的傳輸機制。即時流協定在語法和操作上與HTTP/1.1類似，因此HTTP的擴展機制大都可加入RTSP。協定支援的操作如下：

從媒體伺服器上檢索媒體：

用戶可通過HTTP或其他方法提交一個演示描述。如演示是組播，演示式就包含用於連續媒體的組播位址和埠。如演示僅通過單播發送給用戶，用戶為了安全應提供目的地址。

媒體伺服器邀請進入會議：

媒體伺服器可被邀請參加正進行的會議，或重播媒體，或記錄其中一部分，或全部。這種模式在分散式教育應用上很有用，會議中幾方可輪流按遠端控制按鈕。

將媒體加到現成講座中：

如伺服器告訴用戶可獲得附加媒體內容，對現場講座顯得尤其有用。如HTTP/1.1中類似，RTSP請求可由代理、通道與緩存處理。

6.3.1.2 協定特點

RTSP 特性如下：

可擴展性：

新方法和參數很容易加入RTSP。

易解析：

RTSP可由標準 HTTP或MIME解吸器解析。

安全：

RTSP使用網頁安全機制。

獨立於傳輸：

RTSP可使用不可靠資料報協定（UDP）、可靠資料報協定（RDP），如要實現應用級可靠，可使用可靠流協定。

多伺服器支援：

每個流可放在不同伺服器上，用戶端自動同不同伺服器建立幾個併發控制連接，媒體同步在傳輸層執行。

記錄設備控制：

協定可控制記錄和重播設備。

流控與會議開始分離：

僅要求會議初始化協議提供，或可用來創建唯一會議標識號。特殊情況下， SIP或H.323

可用來邀請伺服器入會。

適合專業應用：

通過SMPTE 時標，RTSP支持幀級精度，允許遠端數字編輯

演示描述中立：

協定沒強加特殊演示或元檔，可傳送所用格式類型；然而，演示描述至少必須包含一個RTSP URI。

代理與防火牆友好：

協議可由應用和傳輸層防火牆處理。防火牆需要理解SETUP方法，為UDP媒體流打開一個"缺口"。

HTTP友好：

此處，RTSP明智的採用HTTP觀念，使現在結構都可重用。結構包括Internet 內容選擇平臺（PICS）。由於在大多數情況下控制連續媒體需要伺服器狀態，RTSP不僅僅向HTTP 添加方法。

適當的伺服器控制：

如用戶啟動一個流，他必須也可以停止一個流。

傳輸協調：

實際處理連續媒體流前，用戶 可協調傳輸方法。

性能協調：

如基本特徵無效，必須有一些清理機制讓用戶決定那種方法沒生效。這允許用戶提出適合的用戶介面。

6.3.1.3擴展RTSP

由於不是所有媒體伺服器有著相同的功能，媒體伺服器有必要支援不同請求集。RTSP 可以如下三種方式擴展，這裏以改變大小排序：

以新參數擴展。如用戶需要拒絕通知，而方法擴展不支援，相應標記就加入要求的段中。

加入新方法。如資訊接收者不理解請求，返回501錯誤代碼（還未實現），發送者不應再次嘗試這種方法。用戶可使用OPTIONS方法查詢伺服器支援的方法。伺服器使用公共回應頭列出支援的方法。

定義新版本協定，允許改變所有部分。（除了協議版本號位置）

6.3.1.4操作模式

每個演示和媒體流可用RTSP URL識別。演示組成的整個演示與媒體屬性由演示描述檔定義。使用HTTP或其他途徑用戶可獲得這個檔，它沒有必要保存在媒體伺服器上。

為了說明，假設演示描述描述了多個演示，其中每個演示維持了一個公共時間軸。為簡化說明，且不失一般性，假定演示描述的確包含這樣一個演示。演示可包含多個媒體流。除媒體參數外，網路目標位址和埠也需要決定。下面區分幾種操作模式：

單播：

以用戶選擇的埠號將媒體發送到RTSP請求源。

組播，伺服器選擇位址：

媒體伺服器選擇組播位址和埠，這是現場直播或准點播常用的方式。

組播，用戶選擇位址：

如伺服器加入正在進行的組播會議，組播位址、埠和密匙由會議描述給出。

6.3.1.5 RTSP狀態

RTSP控制通過單獨協議發送的流，與控制通道無關。例如，RTSP控制可通過TCP連接，而資料流通過UDP。因此，即使媒體伺服器沒有收到請求，資料也會繼續發送。在連接生命期，單個媒體流可通過不同TCP連接順序發出請求來控制。所以，伺服器需要維持能聯繫流與RTSP請求的連接狀態。RTSP中很多方法與狀態無關，但下列方法在定義伺服器流資源的分配與應用上起著重要的作用：

SETUP：

讓伺服器給流分配資源，啟動RTSP連接。

PLAY與RECORD：

啟動**SETUP** 分配流的資料傳輸。

PAUSE :

臨時停止流，而不釋放伺服器資源。

TEARDOWN :

釋放流的資源，**RTSP**連接停止。

標識狀態的**RTSP**方法使用連接頭段識別**RTSP**連接，為回應**SETUP**請求，伺服器連接產生連接標識。

6.3.1.6 與其他協議關係

RTSP在功能上與**HTTP**有重疊，與**HTTP**相互作用體現在與流內容的初始接觸是通過網頁的。目前的協定規範目的在於允許在網頁伺服器與實現**RTSP**媒體伺服器之間存在不同傳遞點。例如，演示描述可通過**HTTP**和**RTSP**檢索，這降低了瀏覽器的往返傳遞，也允許獨立**RTSP** 伺服器與用戶不全依靠**HTTP**。

但是，**RTSP**與**HTTP** 的本質差別在於資料發送以不同協定進行。**HTTP**是不對稱協議，用戶發出請求，伺服器作出回應。**RTSP**中，媒體用戶和伺服器都可發出請求，且其請求都是無狀態的；在請求確認後很長時間內，仍可設置參數，控制媒體流。重用**HTTP**功能至少在兩個方面有好處，即安全和代理。要求非常接近，在緩存、代理和授權上採用**HTTP**功能是有價值的。

當大多數即時媒體使用**RTP**作為傳輸協議時，**RTSP**沒有綁定到**RTP**。**RTSP**假設存在演示描述格式可表示包含幾個媒體流的演示的靜態與臨時屬性。

6.3.2 協議參數

6.3.3 RTSP 信息

RTSP是基於文本的協定，採用**ISO 10646** 字元集，使用**UTF-8**編碼方案。行以**CRLF**中斷，但接收者本身可將**CR**和**LF**解釋成行終止符。基於文本的協定使以自描述方式增加可選參數更容易。由於參數的數量和命令的頻率出現較低，處理效率沒引起注意。如仔細研究，文本協定很容易以腳本語言（如：**Tcl**、**Visual Basic**與**Perl**）實現研究原型。

10646字元集避免敏感字元集切換，但對應用來說不可見。**RTCP**也採用這種編碼方案。帶有重要意義位元的**ISO 8859-1**字元表示如**100001x10xxxxxx**。**RTSP**資訊可通過任何低層傳輸協議攜帶。

請求包括方法、方法作用於其上的物件和進一步描述方法的參數。方法也可設計為在伺服器端只需要少量或不需要狀態維護。當資訊體包含在資訊中，資訊體長度有如下因素決定：

不管實體頭段是否出現在資訊中，不包括資訊體的回應資訊總以頭段後第一和空行結束。

如出現內容長度頭段，其值以位元組計，表示資訊體長度。如未出現頭段，其值為零。

伺服器關閉連接。

注意：**RTSP**目前並不支援**HTTP/1.1**"塊"傳輸編碼，需要有內容長度頭。假如返回適度演示描述長度，即使動態產生，使塊傳輸編碼沒有必要，伺服器也應該能決定其長度。如有實體，即使必須有內容長度，且長度沒顯式給出，規則可確保行為合理。

從用戶到伺服器端的請求資訊在第一行內包括源採用的方法、源標識和所用協定版本。**RTSP**定義了附加狀態代碼，而沒有定義任何**HTTP**代碼。

6.3.4 實體

如不受請求方法或回應狀態編碼限制，請求和回應資訊可傳輸實體，實體由實體頭檔和試題體組成，有些回應僅包括實體頭。在此，根據誰發

送實體、誰接收實體，發送者和接收者可分別指用戶和伺服器。

實體頭定義實體體可選元資訊，如沒有實體體，指請求標識的資源。擴展頭機制允許定義附加實體頭段，而不用改變協議，但這些段不能假定接收者能識別。不可識別頭段應被接收者忽略，而讓代理轉發。

6.3.5 連接

RTSP請求可以幾種不同方式傳送：

- 1、持久傳輸連接，用於多個請求/回應傳輸。
- 2、每個請求/回應傳輸一個連接。
- 3、無連接模式。

傳輸連接類型由RTSP URI來定義。對 "rtsp" 方案，需要持續連接；而 "rtspu" 方案，調用RTSP 請求發送，而不用建立連接。

不象HTTP，RTSP允許媒體伺服器給媒體用戶發送請求。然而，這僅在持久連接時才支持，否則媒體伺服器沒有可靠途徑到達用戶，這也是請求通過防火牆從媒體伺服器傳到用戶的唯一途徑。

6.3.6 方法定義

方法記號表示資源上執行的方法，它區分大小寫。新方法可在將來定義，但不能以\$開頭。

某些防火牆設計與其他環境可能要求伺服器插入RTSP方法和流資料。由於插入將使用戶端和伺服器操作複雜，並強加附加開銷，除非有必要，應避免這樣做。插入二進位資料僅在RTSP通過TCP傳輸時才可使用。流資料（如RTP包）用一個ASCII美圓符號封裝，後跟一個一位元組通道標識，其後是封裝二進位資料的長度，兩位元組整數。流資料緊跟其後，沒有CRLF，但包括高層協議頭。每個\$塊包含一個高層協定資料單元。

當傳輸選擇為RTP，RTCP資訊也被伺服器通過TCP連接插入。缺省情況下，RTCP包在比RTP通道高的第一個可用通道上發送。用戶端可能在另一通道顯式請求RTCP包，這可通過指定傳輸頭插入參數中的兩個通道來做到。當兩個或更多流交叉時，為取得同步，需要RTCP。而且，這為當網路設置需要通過TCP控制連接透過RTP/RTCP提供了一條方便的途徑，可能時，在UDP上進行傳輸。

6.3.7 流水線操作

支持持久連接或無連接的用戶端可能給其請求排隊。伺服器必須以收到請求的同樣順序發出響應。如果請求不是發送給組播組，接收者就確認請求，如沒有確認資訊，發送者可在超過一個來回時間（RTT）後重發同一資訊。

RTT在TCP中估計，初始值為500 ms。應用緩存最後所測量的RTT，作為將來連接的初始值。如使用一個可靠傳輸協議傳輸RTSP，請求不允許重發，RTSP應用反過來依賴低層傳輸提供可靠性。如兩個低層可靠傳輸（如TCP和RTSP）應用重發請求，有可能每個包損失導致兩次重傳。由於傳輸棧在第一次嘗試到達接收者前不會發送應用層重傳，接收者也不能充分利用應用層重傳。如包損失由阻塞引起，不同層的重發將使阻塞進一步惡化。時標頭用來避免重發模糊性問題，避免對圓錐演算法的依賴。每個請求在CSeq頭中攜帶一個系列號，每發送一個不同請求，它就加一。如由於沒有確認而重發請求，請求必須攜帶初始系列號。

實現RTSP的系統必須支援通過TCP傳輸RTSP，並支援UDP。對UDP和TCP，RTSP伺服器的缺省埠都是554。許多目的一致的RTSP包被打包成單個低層PDU或TCP流。RTSP資料可與RTP和RTCP包交叉。不象HTTP，RTSP資訊必須包含一個內容長度頭，無論資訊何時包含負載。否則，RTSP包以空行結束，後跟最後一個資訊頭。

RTP/RTCP流媒體伺服器技術

1 引言

隨著互聯網的飛速發展，流媒體技術的應用越來越廣泛，從網上廣播、電影播放到遠端教學以及線上的新聞網站等都用到流媒體技術。但現有公開

文獻所報導的大多是利用現有的流媒體伺服器來搭建一個流媒體服務系統，或者是針對流媒體資料的編碼方式所進行的研究。本文對流媒體伺服器技術的研究重點在於如何建立一個伺服器，並且在實現流媒體傳輸的兩個基本協議RTP/RTCP的基礎上構建一個基本的流媒體伺服器。

2 流媒體技術簡介

2.1 “流”的定義

現在網上傳輸視頻、音頻主要有下載（Download）和流式傳輸（Streaming）兩種方式。流式傳輸是連續傳送視/音頻信號，當流媒體在客戶機播放時其餘部分在後臺繼續下載。流式傳輸有順序流式傳輸（Progressive Streaming）和即時流式傳輸（Realtime Streaming）兩種方式。即時流式傳輸是即時傳送，特別適合現場事件，即時流式傳輸必須匹配連接帶寬，這意味著圖像品質會因網路速度降低而變差，以減少對傳輸帶寬的需求。“即時”的概念是指在一個應用中資料的交付必須與資料的產生保持精確的時間關係。

在Internet中使用流式傳輸技術的連續時基媒體就稱為流媒體，通常也將其視頻與音頻稱為視頻流和音頻流。實現流式傳輸一般都需要專用伺服器和播放器。

2.2 流媒體系統元件

流媒體是由各種不同軟體構成的，這些軟體在各個不同層面上互相通信，基本的流媒體系統包含以下3個元件：

播放器（Player），用來播放流媒體的軟體。

伺服器（Server），用來向用戶發送流媒體的軟體。

編碼器（Encode），用來將原始的音頻視頻轉化為流媒體格式的軟體。

這些元件之間通過特定的協定互相通信，按照特定的格式互相交換檔資料。有些檔中包含了由特定編解碼器解碼的資料，這種編解碼器通過特定演算法壓縮檔的資料量。

3 流媒體伺服器的基本功能和服務方式

3.1 流媒體伺服器的主要功能

（1）回應客戶的請求，把媒體資料傳送給客戶。流媒體伺服器在流媒體傳送期間必須與客戶的播放器保持雙向通信（這種通信是必需的，因為客戶可能隨時暫停或快放一個檔）。

（2）回應廣播的同時能夠及時處理新接收的即時廣播資料，並將其編碼。

（3）可提供其他額外功能，如：數字許可權管理（DRM），插播廣告，分割或鏡像其他伺服器的流，還有組播。

3.2 流媒體伺服器的服務方式

(1) 單播。在用戶端與媒體伺服器之間建立一個單獨的資料通道，從1台伺服器送出的每個資料包只能傳送給1個客戶機。

(2) 組播。在以組播技術構建的網路上，允許路由器一次將資料包複製到多個通道上。

(3) 點播與廣播。點播連接是用戶端與伺服器之間的主動的連接，在點播連接中，用戶通過選擇內容專案來初始化用戶端連接，用戶可以開始、停止、後退、快進或暫停流。廣播指的是用戶被動地接收流，在廣播過程中，資料包的單獨一個拷貝將發送給網路上的所有用戶，用戶端接收流，但不能控制流。

4 構建流媒體伺服器

4.1 RTP/RTCP協議簡介

即時傳輸協議RTP (Realtime Transport Protocol)：是針對Internet上多媒體資料流程的一個傳輸協定，由IETF(Internet工程任務組)作為RFC1889發佈。RTP被定義為在一對一或一對多的傳輸情況下工作，其目的是提供時間資訊和實現流同步。RTP的典型應用建立在UDP上，但也可以在TCP或ATM等其他協議之上工作。RTP本身只保證即時資料的傳輸，並不能為按順序傳送資料包提供可靠的傳送機制，也不提供流量控制或擁塞控制，它依靠RTCP提供這些服務。

即時傳輸控制協議RTCP (Realtime Transport Control Protocol)：負責管理傳輸品質在當前應用進程之間交換控制資訊。在RTP會話期間，各參與者週期性地傳送RTCP包，包中含有已發送的資料包的數量、丟失的資料包的數量等統計資料，因此，伺服器可以利用這些資訊動態地改變傳輸速率，甚至改變有效載荷類型。RTP和RTCP配合使用，能以有效的回饋和最小的開銷使傳輸效率最佳化，故特別適合傳送網上的即時資料。

RTCP主要有4個功能：

(1) 用回饋資訊的方法來提供分配資料的傳送品質，這種回饋可以用來進行流量的擁塞控制，也可以用來監視網路和用來診斷網路中的問題；

(2) 為RTP源提供一個永久性的CNAME (規範性名字) 的傳送層標誌，因為在發現衝突或者程式更新重啟時SSRC(同步源標識)會變，需要一個運作痕跡，在一組相關的會話中接收方也要用CNAME來從一個指定的與會者得到相聯繫的資料流程 (如音頻和視頻)；

(3) 根據與會者的數量來調整RTCP包的發送率；

(4) 傳送會話控制資訊，如可在用戶介面顯示與會者的標識，這是可選功能。

4.2 RTP/RTCP工作過程

工作時，RTP協議從上層接收流媒體資訊碼流 (如H.263)，裝配成RTP

資料包發送給下層，下層協定提供RTP和RTCP的分流。如在UDP中，RTP使用一個偶數號埠，則相應的RTCP使用其後的奇數號埠。RTP資料包沒有長度限制，它的最大包長只受下層協議的限制。

4.3 伺服器的演算法

伺服器軟體模型主要有兩種，即併發伺服器和迴圈伺服器。迴圈伺服器 (Iterative Server) 是指在一個時刻只處理一個請求的伺服器。併發伺服器 (Concurrent Server) 是指在一個時刻可以處理多個請求的伺服器。事實上，多數伺服器沒有用於同時處理多個請求的冗餘設備，而是提供一種表面上的併發性，方法是依靠執行多個線程，每個線程處理一個請求，從客戶的角度看，伺服器就像在併發地與多個客戶通信。

由於流媒體服務時間的不定性和資料交互即時性的請求，流媒體伺服器一般採用併發伺服器演算法。本文構建了一個基本的流媒體伺服器，能夠同時響應多個用戶的請求，把本地硬碟流媒體檔或即時資料流（H.263格式）發送給用戶。在應用中，把客戶分為請求即時資料的即時客戶和請求檔資料的檔客戶兩類。主要演算法為：

（1）打開設備，分配資源。當設備準備好時，創建一個RTP即時服務線程和一個RTCP即時服務線程。

（2）創建一個UDP套接字並將其綁定到所提供服務的位址之上。

（3）反復調用接收模組，接收來自客戶的RTCP報告，根據其類型做出回應。對新即時客戶的請求，把客戶位址添加到即時服務的客戶列表中，對新檔客戶的請求，則創建一個新RTP檔服務線程和一個新RTCP檔服務線程；對已經在服務中的客戶則根據RTCP報告的內容調整服務。

RTP即時服務線程1：初始化客戶列表和RTP首部。

RTP即時服務線程2：從設備讀取媒體資料，把資料發送給即時服務列表中的客戶。

RTP即時服務線程3：更新RTP首部和統計資料。

RTP即時服務線程4：計算延時，重複第二步。

RTCP即時服務線程1：初始化RTCP首部。

RTCP即時服務線程2：發送發送方報告給即時服務列表中的客戶。

RTCP即時服務線程3：計算延時，重複第二步。

RTP檔服務線程1：初始化RTP首部。

RTP檔服務線程2：從檔讀取媒體資料，把資料發送給客戶。

RTP檔服務線程3：更新已發送資料的統計資訊，為生成發送方報告做準備。

RTP檔服務線程4：計算延時，調整發送速度，正常情況下開始重複第二步。

RTCP檔服務線程1：初始化RTCP首部，發送一個源描述(SDES)報文給客戶。

RTCP檔服務線程2：根據已發送資料的統計資訊生成發送方報告，發送給客戶。

RTCP檔服務線程3：計算延時，正常情況下開始重複第一步。

5 流媒體伺服器實現中應注意的問題

5.1 會話和流的兩級分用

一個RTP會話(Session)包括傳給某個指定目的地對(Destination Pair)的所有通信量，發送方可能包括多個。而從同一個同步源發出的RTP分組序列稱為流(Stream)，一個RTP會話可能包含多個RTP流。一個RTP分組在伺服器端發送出去的時候總是要指定屬於哪個會話和流，在接收時也需要進行兩級分用，即會話分用和流分用。只有當RTP使用同步源標識(SSRC)和分組類型(PTYPE)把同一個流中的分組組合起來，才能夠使用序列號(Sequence Number)和時間戳(Timestamp)對分組進行排序和正確重播。

5.2 多線程的管理

併發伺服器模式要求用多線程來提供服務，所以多線程的管理十分重要。在本文構建的伺服器中，不同客戶的請求和回饋都由伺服器的主線程處理，由於即時資料的獨有性，不同即時客戶可以共用一個RTP即時服務線程和一個RTCP即時服務線程，這樣可以大大減小伺服器的負擔，而每個檔客戶由於請求的檔不同，相應地對速度和開始時間的要求都可能不同，所以需要有自己獨有的RTP檔服務線程和RTCP檔服務線程。

RTP服務線程負責把即時資料流發送給客戶，RTCP服務線程根據RTP線程的統計資料，產生發送方報告給客戶。RTP線程和RTCP線程之間通過一段共用記憶體交互統計資料，對共用記憶體必須設置互斥體進行保護，防止出現錯誤讀寫。在這種方式下，伺服器可以根據每個用戶的不同請求和具體情況方便地提供不同的服務。

5.3 時間戳的處理

時間戳欄位是RTP首部中說明資料包時間的同步資訊，是資料能以正確的時間順序恢復的關鍵。時間戳的值給出了分組中資料的第一個位元組的採樣時間(Sampling Instant)，要求發送方時間戳的時鐘是連續、單調增長的，即使在沒有資料登錄或發送資料時也是如此。在靜默時，發送方不必發送資料，保持時間戳的增長，在接收端，由於接收到的資料分組的序號沒有丟失，就知道沒有發生資料丟失，而且只要比較前後分組的時間戳的差異，就可以確定輸出的時間間隔。

RTP規定一次會話的初始時間戳必須隨機選擇，但協議沒有規定時間戳的單位，也沒有規定該值的精確解釋，而是由負載類型來確定時鐘的顆粒，這樣各種應用類型可以根據需要選擇合適的輸出計時精度。

在RTP傳輸音頻資料時，一般選定邏輯時間戳速率與採樣速率相同，但是在傳輸視頻資料時，必須使時間戳速率大於每幀的一個滴答。如果資料是在同一時刻採樣的，協議標準還允許多個分組具有相同的時間戳值。

5.4 媒體資料發送速度的控制

由於RTP協議沒有規定RTP分組的長度和發送資料的速度，因而需要根據具體情況調整伺服器端發送媒體資料的速度。對來自設備的即時資料可以採取等時間間隔訪問設備緩衝區，在有新資料登錄時發送資料的方式，時間戳的設置相對容易。對已經錄製好的本地硬碟上的媒體檔，以H.263格式的檔為例，由於檔本身不包含幀率資訊，所以需要知道錄製時的幀率或者設置一個初始值，在發送資料的時候找出發送資料中的幀數目，根據幀率和預置值來計算時延，以適當的速度發送資料並設置時間戳資訊。

5.5 多種流同步

RTCP的一個關鍵作用就是能讓接收方同步多個RTP流，例如：當音頻與視頻一起傳輸的時候，由於編碼的不同，RTP使用兩個流分別進行傳輸，這樣兩個流的時間戳以不同的速率運行，接收方必須同步兩個流，以保證聲音與影像的一致。為能進行流同步，RTCP要求發送方給每個傳送一個唯一的標識資料源的規範名(Canonical Name)，儘管由一個資料源發出的不同的流具有不同的同步源標識(SSRC)，但具有相同的規範名，這樣接收方就知道哪些流是有關聯的。而發送方報告報文所包含的資訊可被接收方用於協調兩個流中的時間戳值。發送方報告中含有一個以網路時間協定NTP(Network Time Protocol)格式表示的絕對時間值，接著RTCP報告中給出一個RTP時間戳值，產生該值的時鐘就是產生RTP分組中的TimeStamp欄位的那個時鐘。由於發送方發出的所有流和發送方報告都使用同一個絕對時鐘，接收方就可以比較來自同一資料源的兩個流的絕對時間，從而確定如何將一個流中的時間戳值映射為另一個流中的時間戳值。

6 結論

流媒體技術的應用日益廣泛，對流媒體技術的研究具有很大的實際意義，本文通過對RTP/RTCP協議的研究，分析流媒體伺服器的一般功能和結構，給出構建一個基本的流媒體伺服器的實現方案，實驗證明可以同時滿足多個即時和檔客戶的要求，並已經應用於一個遠端監控系統中。

Application

(Layer 7) This layer supports application and end-user processes.

Communication partners are identified, quality of service is identified, user authentication and privacy are considered, and any constraints on data syntax are identified. Everything at this layer is application-specific. This layer provides application services for file transfers, e-mail, and other network software services. Telnet and FTP are applications that exist entirely in the application level. Tiered application architectures are part of this layer.

Presentation

(Layer 6) This layer provides independence from differences in data representation (e.g., encryption) by translating from application to network format, and vice versa. The presentation layer works to transform data into the form that the application layer can accept.

This layer formats and encrypts data to be sent across a network, providing freedom from compatibility problems. It is sometimes called the syntax layer.

Session

(Layer 5) This layer establishes, manages and terminates connections between applications. The session layer sets up, coordinates, and

terminates conversations, exchanges, and dialogues between the applications at each end. It deals with session and connection coordination.

Transport

(Layer 4) This layer provides transparent transfer of data between end systems, or hosts, and is responsible for end-to-end error recovery and flow control. It ensures complete data transfer.

Network

(Layer 3) This layer provides switching and routing technologies, creating logical paths, known as virtual circuits, for transmitting data from node to node. Routing and forwarding are functions of this layer, as well as addressing, internetworking, error handling, congestion control and packet sequencing.

Data Link

(Layer 2) At this layer, data packets are encoded and decoded into bits. It furnishes transmission protocol knowledge and management and handles errors in the physical layer, flow control and frame synchronization. The data link layer is divided into two sublayers: The Media Access Control (MAC) layer and the Logical Link Control (LLC) layer. The MAC sublayer controls how a computer on the network gains access to the data and permission to transmit it. The LLC layer controls frame synchronization, flow control and error checking.

Physical

(Layer 1) This layer conveys the bit stream - electrical impulse, light or radio signal -- through the network at the electrical and mechanical level. It provides the hardware means of sending and receiving data on a carrier, including defining cables, cards and physical aspects. Fast Ethernet, RS232, and ATM are protocols with physical layer components.

張貼者： 史丹利 於 下午1:17

標籤： **PROTOCOL**

4 則留言：



Kenneth_YITA 提到...

Hello, 版主，

請問RTSP的SETUP可用destination指定Streaming server回送的UDP到哪一個IP address嗎？

THANKS.

2008年6月25日上午10:34



Jerome 提到...

請教版主，支援RTSP的DEVICE是不是一定要搭配RTP & RTCP來使用呢？謝謝。

2008年9月9日下午6:45



史丹利 提到...

是呀,RTSP是溝通協定呀,支援RTSP的Device當然要用RTSP來溝通

呀.

一個只會講客家話,一個只會講台語,應該是無法溝通的吧,如果兩個人都會講國語,才會通呀.

2008年9月10日上午2:24

 **EddieLio168** 提到...

版主你好,在看過文章後,對RTSP有些初步的了解,不知道你是否能給我一些在學習RTSP上的意見,因為一些中文的介紹及實作上似乎有點少,英文書也不知道該買哪一本來研究,我主要想做接收RTSP傳來的影像做儲存及播放的程式,麻煩你了。

2011年10月4日下午6:27

張貼意見

[較新的文章](#)

[首頁](#)

[較舊的文章](#)

訂閱：[張貼意見 \(Atom\)](#)

