

RTP-AA Functional Specification

Dmitri Kondratiev

dkondr@bigfoot.com

Table of Contents

Abstract	3
Scope and Inter-dependencies	3
Components and Interactions	3
Requirements for External Components.....	4

Abstract

This document describes the functional requirements for RTP-AA subsystem, a set of active applications multi-casting RTP packets in Cambira Video/Audio Streaming software.

Scope and Inter-dependencies

Scope

This document describes only the functions of RTP-AA subsystem, namely RTPSender and RTPReceiver. Also requirements for external components are presented.

Inter-dependencies

RTP-AA subsystem depends on the following components:

- Video Proxy Manager
- SAM - Session Announcement Manager
- SM - Smart Messages, implementation for independent QNX processes.
- RTP Session Ring Buffer - component that stores packets of RTP session in a ring buffer, separate ring buffer for every RTP session.

Components and Interactions

RTP-AA subsystem includes :

- RTPReceiver - runs on ingress GSN in the proximity of origin Media Server.
- RTPSender - runs egress GSN in the proximity of Media Clients.

RTPReceiver functions

Main function of RTPReceiver AA is to multi-cast RTP sessions to egress GSNs. For every video and audio RTP session a separate instance of RTPReceiver AA is started by Video Proxy Manager. Every RTP session (video or audio) is multi-casted in a separate ARMTTP session. To achieve its main goal RTPReceiver AA performs these functions in the following order (order is important!) :

1. Initialise itself with parameters received from Video Proxy Manager. These parameters include:
 - Ingress port of RTP session.
 - RTSP URL defining RTP session.
 - RTP Session (track) ID.

2. Request SAM to start announcing new RTP/ARMTP session. Session Announcement (SA) request should have the following parameters set in Application Info field:
 - RTSP URL defining RTP session.
 - RTP Session (track) ID.
3. Wait for at least one client to join RTP/ARMTP session.
4. When at least one client has joined RTP/ARMTP session build ARMTP sender channel. Thus a separate ARMTP sender channel is built for every RTP session.
5. Register ingress filter for RTP session.
6. Start receiving RTP session.
7. Encapsulate RTP packets into ARMTP packets and send them to egress GSN using ARMTP sender channel.
8. Process 'shutdown RTP session' request from Video Proxy Manager. Destroy ARMTP sender channel corresponding to the given RTP session.

RTPSender functions

RTPSender runs on egress GSN. Main role of RTPSender AA is to receive packets from RTP/ARMTP session, extract RTP packets from ARMTP packets and put them in RTP Session Ring Buffer. RTP Session Ring Buffer is a component external to this specification. Every RTP session is multi-casted in a separate ARMTP session. For every RTP session a separate instance of RTPSender AA is started by Video Proxy Manager. Single instance of RTPSender serves *all* video receivers interested in the particular RTP session from the given egress GSN. That means that Video Proxy Manager must start only one RTPSender instance for the given RTP session. Video receivers (clients) joining later this RTP session will get data from RTP Session Ring Buffer. To achieve its main goal RTPSender AA performs these functions in the following order (order is important!) :

1. Initialise itself with parameters received from Video Proxy Manager. These parameters include:
 - RTSP URL defining RTP session.
 - RTP Session (track) ID.
2. Register with SAM to wait for RTP/ARMTP session announcements.
3. Process RTP/ARMTP session announcements comparing RTSP URL and RTP Session (track) ID with corresponding parameters sent in Session Announcement (SA) packet. In case parameters match request SAM to join this RTP/ARMTP session.
4. Wait when joining RTP/ARMTP session completes.
5. Build ARMTP receiver channel and start it.
6. Receive ARMTP session, extract RTP packets from ARMTP packets and put them in corresponding RTP Session Buffer.
7. Process 'shutdown RTP session' request from Video Proxy Manager. Destroy ARMTP receiver channel corresponding to the given RTP session.

Requirements for External Components

Video Proxy Manager

This component must provide:

1. RTSender and RTPReceiver initialisation parameters passed as s-messages (smart messages). See RTSender and RTPReceiver functions for parameter description.
2. Must terminate RTPSender and RTPReceiver sending 'shutdown' s-message defined by these AA. See RTSender and RTPReceiver functions for description of 'shutdown' feature.

SAM - Session Announcement Manager

This component must provide:

1. S-message API to register for session announcements events.
2. S-message API to start session announcement.
3. S-message API to request joining session.

SM - Smart Messages, implementation for independent QNX processes.

This component must provide API to send / receive messages from external QNX processes.

RTP Session Ring Buffer - component that stores packets of RTP session in a ring buffer, separate ring buffer for every RTP session.

This component must provide:

1. S-message API to get unique (for the given node) Ring Buffer ID or RBID in return for a tuple: (RTSP URL, RTP Session ID).

Signature: **RBID getRBID (RTSP_URL, RTP_SESSION_ID)**

2. S-message API to put RTP packet in ring buffer:

Signature: **Bool putRTP_Pkt(RBID, RTP_PKT)**

This function puts RTP packet into ring buffer slot pointed by ring buffer 'put' pointer.

3. S-message API to get next RTP packet from ring buffer:

Signature: **RTP_PKT getRTP_Pkt(RBID)**

This function gets RTP packet from ring buffer slot pointed by ring buffer 'get' pointer.

Note: This function will not be used by RTP-AA subsystem but provided here for completeness of RTP Session Ring Buffer component functional specification. It is needed by Video system on egress node to get the particular RTP session packets and then send them to all client video receivers, interested in the \corresponding RTP session.

