


|   |  |            |
|---|--|------------|
|  |  | Page1 (76) |
| <i><b>PUCC Streaming Protocol</b></i>   |  |            |

**PUCC Streaming Protocol**  
**(Version 3.0 – March 22, 2012)**

**Peer-to-Peer Universal Computing Consortium (PUCC)**

**Intellectual Property Notice**

©Copyright PUCC 2012. Confidential – Disclosure to PUCC members only. The information contained in this work is confidential and must not be reproduced, disclosed to non-PUCC-members without the prior written permission of PUCC, or used except as expressly authorized in writing by PUCC

**Table of Content**

|   |    |
|---|----|
| 1. Introduction .....   | 3  |
| 2. Terminology .....  | 3  |
| 2.1. Definition .....   | 3  |
| 2.2. Abbreviations .....  | 3  |
| 3. References .....   | 4  |
| 4. Goal and Requirements .....  | 5  |
| 4.1. Goals .....  | 5  |
| 4.2. Requirements .....   | 5  |
| 5. Protocol Overview .....  | 5  |
| 5.1. Protocol Design .....  | 6  |
| 5.2. PUCC Streaming Control Protocol .....  | 6  |
| 6. PUCC Streaming Control Protocol .....  | 7  |
| 6.1. Setup Method .....   | 10 |
| 6.2. Release Method .....   | 24 |
| 6.3. Abort Method .....   | 26 |
| (1) In case of Unicast Streaming .....  | 27 |
| (2) In case of Multicast/Broadcast Streaming .....  | 29 |
| 6.4. Prune Method .....   | 33 |
| 6.5. Deliver Method .....   | 36 |
| 6.6. Keepalive Method .....   | 40 |
| 6.7. FlowInformationAdvertise Method .....  | 46 |
| (1) In case of Receiver initiative streaming .....  | 46 |
| (2) In case of Sender-initiated streaming .....   | 47 |
| 6.8. MulticastSetup Method .....  | 52 |
| 6.9. MulticastKeepalive Method .....  | 63 |
| Appendix A. Version History .....   | 70 |
| Appendix B. Example of Device Discovery and PUCC Streaming Sequence .....                             | 71 |
| Appendix C: Namespace Definitions .....   | 73 |
| Appendix D: How to Set an SDP Session Description in Flow Information .....                           | 74 |
| Appendix E: Transport binding of the content transfer protocol messages over the Bluetooth(SPP) ..... | 75 |

|   |  |            |
|---|--|------------|
|  |  | Page3 (76) |
| <b><i>PUCC Streaming Protocol</i></b>   |  |            |

## 1. Introduction

This document is a template for Technical documents.

## 2. Terminology

### 2.1. Definition

**[Node]** A node is an entity that can communicate with other nodes. “Node” is a synonym for “Peer” in this spec. Each node is uniquely identified by a unique node id.

**[Node ID]** Node ID is a unique identifier for a node that distinguishes it from all other nodes on the network. For example, the Node ID is defined as FQDN (Fully Qualified Domain Name) or UUID (Universal Unique Identifier).

**[Community]** A community is a set of nodes that have a common set of interests or that obey a common set of policies. Each community is uniquely identified by a unique Community ID. Nodes belong to the same community can communicate one another.

**[Community ID]** Community ID is a unique identifier for a community that distinguishes it from all other communities. For example, the community id is defined as URN with specific namespace.

**[Multicast Group]** A multicast group is a collection of nodes under a defined multicast Group.

**[Multicast Group ID]** Multicast group ID is a unique identifier for a multicast group that distinguishes it from all other node groups on the network. For example, the Node Group ID is defined as URN with specific namespace.

**[Message]** The basic unit of PUCC communication; consists of an XML-structured sequence of octets and transmitted via a connection.

**[Message ID]** Message ID is a unique identifier for a message that distinguishes it from all other messages. For example, the message ID can be defined as, node provided sequence number + current time + node ID

**[Application Protocol ID]** Application Protocol ID is a unique identifier for a protocol that distinguishes it from all other application protocols. For example, XML namespace can be used to distinguish the application protocols.

**[Communication Mode]** It indicates the message exchange method. Proactive Mode and Reactive Mode are defined.

**[Communication Type]** It indicates the method of message sending. Unicast, Multicast, and Broadcast are defined.

### 2.2. Abbreviations

**PUCC** Peer-to-Peer Universal Computing Consortium

***PUCC Streaming Protocol***

|             |                                       |
|-------------|---------------------------------------|
| <b>TCP</b>  | Transmission Control Protocol         |
| <b>UDP</b>  | User Datagram Protocol                |
| <b>IP</b>   | Internet Protocol                     |
| <b>XML</b>  | eXtensible Markup Language            |
| <b>P2P</b>  | Peer-to-peer                          |
| <b>MIME</b> | Multipurpose Internet Mail Extensions |
| <b>URI</b>  | Universal Resource Identifier         |
| <b>RTP</b>  | Real-time Transport Protocol          |
| <b>AVP</b>  | Audio/Video Profile                   |
| <b>MMS</b>  | Microsoft Media Server                |
| <b>CIP</b>  | Common Isochronous Packet             |
| <b>SDP</b>  | Session Description Protocol          |

### 3. References

[PUCC] “Peer-to-Peer Universal Computing Consortium”,

URL: <http://www.pucc.jp/>

[XML] “Extensible Markup Language (XML) 1.0 (Second Edition)”, W3C Recommendation 6 October 2000, T. Bray et al. URL: <http://www.w3.org/TR/2000/REC-xml-20001006>

[HTTP] “Hypertext Transfer Protocol -- HTTP/1.1”, RFC2616, R. Fielding et al., June 1999. URL: <http://www.ietf.org/rfc/rfc2616.txt>

[TCP] “Transmission Control Protocol”, RFC793, J. Postel. September 1981. URL: <http://www.ietf.org/rfc/rfc793.txt>

[UDP] “User Datagram Protocol”, RFC768, J. Postel. August 28 1980. URL: <http://www.ietf.org/rfc/rfc768.txt>

[RTP] “A Transport Protocol for Real-Time Applications”, RFC1889, H. Schulzrinne, January 1996. URL: <http://www.ietf.org/rfc/rfc1889.txt>

[IP] “Internet Protocol”, RFC791, J. Postel. September 1981. URL: <http://www.ietf.org/rfc/rfc791.txt>

[URI] “Universal Resource Identifier” URL: <http://www.w3.org/Addressing/>

[Bluetooth] “Bluetooth” URL: <http://www.bluetooth.com/>

[IEEE1394] “IEEE1394” URL: <http://www.1394ta.org/>

[SDP] “Session Description Protocol”

URL: <http://www.ietf.org/rfc/rfc2327.txt>

## 4. Goal and Requirements

### 4.1. Goals

The goal of this document is:

- ◆ To give you guidelines for preparing technical documents.

### 4.2. Requirements

#### 1. Clearness

Technical documents should be easy to understand.

#### 2. Accuracy

Technical specifications should be defined precisely.

#### 3. Language

Technical specifications must be in English

## 5. Protocol Overview

The PUCC streaming is a method to transmit streaming data from a certain node to another node or a multicast group member in real time using the bucket relay model according to the topology of the PUCC network.

The following are assumed as use cases for the PUCC streaming.

Table5-1 PUCC Streaming Use cases

|   | Use Case   | Model                  |
|---|--|------------------------|
| 1 | A certain PUCC node becomes a broadcasting station, and the streaming data is broadcast to two or more viewers' PUCC nodes.  | Broadcast<br>Multicast |
| 2 | The node with the camera and the microphone delivers the streaming data to another node.   | Broadcast<br>Unicast   |
| 3 | A certain PUCC node requests the delivery of data to the PUCC node that holds the streaming data such as videos and the radios, and as a result the streaming data is delivered. | On demand<br>Unicast   |

The PUCC node related to the transmission of certain streaming data plays either of the

following roles.

Table5-2 Role of Pucc nodes

|   | Role     | Description  |
|---|----------|--|
| 1 | Sender   | A Pucc node that installs an application that transmits streaming data, and begins delivery of streaming data by the user's instruction. The Sender transmits the streaming data to the Relay or the Receiver.   |
| 2 | Relay    | A Pucc node that installs an application that transmits streaming data, and doesn't offer user streaming data. The Relay transmits streaming data to another Relay or the Receiver. The Pucc node that installs application that transmits streaming data, and doesn't offer user streaming data. The Relay transmits the streaming data to another Relay or the Receiver.                                       |
| 3 | Receiver | A Pucc node that installs an application that receives and transmits streaming data and offers user streaming data. Like the Relay, the Receiver transmits the streaming data if there is any Relay or Receiver that becomes a destination of the forwarded streaming data. The transmission of the streaming data ends if there is no Relay or Receiver that becomes a destination of forwarded streaming data. |

### 5.1. Protocol Design

In Pucc streaming, an out channel signal system is adopted as a protocol. That is, the transmission way (channel) of the message which controls streaming and the transmission way (the channel) of the streaming data are divided. The Pucc protocol message for controlling a Pucc network has adopted the XML in text format. However, XML is not suitable for transmission of streaming data. Then, the transmission protocol used conventionally is used for transmission of streaming data.

Therefore, the Pucc streaming protocol consists only of the Pucc streaming control protocol which controls streaming. Transmission of streaming data uses RTP, HTTP and IEEE1394 Isochronous transmission protocols, which are used for general purposes.

### 5.2. Pucc Streaming Control Protocol

The Pucc Streaming Control Protocol performs the setup and release of a transmission way for transmitting streaming data. A protocol message is exchanged by the same channel (Pucc streaming control channel) as Pucc's existing protocols.

The protocol structure of PUCC streaming control channel is shown in Fig. 5.2.

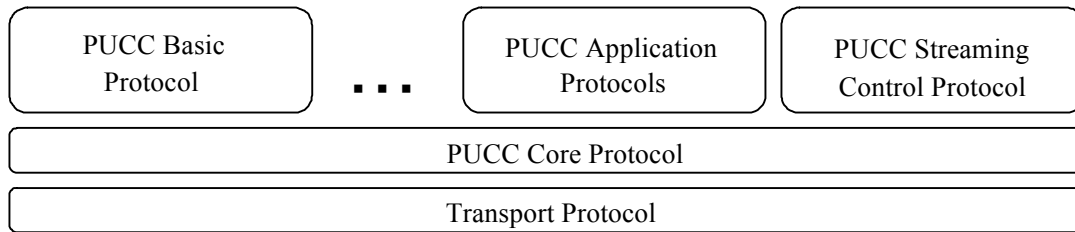


Figure5.2. Protocol Structure of PUCC Streaming Control Channel

## 6. PUCC Streaming Control Protocol

The PUCC streaming control protocol consists of the following nine methods.

Table6. Method list of PUCC streaming control protocol(1/3)

| No. | Method         | Summary   |
|-----|----------------|---|
| 1   | Setup Method   | <p>This method is used in order to set up the streaming data transmission channel from the Sender node to the Receiver node.</p> <p>It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Request Type : Setup Message</li> <li>• Response Type : SetupResponse Message</li> </ul> |
| 2   | Release Method | <p>This method is used in order to release a channel to transmit the streaming data transmission channel from the Sender node to the Receiver node.</p> <p>It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Advertise Type : Release Message</li> </ul>                       |

Table6. Method list of PUCC streaming control protocol(2/3)

| No. | Method                          | Summary  |
|-----|---------------------------------|--|
| 3   | Abort Method                    | <p>This method is used in order to notify the Sender/Relay/Receiver nodes which adjoin in the upstream direction of the stop of the streaming.</p> <p>It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Advertise Type : Abort Message</li> </ul>   |
| 4   | Prune Method                    | <p>This method is used in order to notify the end of a relay. The Relay node located at the end of a streaming data route notifies an upstream node of the secession from the flow.</p> <p>It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Advertise Type : Prune Message</li> </ul>  |
| 5   | Deliver Method                  | <p>This method is used in order to perform the demand of Receiver initiative streaming from the Receiver to the Sender.</p> <p>It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Request Type : DeliverRequest Message</li> <li>• Response Type : DeliverResponse Message</li> </ul>  |
| 6   | Keepalive Method                | <p>This method is used in order to maintain the flow under streaming data transmission in PUCC unicast streaming. Each node on the route of the flow exchanges messages with the adjacent nodes which have transmitted the Setup message.</p> <p>It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Request Type : Keepalive Message</li> <li>• Response Type : KeepaliveResponse Message</li> </ul> |
| 7   | FlowInformationAdvertise Method | <p>This method is used in order to advertise flow information from the Sender to the Receiver (multicast member or community member).</p> <p>It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Advertise Type : FlowInformationAdvertise Message</li> </ul>   |



Table6. Method list of PUCC streaming control protocol(3/3)

| No. | Method                       | Summary  |
|-----|------------------------------|--|
| 8   | MulticastSetup<br>Method     | <p>This method is used in order to set up the streaming data transmission channel from the Receiver node to the Sender node. A route is set up in the section from the Receiver node (multicast member or community member) to the Sender node. It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Request Type : MulticastSetup Message</li> <li>• Response Type : MulticastSetupResponse Message</li> </ul>                          |
| 9   | MulticastKeepalive<br>Method | <p>This method is used in order to maintain the flow under streaming data transmission in PUCC multicast (broadcast) streaming. Each node on the route of the flow exchanges messages with the adjacent nodes which have transmitted the MulticastSetup message. It consists of the following message.</p> <ul style="list-style-type: none"> <li>• Request Type : MulticastKeepalive Message</li> <li>• Response Type : MulticastKeepaliveResponse Message</li> </ul> |

### 6.1. Setup Method

This method is used in order to set up the streaming data transmission channel from the Sender node to the Relay/Receiver node.

This method consists of the following messages.

- ◆ Request Type : Setup Message
- ◆ Response Type : SetupResponse Message

Message exchange is performed by single-hop unicast. This message is transmitted and received among the following nodes.

- Between the Sender and the Relay/Receiver
- Between the Relay and the Relay/Receiver
- Between the Receiver and the Relay/Receiver

The sequence of the Setup method is shown in the illustrated model below as an example. (The arrow means the direction where streaming data flows.)

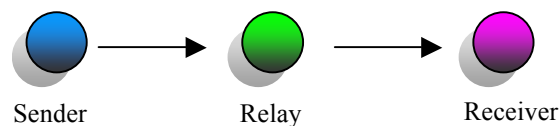


Figure4.1-1 Ex: Network model of Setup method

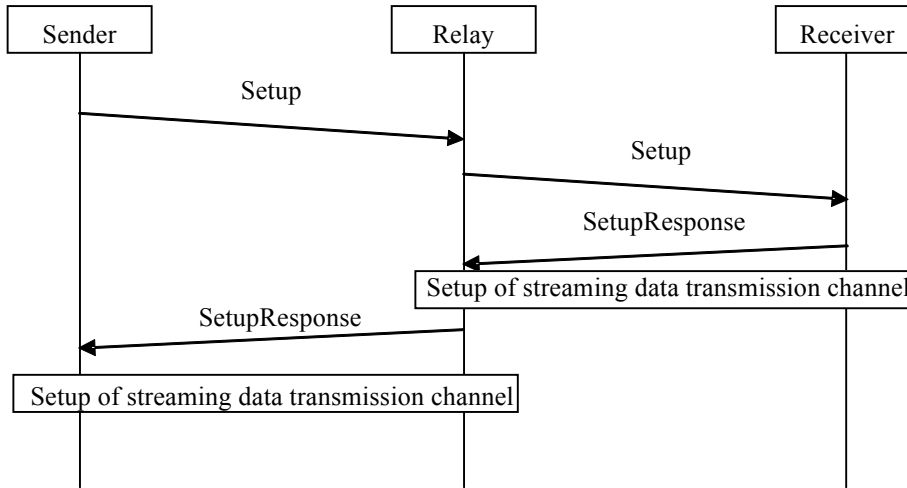


Figure4.1-2 Sequence of Setup method

The Sender sets up a route for transmitting streaming data and the flow information. The Sender transmits a Setup message to the following downstream node on the route. A flow ID which identifies the flow, the name of the flow, the start/end time, a MIME type of media data, optional information of the Sender's application, the flow classification, the directivity of the flow, bandwidth information and so on is included in the flow information. The node which has received the Setup message allocates a streaming data transmission channel according to the contents of flow information and transmits the Setup message to the following downstream node according to the route set as the message. Upon receipt of the Setup message, the Receiver allocates resources for the streaming to be performed with the source node and returns to the source node a SetupResponse message which contains the resource information and the decision for or against the resource allocation. The node which receives the SetupResponse message establishes a transmission channel with the source node, using the reserved streaming resources. At the same time, the node sends a SetupResponse which sets information on the streaming resources to the sender of the Setup.

Thus, the Setup message is transmitted sequentially from the Sender to the Receiver via the nodes on the route. When the Sender receives the Setup Response message forwarded from the Receiver back to the Sender through the transmission channels set up between each pair of the nodes on the route, a streaming data transmission channel is finally established.

Moreover, in unicast streaming, bi-directional streaming data transmissions are supported. When performing route setup to a bi-directional flow, each node sets the address for transmission as a Setup message. All the nodes on the route mutually notify the waiting port for the streaming data between each node. By performing such a Setup, it is possible to establish two transmission channels, from the Sender

***PUCC Streaming Protocol***

to the Receiver and from the Receiver to the Sender, respectively.

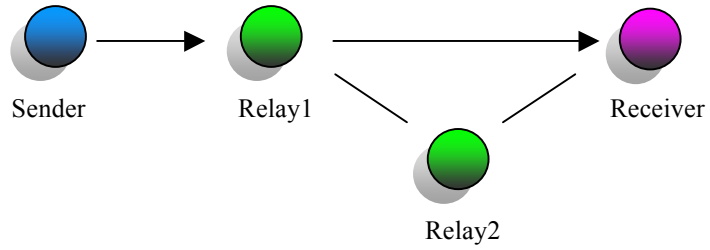


Figure4.1-3 Ex: Network model of Setup method -failure of route setup-

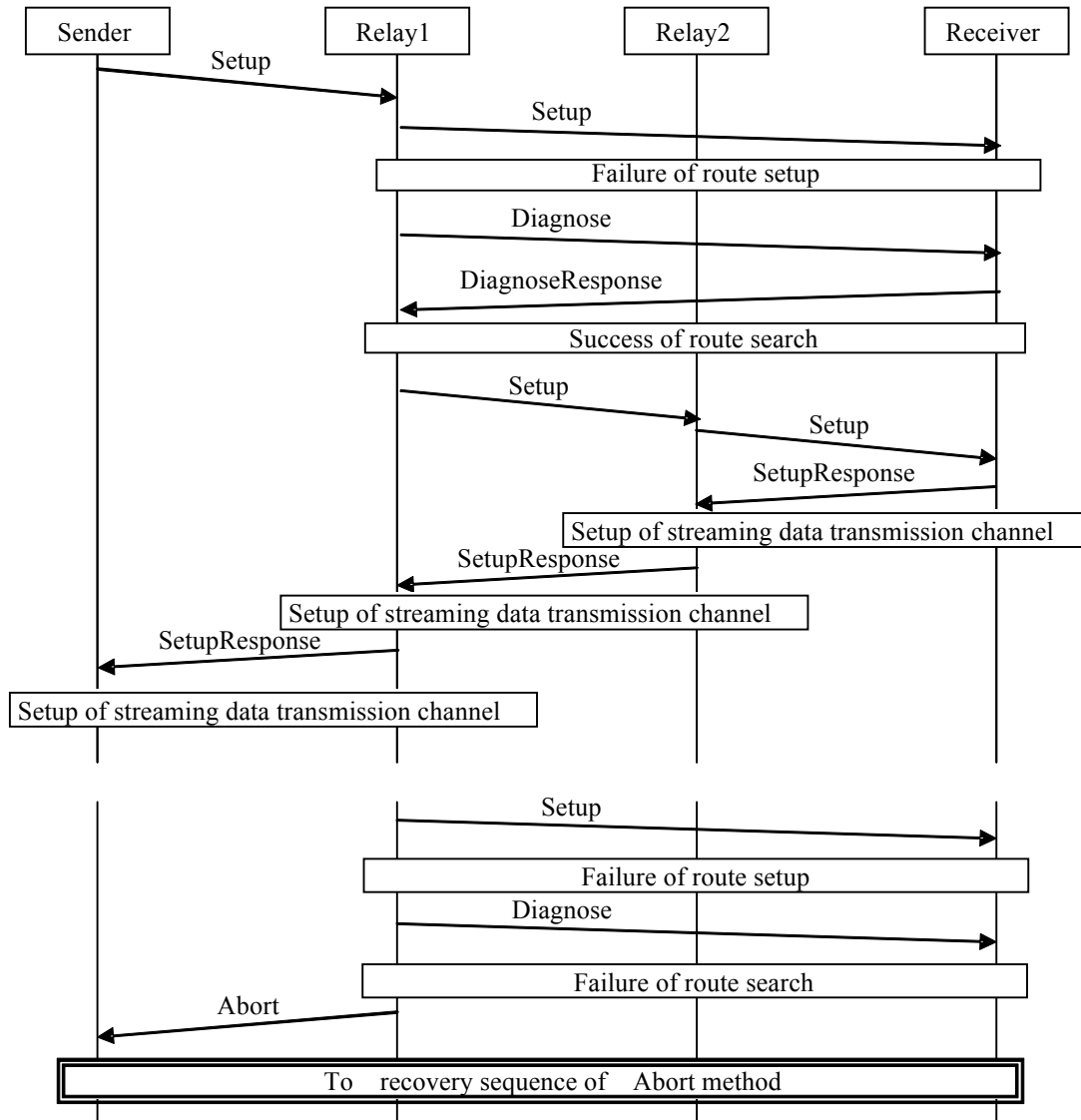



Figure4.1-4 Sequence of Setup method –Failure of route setup

When the Setup goes wrong among arbitrary nodes, the node which has detected the failure tries to set up a flow, using the new route to the Receiver. In this case, the Setup message is set up as follows.

- Flow element : Same flow ID
- Source element : Self-node
- Destination element : New route

The "reset" parameter is added to the Setup message, aiming at a re-setup of the route. When the route

|   |  |              |
|---|--|--------------|
|  |  | Page 14 (76) |
| <b><i>PUCC Streaming Protocol</i></b>   |  |              |

set up by the same flow ID exists, the node returns a SetupResponse.

When the search or setup of a course goes wrong, a detection node performs the following processing.

- A detection node : Relay node

The Relay node transmits an Abort message to the adjacent node in the Sender' direction.

- A detection node : the Receiver (the end of setup)

The Receiver processes nothing.

- A detection node : Sender (the start of setup)

The Sender stops a flow setup and notifies a failure in the flow setup to the user's application.

**PUCC Streaming Protocol**

The parameter of the Setup message is defined as follows.

Table4.1-1 Setup message(1/2)


| Element name | Element value  | Attribute name | Attribute value   | Occurrence | Status   |
|--------------|--|----------------|---|------------|----------|
| Setup        | XML fragment<br>(Source element, Destination element, Flow element, RequestedMedium element, TransmissionMedium element)   | -              | -   | 1          | Required |
|              |  | xmlns          | Name space identifier (URI) of PUCC Streaming Control Protocol                      | 1          | Required |
|              |  | type           | Classification of setup<br>"initial"(initialization),<br>"reset"(resetting)         | 1          | Option   |
| Source       | Node ID of setup start node  | -              | -   | 1          | Required |
| Destination  | XML fragment<br>(Route element, Target element)  | -              | -   | 1          | Required |
| Route        | XML fragment<br>(Route element, Target element)  | -              | -   | 1          | Required |
|              |  | Node           | Node ID of Route node(Relay/Receiver)   | 1          | Required |
| Target       | Receiver node ID   | -              | -   | 1          | Required |
| Flow         | XML fragment<br>(Name element, Description element, Sender element, Bandwidth element, ActiveTime element, Repeat element, MediaData element, Application element, Url element, Sdp element) | -              | -   | 1          | Required |
|              |  | ID             | ID of unique identifier<br>(<sequence number><br><Data><Time>”@”<Sender<br>nodeID>) | 1          | Required |
|              |  | initiative     | Initiative node of setup start<br>("Sender",<br>"Receiver")                         | 1          | Option   |
|              |  | directionality | Direction of flow<br>("UniDirectional",<br>"BiDirectional")                         | 1          | Option   |
| Name         | Name of flow   | -              | -   | 1          | Option   |
| Description  | Description about flow   | -              | -   | 1          | Option   |
| Sender       | Sender node ID   | -              | -   | 1          | Option   |
| Bandwidth    | Band information (numerical value[bps])  | -              | -   | 1          | Option   |
| ActiveTime   | XML fragment<br>(Start element, End element)   | -              | -   | 1          | Option   |
| Start        | Starting time of flow<br>(ISO8601 form)  | -              | -   | 1          | Required |
| End          | Ending time of flow<br>(ISO8601 form)  | -              | -   | 1          | Required |

**PUC Streaming Protocol**

Table 4.1-1 Setup message(2/2)

| Element name       | Element value  | Attribute name | Attribute value   | Occurrence | Status   |
|--------------------|--|----------------|---|------------|----------|
| Repeat             | XML fragment (Interval element, Duration element,)   | -              | -   | 1          | Option   |
| Interval           | Time interval of repetition(Unit[ms])                | -              | -   | 1          | Required |
| Duration           | Active duration(Unit[ms])                            | -              | -   | 1          | Required |
| MediaData          | XML fragment (MimeType element)                      | -              | -   | 1          | Option   |
| MimeType           | MIME type of media data                              | -              | -   | Multiple   | Required |
| Application        | XML fragment (Option element)                        | -              | -   | 1          | Option   |
|                    |  | name           | Name of application which distributes media data                            | 1          | Required |
| Option             | Arbitrary characters                                 | -              | -   | 1          | Option   |
| Url                | Url of streaming contents                            | -              | -   | 1          | Option   |
| Sdp                | SDP session description                              | -              | -   | 1          | Option   |
| RequestedMedium    | XML fragment (Address element, ResourceList element) | -              | -   | 1          | Required |
|                    |  | medium         | Classification of transport medium ("RTP/AVP", "MMS", "CIP", "HTTP")        | 1          | Required |
|                    |  | protocol       | Classification of transport address ("UDP", "IEEE1394", "Bluetooth", "TCP") | 1          | Required |
|                    |  | type           | Classification of address "IPv4", "IPv6", "IEEE1394", "Bluetooth"           | 1          | Required |
| TransmissionMedium | XML fragment (Address element, ResourceList element) | -              | -   | 1          | Required |
|                    |  | medium         | Classification of transport medium ("RTP/AVP", "MMS", "CIP", "HTTP")        | 1          | Required |
|                    |  | protocol       | Classification of transport address ("UDP", "IEEE1394", "Bluetooth", "TCP") | 1          | Required |
|                    |  | type           | Classification of address "IPv4", "IPv6", "IEEE1394", "Bluetooth"           | 1          | Required |
| Address            | Streaming data transmission address                  | -              | -   | 1          | Required |
| ResourceList       | XML fragment (Resource element)                      | -              | -   | 1          | Required |
| Resource           | Resources  | -              | -   | Multiple   | Required |
|                    |  | type           | Type of resource. ("port", "channel")                                       | 1          | Required |
|                    |  | mimetype       | MIME type of media data   | 1          | Required |



|   |  |             |
|---|--|-------------|
|  |  | Page17 (76) |
| <i><b>PUCC Streaming Protocol</b></i>   |  |             |

**PUC Streaming Protocol**

The Setup message is mapped by the PUC core protocol as follows.

Table4.1-2 Mapping of Setup message for PUC Core Protocol

| Element name | Element value                 | Attribute name | Attribute value                                  | Occurrence | Status   |
|--------------|-------------------------------|----------------|--|------------|----------|
| Core         | XML fragment                  | -              | -  | 1          | Required |
|              |                               | xmlns          | Name space identifier (URI) of PUC Core Protocol | 1          | Required |
| ComType      | “Unicast”                     | -              | -  | 1          | Required |
| MsgID        | Message ID                    | -              | -  | 1          | Required |
| MsgType      | “Request”                     | -              | -  | 1          | Required |
| CommunityID  | Community ID                  | -              | -  | 1          | Option   |
| Source       | Source node ID                | -              | -  | 1          | Required |
| Destination  | XML fragment (Target element) | -              | -  | 1          | Required |
| Target       | Destination node ID           | -              | -  | 1          | Required |
| TraceRoute   | XML fragment (Route element)  | -              | -  | 1          | Option   |
| Route        | XML fragment (Route element)  | -              | -  | 1          | Required |
|              |                               | Node           | Relay node ID                                    | 1          | Required |
| SessionID    | Session ID                    | -              | -  | 1          | Option   |
| MsgBody      | XML fragment (Setup element)  | -              | -  | 1          | Required |
|              |                               | protocol       | Protocol ID of PUC Streaming Control Protocol    | 1          | Required |

The following is an example of the Setup message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Request</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029" />
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <Setup xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl" type="initial">
      <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
      <Destination>
        <Route Node="975870cd-f9bb-4305-1234-e65e29f6030">
          <Target>975870cd-f9bb-4305-1234-e65e29f6031</Target>
        </Route>
      </Destination>
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="UniDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
      </Flow ID>
    </Setup>
  </MsgBody>
</Core>

```

**PUCC Streaming Protocol**

```

<Repeat>
  <Interval>01:30.00</Interval>
  <Duration>15:00.00</Duration>
</Repeat>
<MediaData>
  <MimeType>video/x-ms-wmv</MimeType>
</MediaData>
< Application name="WindowsMediaService9">
  <Option xmlns="http://www.microsoft.com/japan/windows/windowsmedia/9series/server.aspx">
    <MulticastAddress>224.0.0.1</MulticastAddress>
    <Port>6600</Port>
  </Option>
</Application>
<RequestedMedium medium="RTP/AVP" protocol="UDP" type="IPv4"/>
</Flow>
</Setup>
</MsgBody>
</Core>

```

Figure4.1-5 Ex1:Mapping of Setup Message

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Request</MsgType>
  <CommunityID>urn:ED:Community:PUCC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029" />
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <Setup xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl" type="reset">
      <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
      <Destination>
        <Route Node="975870cd-f9bb-4305-1234-e65e29f6030">
          <Target>975870cd-f9bb-4305-1234-e65e29f6031</Target>
        </Route>
      </Destination>
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="BiDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
        <Repeat>
          <Interval>90000</Interval>
          <Duration>900000</Duration>
        </Repeat>
        <MediaType>
          <MimeType>video/x-ms-wmv</MimeType>
          <MimeType>audio/x-ms-wma</MimeType>
        </MediaType>
        < Application name="WindowsMediaService9"/>
      </Flow>
      <RequestedMedium medium="RTP/AVP" protocol="UDP" type="IPv4" />
      <TransmissionMedium medium="RTP/AVP" protocol="UDP" type="IPv4">
        <Address>192.168.0.1</Address>
      </TransmissionMedium>
    </Setup>
  </MsgBody>
</Core>

```

**PUCC Streaming Protocol**

```

<ResourceList>
  <Resource type="port" mimetype="video/x-ms/wmv">9900</Resource>
  <Resource type="port" mimetype="audio/x-ms/wma">9902</Resource>
</ResourceList>
</TransmissionMedium>
</Setup>
</MsgBody>
</Core>

```

Figure4.1-6 Ex2:Mapping of Setup Message

The parameter of the SetupResponse message is defined as follows.

Table4.1-3 SetupResponse message (1/2)

| Element name  | Element value   | Attribute name | Attribute value  | Occurrence | Status   |
|---------------|---|----------------|--|------------|----------|
| SetupResponse | XML fragment (Result element, Reason element, Flow element, TransmissionMedium element)   | -              | -  | 1          | 必須       |
|               |   | xmlns          | Name space identifier (URI) of PUCC Streaming Control Protocol                           | 1          | Required |
|               |   | type           | Classification of setup "initial"(initialization), "reset"(resetting)                    | 1          | Option   |
| Result        | "Success", "Failure"  | -              | -  | 1          | Required |
| Reason        | "NotEnoughBandwidth", "UnsupportedTransport", "PortUnavailable", "InternalError"  | -              | -  | 1          | Option   |
| Flow          | XML fragment (Name element, Description element, Sender element, Bandwidth element, ActiveTime element, Repeat element, MediaData element, Application element, Url element, Sdp element) | -              | -  | 1          | Required |
|               |   | ID             | ID for discriminating flow uniquely (<sequence number> <Data><Time>'"@'"<Sender nodeID>) | 1          | Required |
|               |   | initiative     | Initiative node of setup start ("Sender", "Receiver")                                    | 1          | Option   |
|               |   | directionality | Direction of flow "UniDirectional", "BiDirectional"                                      | 1          | Option   |
| Name          | Name of flow  | -              | -  | 1          | Option   |
| Description   | Description of flow   | -              | -  | 1          | Option   |
| Sender        | Sender node ID  | -              | -  | 1          | Option   |
| Bandwidth     | Band information (numerical value[bps])   | -              | -  | 1          | Option   |
| ActiveTime    | XML fragment (Start element, End element)   | -              | -  | 1          | Option   |
| Start         | Starting time of flow (ISO8601 form)  | -              | -  | 1          | Required |
| End           | Ending time of flow (ISO8601 form)  | -              | -  | 1          | Required |
| Repeat        | XML fragment (Interval element, Duration element)   | -              | -  | 1          | Option   |
| Interval      | Time interval of repetition(Unit[ms])   | -              | -  | 1          | Required |
| Duration      | Active duration(Unit[ms])   | -              | -  | 1          | Required |

Table4.1-3 SetupResponse message (2/2)

| Element Name       | Element Name                                     | Attribute Name | Attribute Value   | Occurrence | State    |
|--------------------|--|----------------|---|------------|----------|
| MediaData          | XML Fragment (MimeType element)                  | -              | -   | 1          | Option   |
| MimeType           | MIME Type of Media Data                          | -              | -   | Multiple   | Required |
| Application        | XML Fragment (Option Element)                    | -              | -   | 1          | Option   |
|                    |  | name           | Name of application which distributes media data                            | 1          | Required |
| Option             | optional character                               | -              | -   | 1          | Option   |
| Url                | Url of streaming contents                        | -              | -   | 1          | Option   |
| Sdp                | SDP session description                          | -              | -   | 1          | Option   |
| TransmissionMedium | XML fragment (Address element, PortList element) | medium         | Classification of transport medium ("RTP/AVP", "MMS", "CIP", "HTTP")        | 1          | Required |
|                    |  | protocol       | Classification of transport address ("UDP", "IEEE1394", "Bluetooth", "TCP") | 1          | Required |
|                    |  | type           | Classification of address "IPv4", "IPv6", "IEEE1394", "Bluetooth"           | 1          | Required |
| Address            | Streaming data transmission address              | -              | -   | 1          | Required |
| ResourceList       | XML fragment (Resource element)                  | -              | -   | 1          | Required |
| Resource           | Resources  | -              | -   | Multiple   | Required |
|                    |  | type           | Type of resource. ("port", "channel" )                                      | 1          | Required |
|                    |  | mimetype       | MIME type of media data   | 1          | Required |

Table4.1-4 Set value of Reason element of SetupResponse message

|   | Factor setting value | Summary   |
|---|----------------------|---|
| 1 | NotEnoughBandwidth   | Bandwidth required to transmit streaming data cannot be reserved. |
| 2 | UnsupportedTransport | The transport protocol is not supported.                          |
| 3 | PortUnavailable      | Port cannot be used.  |
| 4 | InternalError        | Internal error happened.  |

Values of the type attribute of the Setup message and the SetupResponse message are the same.

The Reason element is required when Result value is Failure. However, the Reason element should be absent when Result value is Success.

**Pucc Streaming Protocol**

The SetupResponse message is mapped by the Pucc core protocol as follows.

Table4.1-5 Mapping of SetupResponse message for Pucc Core Protocol SetupResponse

| Element name | Element value                           | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|---|----------------|---|------------|----------|
| Core         | XML fragment                            | -              | -   | 1          | Required |
|              |   | xmlns          | Name space identifier (URI) of Pucc Core Protocol | 1          | Required |
| ComType      | “Unicast”                               | -              | -   | 1          | Required |
| MsgID        | Message ID                              | -              | -   | 1          | Required |
| ReplyID      | Element value of MsgID of Setup message | -              | -   | 1          | Required |
| MsgType      | “Response”                              | -              | -   | 1          | Required |
| CommunityID  | Community ID                            | -              | -   | 1          | Option   |
| Source       | Source node ID                          | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)           | -              | -   | 1          | Required |
| Target       | Destination node ID                     | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)            | -              | -   | 1          | Option   |
| Route        | XML fragment (Route element)            | -              | -   | 1          | Required |
|              |   | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                              | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (SetupResponse element)    | -              | -   | 1          | Required |
|              |   | protocol       | Protocol ID of Pucc Streaming Control Protocol    | 1          | Required |

The following is an example of the SetupResponse message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:14:10Z@975870cd-f9bb-4305-1234-e65e29f6031</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>
  <MsgType>Response</MsgType>
  <CommunityID>urn:ED:Community:Pucc</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6031</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6031"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <SetupResponse xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl"
      type="initial">
      <Result>Success</Result>
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="UniDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
        <Repeat>
          <Interval>90000</Interval>
        </Repeat>
      </SetupResponse>
    </MsgBody>
  </Core>

```

**PUC Streaming Protocol**

```

    <Duration>900000</Duration>
  </Repeat>
  <MediaData>
    <MimeType>video/x-ms-wmv</MimeType>
    <MimeType>audio/x-ms-wma</MimeType>
  </MediaData>
  < Application name="WindowsMediaService9"/>
</Flow>
<RequestedMedium medium="RTP/AVP" protocol="UDP" type=""IPv4" />
<TransmissionMedium medium="RTP/AVP" protocol="UDP" type=""IPv4">
  <Address>192.168.0.1</Address>
  <ResourceList>
    <Resource type="port" mimetype="video/x-ms/wmv">9900</Resource>
    <Resource type="port" mimetype="video/x-ms/wma">9902</Resource>
  </ResourceList>
</TransmissionMedium>
</SetupResponse>
</MsgBody>
</Core>

```

Figure4.1-7 Ex1:Mapping of SetupResponse message

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:13:26Z@975870cd-f9bb-4305-1234-e65e29f6030</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>
  <MsgType>Response</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6030</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6029</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6030"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <SetupResponse xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl" type="reset">
      <Result>Failure</Result>
      <Reason>NotEnoughBandwidth</Reason>
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="BiDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
        <Repeat>
          <Interval>90000</Interval>
          <Duration>900000</Duration>
        </Repeat>
        <MediaData>
          <MimeType>video/x-ms-wmv</MimeType>
          <MimeType>audio/x-ms-wma</MimeType>
        </MediaData>
        < Application name="WindowsMediaService9"/>
      </Flow>
      <RequestedMedium medium="RTP/AVP" protocol="UDP" type=""IPv4" />
    </SetupResponse>
  </MsgBody>
</Core>

```

Figure4.1-8 Ex2:Mapping of SetupResponse message

## 6.2. Release Method

The Release method is used in order to release the streaming data transmission channel from the Sender node to the Receiver node.

This method consists of the following messages.

- ◆ Advertise Type : Release Message

Message exchange is performed by single-hop unicast. This message is transmitted and received among the following nodes.

- From the Sender to the Relay/Receiver
- From the Relay to the Relay/Receiver
- From the Receiver to the Relay/Receiver

The sequence of Release method is shown in the illustrated model below as an example. (The arrow means the direction where streaming data flows.)

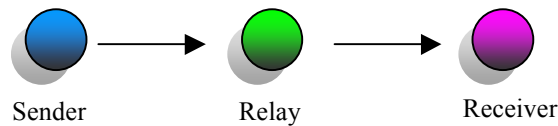


Figure4.2-1 Ex: Network model of Release method



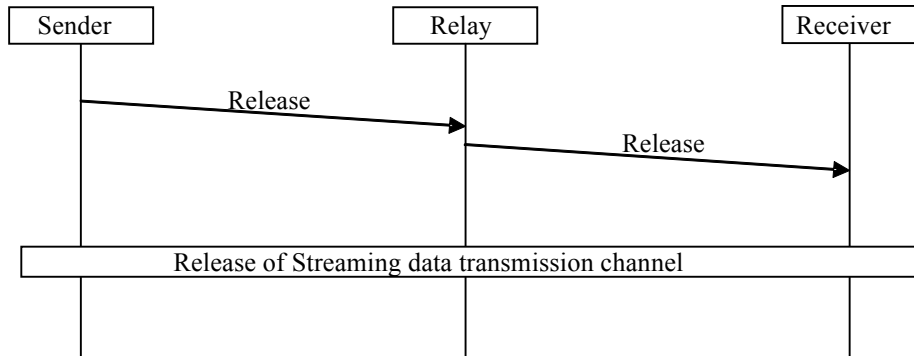


Figure4.2-2 Sequence of Release method

The Sender sets up a Flow ID and transmits a Release message. A transmitting route is not set to the Release message. Each node transmits the message to a downstream adjacent node according to the table of the streaming data transmission route which self has. The node which received the Release message releases a streaming data transmission channel. Furthermore, this node relays a message according to the table of a streaming data transmission route. Finally, the Release message reaches to the Receiver. In this manner, a streaming data transmission channel is released.

*The parameter of the Release message is defined as follows.*

Table4.2-1 Release message

| Element name | Element value                       | Attribute name | Attribute value   | Occurrence | Status   |
|--------------|-------------------------------------|----------------|---|------------|----------|
| Release      | XML fragment<br>(FlowID element)    | -              | -   | 1          | Required |
|              |                                     | xmlns          | Name space identifier(URI) of PUCC Streaming Control Protocol | 1          | Required |
| FlowID       | ID for discriminating flow uniquely | -              | -   | 1          | Required |

**PUCC Streaming Protocol**

The Release message is mapped by the PUCC core protocol as follows.

Table4.2-2 Mapping of Release message for PUCC Core Protocol

| Element name | Element value                                | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|--|----------------|---|------------|----------|
| Core         | XML fragment                                 | -              | -   | 1          | Required |
|              |  | xmlns          | Name space identifier (URI) of PUCC Core Protocol | 1          | Required |
| ComType      | “Unicast”                                    | -              | -   | 1          | Required |
| MsgID        | Message ID                                   | -              | -   | 1          | Required |
| MsgType      | “Advertise”                                  | -              | -   | 1          | Required |
| CommunityID  | Community ID                                 | -              | -   | 1          | Option   |
| Source       | Source node ID                               | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)                | -              | -   | 1          | Required |
| Target       | Destination node ID                          | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)                 | -              | -   | 1          | Option   |
| Route        | XML fragment (Target element, Route element) | -              | -   | 1          | Required |
|              |  | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                                   | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (Release element)               | -              | -   | 1          | Required |
|              |  | protocol       | Protocol ID of PUCC Streaming Control Protocol    | 1          | Required |

The following is an example of the Release message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:23:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Advertise</MsgType>
  <CommunityID>urn:ED:Community:PUCC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <Release xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
    </Release>
  </MsgBody>
</Core>

```

Figure4.2-5 Ex: Mapping of Release message

**6.3. Abort Method**

The Abort method is used in order to notify the Sender/Relay/Receiver nodes which adjoin in the upstream direction of the stop of the streaming.

This method consists of the following messages.

***PUCC Streaming Protocol***

◆ Advertise Type : Abort Message

Message exchange is performed by single-hop unicast. This message is transmitted and received among the following nodes.

- From Relay to the Sender/Relay/Receiver

**(1) In case of Unicast Streaming**

The sequence of the Abort method in the case of unicast is shown in the illustrated model below as an example. (The arrow means the direction where streaming data flows.)

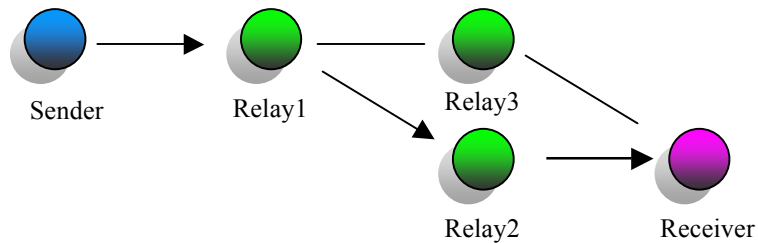


Figure4.3-1 Ex:Network model of Abort method –Unicast

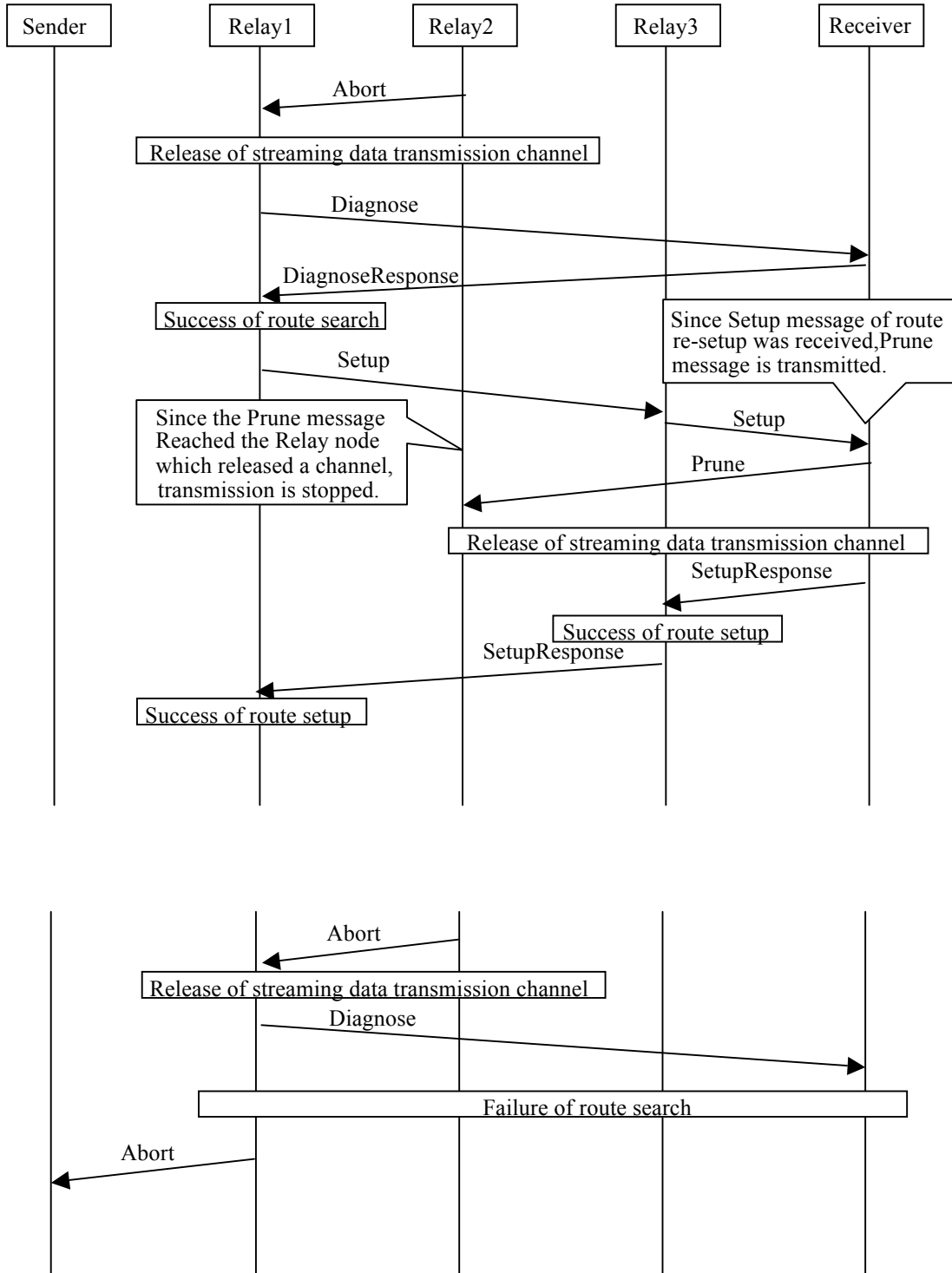


Figure4.3-2 Sequence of Abort method –Unicast

The Relay which ends the relay of streaming data sets up a Flow ID and transmits an Abort message to an upstream adjacent node. Furthermore, the streaming data transmission channel among upstream adjacent nodes is released. The node which has received the Abort message searches for a route to the last transmission node (Receiver) of the Setup message by the Diagnose method, and tries to set up to a new route. In this case, Setup message is set up as follows.

- Flow element : Same flow ID
- Source element : Self-node
- Destination element : New route
- Setup element – Type attribute : reset

When the route set up with the same flow ID exists, the node returns a SetupResponse. The node which received the Setup message for a route re-setup transmits the Prune message for cutting the unnecessary old course. When the search or setup of a route goes wrong, the node which has received the Abort message performs the following processing.

- The node which has received the Abort message : Relay node

The Relay node transmits the Abort message to the adjacent node in the Sender's direction.

- The node which has received the Abort message : Receiver (the end of setup)

Receiver processes nothing.

- The node which has received the Abort message : The Sender (the start of setup)

The Sender stops the flow setup and notifies the user's application of a failure in the flow setup.

**(2) In case of Multicast/Broadcast Streaming**

The sequence of the Abort method in the case of multicast/broadcast is shown in the illustrated model as an example. (The arrow means the direction where streaming data flows.)

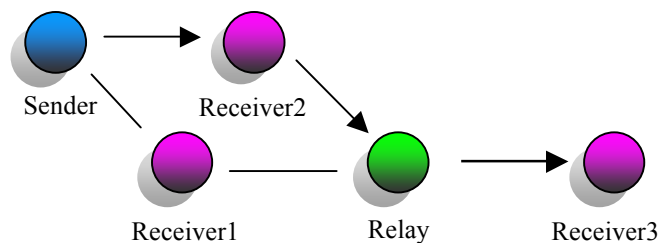


Figure4.3-3 Ex:Network model of Abort method –Multicast/Broadcast

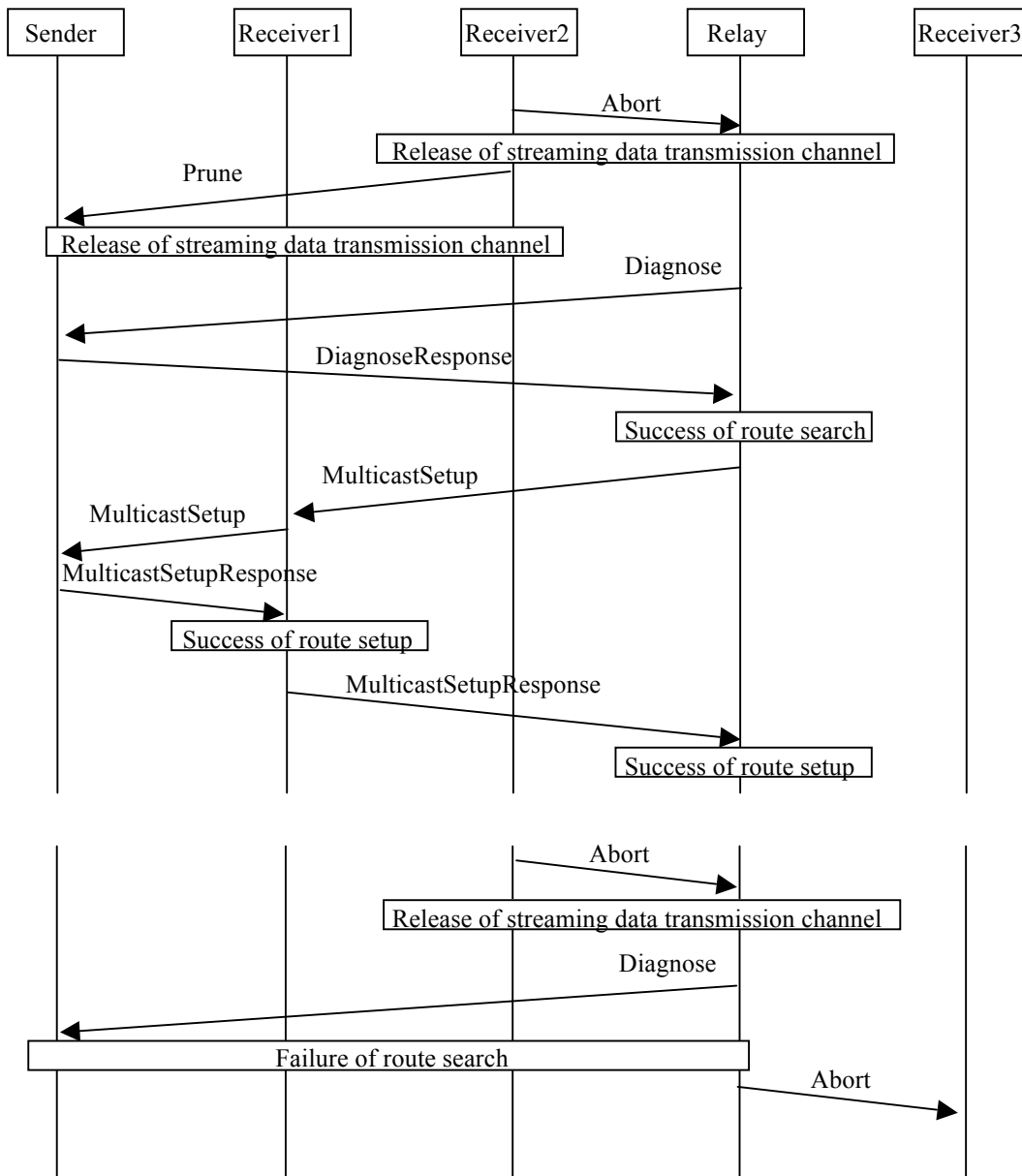


Figure4.3-4 Sequence of Abort method –Multicast/Broadcast

The Relay node which ends the relay of streaming data sets up the Flow ID and transmits an Abort message to a downstream adjacent node. Since this node turns into a relay node at the end of the route as a result of being cut by the Abort message, it transmits the Prune message. Furthermore, the streaming data transmission channel among a downstream adjacent node is released. The node which has received the Abort message searches for a route to the destination node (the Sender) of the MulticastSetup message by the Diagnose method, and tries to set up to a new route. In this case, the Setup message is set up as follows.

**PUCC Streaming Protocol**

- Flow element : Same flow ID
- Source element : Self-node
- Destination element : New route
- Setup element – Type attribute : reset

When the route set up with the same flow ID exists, the node returns a MulticastSetupResponse.

When the search or setup of a route goes wrong, the node which has received an Abort message performs the following processing.

- The node which has received the Abort message : The Relay node

The Relay node transmits the Abort message to the adjacent node in the Receiver’s direction.

- The node which has received the Abort message : The Sender (the end of MulticastSetup)

The Sender processes nothing.

- The node which has received the Abort message : The Receiver (the start of MulticastSetup)

Receiver stops a flow setup and notifies failure in a flow setup to user application.

The parameter of the Abort message is defined as follows.

Table4.3-1 Abort message

| Element name | Element value                       | Attribute name | Attribute value  | Occurrence | Status   |
|--------------|-------------------------------------|----------------|--|------------|----------|
| Abort        | XML fragment (FlowID element)       | -              | -  | 1          | Required |
|              |                                     | xmlns          | Name space identifier (URI) of PUCC Streaming Control Protocol | 1          | Required |
| FlowID       | ID for discriminating flow uniquely | -              | -  | 1          | Required |

**PUCC Streaming Protocol**

The Abort message is mapped by the PUCC core protocol as follows.

Table4.3-2 Mapping of Abort message for PUCC Core Protocol

| Element name | Element value                                | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|--|----------------|---|------------|----------|
| Core         | XML fragment                                 | -              | -   | 1          | Required |
|              |  | xmlns          | Name space identifier (URI) of PUCC Core Protocol | 1          | Required |
| ComType      | “Unicast”                                    | -              | -   | 1          | Required |
| MsgID        | Message ID                                   | -              | -   | 1          | Required |
| MsgType      | “Advertise”                                  | -              | -   | 1          | Required |
| CommunityID  | Community ID                                 | -              | -   | 1          | Option   |
| Source       | Source node ID                               | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)                | -              | -   | 1          | Required |
| Target       | Destination node ID                          | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)                 | -              | -   | 1          | Option   |
| Route        | XML fragment (Target element, Route element) | -              | -   | 1          | Required |
|              |  | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                                   | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (Abort element)                 | -              | -   | 1          | Required |
|              |  | protocol       | Protocol ID of PUCC Streaming Control Protocol    | 1          | Required |

The following is an example of the Setup message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:23:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Advertise</MsgType>
  <CommunityID>urn:ED:Community:PUCC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <Abort xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
    </Abort>
  </MsgBody>
</Core>

```

Figure4.3-5 Ex:Mapping of Abort message



#### 6.4. Prune Method

The Prune method is used in order to notify the end of the relay. The Relay node located at the end of a streaming data route notifies an upstream node of the secession from the flow.

This method consists of the following messages.

- ◆ Advertise Type : Prune Message

Message exchange is performed by single-hop unicast. This message is transmitted and received among the following nodes.

- From Relay to the Sender/Relay/Receiver

The sequence of the Prune method is shown in the illustrated model as an example. (The arrow means the direction where streaming data flows.)

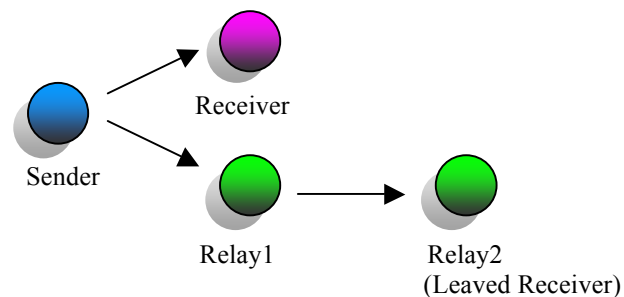


Figure4.4-1 Ex:Network model of Prune method

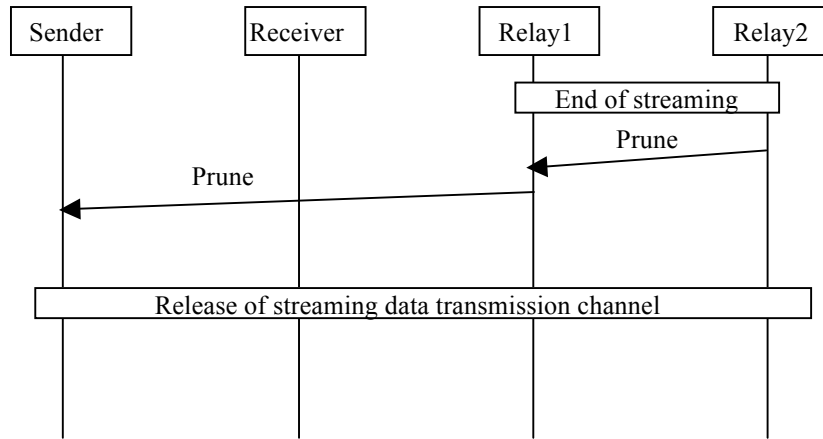


Figure4.4-2 Sequence of Prune method

The Relay node located at the end of a streaming data transmission route sets up a Flow ID and transmits a Prune message. A transmitting route is not set to the Prune message. Each node transmits the message to an upstream adjacent node according to the table of the streaming data transmission route which self has. Subsequently, the streaming data transmission channel among an upstream adjacent node is released.

The node which has received the Prune message transmits the Prune message to an upstream adjacent node, when fulfilling the following conditions.

- Any downstream adjacent node (Relay/Receiver) does not exist.
- The self-node has not received the streaming data.

The parameter of the prune message is defined as follows.

Table4.4-1 Prune message

| Element name | Element value                       | Attribute name | Attribute value  | Occurrence | Status   |
|--------------|-------------------------------------|----------------|--|------------|----------|
| Prune        | XML fragment (FlowID element)       | -              | -  | 1          | Required |
|              |                                     | xmlns          | Name space identifier (URI) of PUCC Streaming Control Protocol | 1          | Required |
| FlowID       | ID for discriminating flow uniquely | -              | -  | 1          | Required |

**PUCC Streaming Protocol**

The Prune message is mapped by the PUCC core protocol as follows.

Table4.4-2 Mapping of Prune message for PUCC Core Protocol

| Element name | Element value                                | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|--|----------------|---|------------|----------|
| Core         | XML fragment                                 | -              | -   | 1          | Required |
|              |  | xmlns          | Name space identifier (URI) of PUCC Core Protocol | 1          | Required |
| ComType      | “Unicast”                                    | -              | -   | 1          | Required |
| MsgID        | Message ID                                   | -              | -   | 1          | Required |
| MsgType      | “Advertise”                                  | -              | -   | 1          | Required |
| CommunityID  | Community ID                                 | -              | -   | 1          | Option   |
| Source       | Source node ID                               | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)                | -              | -   | 1          | Required |
| Target       | Destination node ID                          | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)                 | -              | -   | 1          | Option   |
| Route        | XML fragment (Target element, Route element) | -              | -   | 1          | Required |
|              |  | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                                   | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (Prune element)                 | -              | -   | 1          | Required |
|              |  | protocol       | Protocol ID of PUCC Streaming Control Protocol    | 1          | Required |

The following is an example of the prune message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:23:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Advertise</MsgType>
  <CommunityID>urn:ED:Community:PUCC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <Prune xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
    </Prune>
  </MsgBody>
</Core>

```

Figure4.4-3 Ex:Mapping of Prune message

### 6.5. Deliver Method

The Deliver method is used in order to perform the demand of Receiver initiative streaming from Receiver to the Sender.

This method consists of the following messages.

- ◆ Request Type : DeliverRequest Message
- ◆ Response Type : DeliverResponse Message

Message exchange is performed by multi-hop unicast. This message is transmitted and received among the following nodes.

- Between the Receiver and the Sender

The sequence of the Deliver method is shown in the illustrated model as an example. (The arrow means the direction where streaming data flows.)

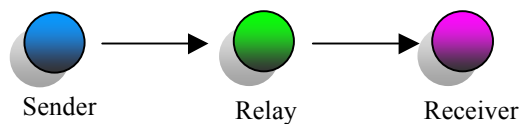


Figure4.5-1 Ex:Network model of Deliver method

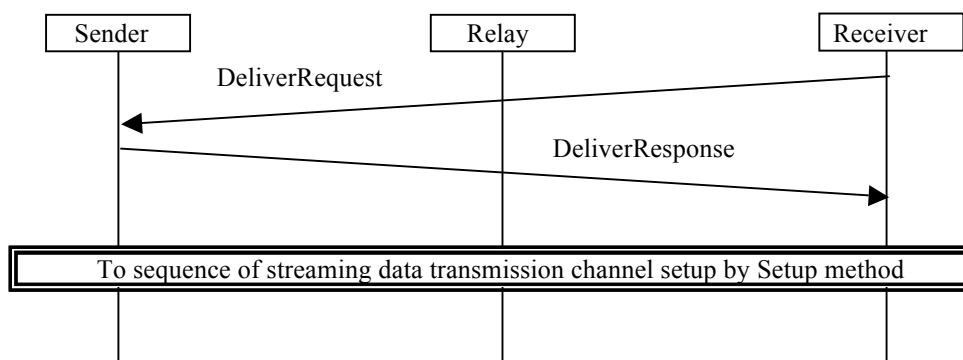


Figure4.5-2 Sequence of Deliver method

Receiver sets up the flow ID of the required flow, and transmits a DeliverRequest message to the Sender. The Sender which has received the DeliverRequest message judges whether the specified flow can be offered, and replies a DeliverResponse message. Then, the Sender sets a route as the reverse order of the

**PUCC Streaming Protocol**

transmission route of the DeliverRequest message. The Receiver-initiated streaming is started with such a flow.

The parameter of the DeliverRequest message is defined as follows.

Table4.5-1 DeliverRequest message

| Element name                   | Element value   | Attribute name | Attribute value  | Occurrence | Status   |
|--------------------------------|---|----------------|--|------------|----------|
| DeliverRequest                 | XML fragment (FlowID element, ApplicationSpecificParameter element)                       | -              | -  | 1          | Required |
|                                |   | xmlns          | Name space identifier (URI) of PUCC Streaming Control Protocol | 1          | Required |
| FlowID                         | ID for discriminating flow uniquely   | -              | -  | 1          | Required |
| Application Specific Parameter | XML fragment<br>ex. parameters for transcoding, for example, bitrate, fps, width, height. | -              | -  | 1          | Option   |
|                                |   | xmlns          | Name space identifier(URI) of application                      | 1          | Required |

The DeliverRequest message is mapped by the PUCC core protocol as follows.

Table4.5-2 Mapping of DeliverRequest message for PUCC Core Protocol

| Element name | Element value                         | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|---------------------------------------|----------------|---|------------|----------|
| Core         | XML fragment                          | -              | -   | 1          | Required |
|              |                                       | xmlns          | Name space identifier (URI) of PUCC Core Protocol | 1          | Required |
| ComType      | “Unicast”                             | -              | -   | 1          | Required |
| MsgID        | Message ID                            | -              | -   | 1          | Required |
| MsgType      | “Request”                             | -              | -   | 1          | Required |
| CommunityID  | Community ID                          | -              | -   | 1          | Option   |
| Source       | Source node ID                        | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)         | -              | -   | 1          | Required |
| Target       | Relay node ID                         | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)          | -              | -   | 1          | Required |
|              |                                       | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                            | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (DeliverRequest element) | -              | -   | 1          | Required |
|              |                                       | protocol       | Protocol ID of PUCC Streaming Control Protocol    | 1          | Required |

The following is an example of the DeliverRequest message.

```
<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Request</MsgType>
  <CommunityID>urn:ED:Community:PUCC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029" />
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <DeliverRequest xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
    </DeliverRequest>
  </MsgBody>
</Core>
```

Figure4.5-3 Ex:Mapping of DeliverRequest message

The parameter of the DeliverResponse message is defined as follows.

Table4.5-3 DeliverResponse message

| Element name    | Element value   | Attribute name | Attribute value   | Occurrence | Status   |
|-----------------|---|----------------|---|------------|----------|
| DeliverResponse | XML fragment (Result element, Reason element, FlowID element)         | -              | -   | 1          | Required |
|                 |   | xmlns          | Name space identifier (URI) of P UCC Streaming Control Protocol | 1          | Required |
| Result          | "Success", "Failure"  | -              | -   | 1          | Required |
| Reason          | "FlowUnavailable", "WrongFlowID", "NotAuthenticated", "InternalError" | -              | -   | 1          |          |
| FlowID          | ID for discriminating flow uniquely                                   | -              | -   | 1          | Required |

Table4.5-4 Setting values of Reason element of DeliverResponse message

|   | Factor setting value | Summary                            |
|---|----------------------|------------------------------------|
| 1 | FlowUnavailable      | The specified flow cannot be used. |
| 2 | WrongFlowID          | The specified flow ID is mistaken. |
| 3 | NotAuthenticated     | Authentication failed.             |
| 4 | InternalError        | Internal error happened.           |

**PUC Streaming Protocol**

The DeliverResponse message is mapped by the PUC core protocol as follows.

Table4.5-5 Mapping of DeliverResponse message for PUC Core Protocol

| Element name | Element value                                    | Attribute name | Attribute value                                  | Occurrence | Status   |
|--------------|--|----------------|--|------------|----------|
| Core         | XML fragment                                     | -              | -  | 1          | Required |
|              |  | xmlns          | Name space identifier (URI) of PUC Core Protocol | 1          | Required |
| ComType      | “Unicast”  | -              | -  | 1          | Required |
| MsgID        | Message ID                                       | -              | -  | 1          | Required |
| ReplyID      | Element value of MsgID of DeliverRequest message | -              | -  | 1          | Required |
| MsgType      | “Response”                                       | -              | -  | 1          | Required |
| CommunityID  | Community ID                                     | -              | -  | 1          | Option   |
| Source       | Source node ID                                   | -              | -  | 1          | Required |
| Destination  | XML fragment (Target element)                    | -              | -  | 1          | Required |
| Target       | Destination node ID                              | -              | -  | 1          | Required |
| TraceRoute   | XML fragment (Route element)                     | -              | -  | 1          | Option   |
| Route        | XML fragment (Route element)                     | -              | -  | 1          | Required |
|              |  | Node           | Relay node ID                                    | 1          | Required |
| SessionID    | Session ID                                       | -              | -  | 1          | Option   |
| MsgBody      | XML fragment (DeliverResponse element)           | -              | -  | 1          | Required |
|              |  | protocol       | Protocol ID of PUC Streaming Control Protocol    | 1          | Required |

The following is an example of the DeliverResponse message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:14:10Z@975870cd-f9bb-4305-1234-e65e29f6031</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>
  <MsgType>Response</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6031</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6031"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <DeliverResponse xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <Result>Success</Result>
      <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
    </DeliverResponse>
  </MsgBody>
</Core>

```

Figure4.5-4 Ex:Mapping of DeliverResponse message(1)

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:13:26Z@975870cd-f9bb-4305-1234-e65e29f6030</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>

```

**PUCC Streaming Protocol**

```

<MsgType>Response</MsgType>
<CommunityID>urn:ED:Community:PUCC</CommunityID>
<Source>975870cd-f9bb-4305-1234-e65e29f6030</Source>
<Destination>
  <Target>975870cd-f9bb-4305-1234-e65e29f6029</Target>
</Destination>
<TraceRoute>
  <Route Node=""975870cd-f9bb-4305-1234-e65e29f6030""/>
</TraceRoute>
<MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
  <DeliverResponse xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <Result>Failure</Result>
    <Reason>WrongFlowID</Reason>
    <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
  </DeliverResponse>
</MsgBody>
</Core>

```

Figure4.5-5 Ex:Mapping of DeliverResponse message(2)

**6.6. Keepalive Method**

The Keepalive method is used in order to maintain the flow under streaming data transmission in PUCC unicast streaming. Each node on the route of the flow exchanges messages with the adjacent nodes which have transmitted the Setup message.

This method consists of the following messages.

- ◆ Request Type : Keepalive Message
- ◆ Response Type : KeepaliveResponse Message

Message exchange is performed by single-hop unicast. This message is transmitted and received among the following nodes.

- Between the Sender and the Relay/Receiver
- Between the Relay and Relay/Receiver
- Between the Receiver and the Relay/Receiver

The sequence of the Keepalive method is shown in the illustrated model as an example. (The arrow means the direction where streaming data flows.)



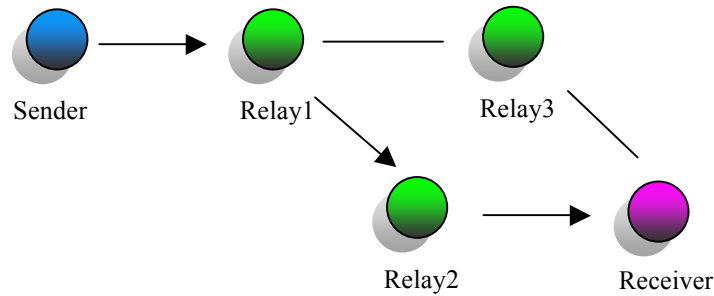


Figure4.6-1 Ex:Network model of Keepalive method

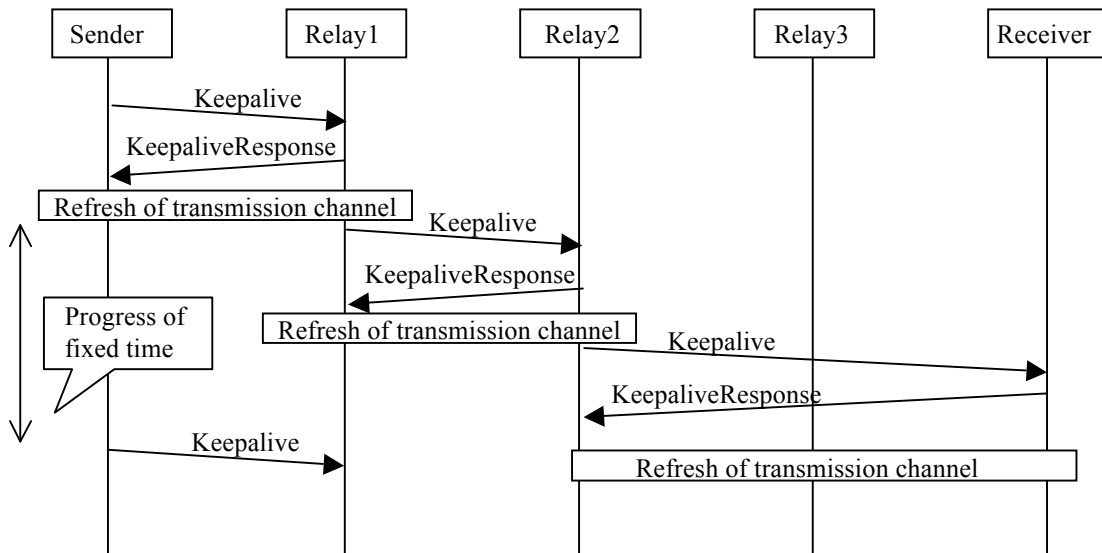


Figure4.6-2 Sequence of Keepalive method –In case of success to Keepalive

Each node which has set up the flow transmits the Keepalive message to a downstream adjacent node periodically. The Flow ID is set to the Keepalive message. The node which has received the Keepalive message resets the soft state timer of the streaming data transmission channel among upstream adjacent nodes, and replies with a KeepaliveResponse message. All the nodes on the route maintain the flow by refreshing the transmission channel with the adjacent nodes periodically.

**PUCC Streaming Protocol**

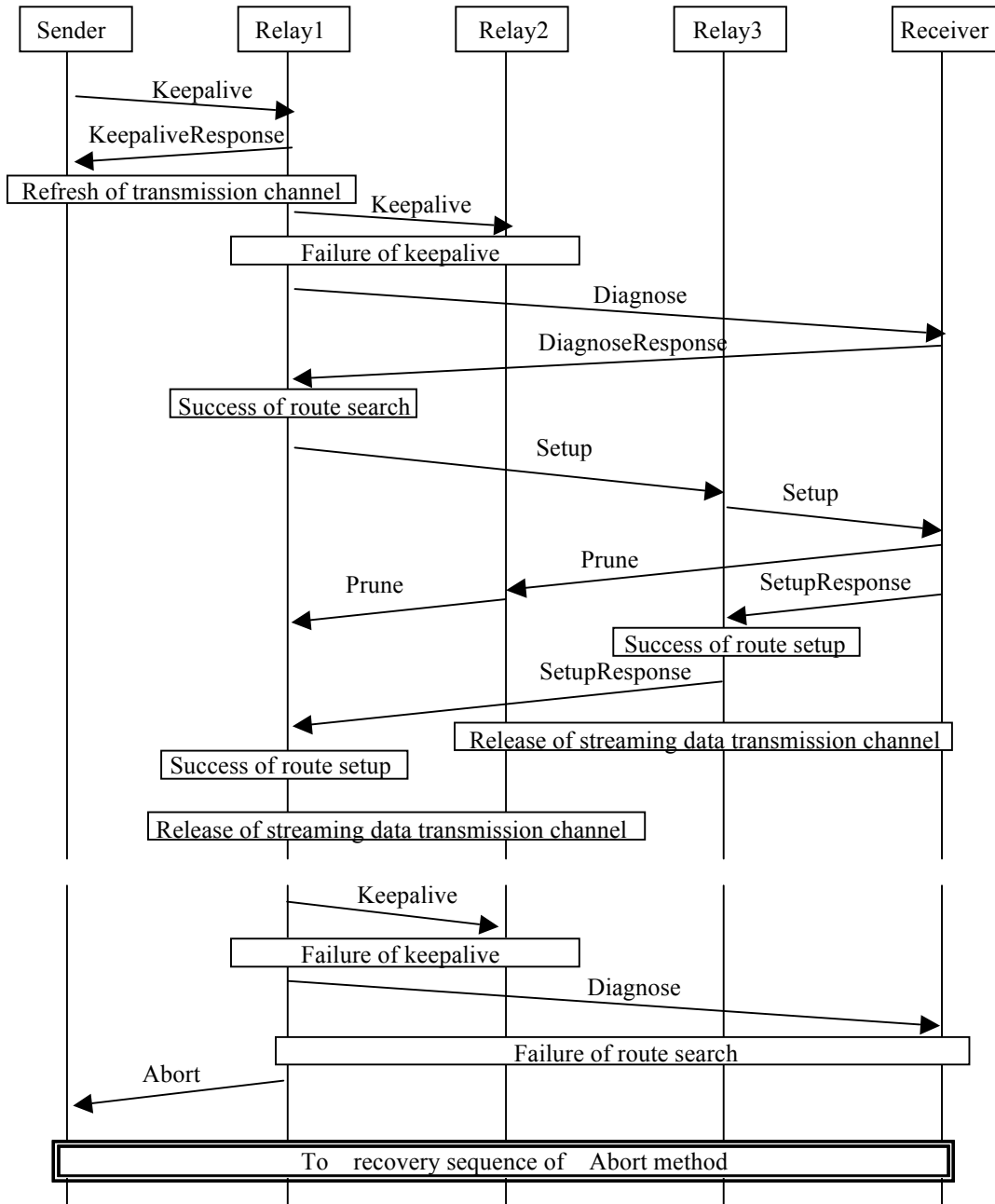


Figure4.6-3 Sequence of Keepalive method –In case of failure to Keepalive

***PUCC Streaming Protocol***

When a Keepalive goes wrong among the arbitrary nodes on the route, the node which has detected the failure tries to set up a flow in a new route to the Receiver. In this case, the Setup message is set up as follows.

- Flow element : Same flowID
- Source element : Self-node
- Destination element : New route
- Setup element – Type attribute : reset

When a route set up with the same flow ID exists, the node returns a SetupResponse. The node which has received the Setup message for a route re-setup transmits the Prune message for cutting the unnecessary old course. When the search or setup of a route goes wrong, the node which has detected the failure performs the following processing.

- The node which has detected the failure : Relay node

The Relay node transmits the Abort message to the adjacent node in the Sender’s direction.

- The node which has detected the failure : Receiver (the end of setup)

The Receiver processes nothing.

- The node which has detected failure : Sender (the start of setup)

The Sender stops a flow setup and notifies the user’s application of the failure in the flow setup.

The parameter of the Keepalive message is defined as follows.

Table4.6-1 Keepalive message

| Element name | Element value                       | Attribute name | Attribute value  | Occurrence | Status   |
|--------------|-------------------------------------|----------------|--|------------|----------|
| Keepalive    | XML fragment (FlowID element)       | -              | -  | 1          | Required |
|              |                                     | xmlns          | Name space identifier (URI) of PUCC Streaming Control Protocol | 1          | Required |
| FlowID       | ID for discriminating flow uniquely | -              | -  | 1          | Required |

**PUCC Streaming Protocol**

The Keepalive message is mapped by the PUCC core protocol as follows.

Table4.6-2 Mapping of Keepalive message for PUCC Core Protocol

| Element name | Element value                    | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|----------------------------------|----------------|---|------------|----------|
| Core         | XML fragment                     | -              | -   | 1          | Required |
|              |                                  | xmlns          | Name space identifier (URI) of PUCC Core Protocol | 1          | Required |
| ComType      | “Unicast”                        | -              | -   | 1          | Required |
| MsgID        | Message ID                       | -              | -   | 1          | Required |
| MsgType      | “Request”                        | -              | -   | 1          | Required |
| CommunityID  | Community ID                     | -              | -   | 1          | Option   |
| Source       | Source node ID                   | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)    | -              | -   | 1          | Required |
| Target       | Destination nodeID               | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)     | -              | -   | 1          | Option   |
|              |                                  | -              | -   | 1          | Required |
| Route        | XML fragment (Route element)     | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                       | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (Keepalive element) | -              | -   | 1          | Required |
|              |                                  | protocol       | Protocol ID of PUCC Streaming Control Protocol    | 1          | Required |

The following is an example of Keepalive message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Request</MsgType>
  <CommunityID>urn:ED:Community:PUCC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029" />
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <Keepalive xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
    </Keepalive>
  </MsgBody>
</Core>

```

Figure4.6-4 Ex:Mapping of Keepalive message

**Pucc Streaming Protocol**

The parameter of the KeepaliveResponse message is defined as follows.

Table4.6-3 KeepaliveResponse message

| Element name      | Element value                       | Attribute name | Attribute value  | Occurrence | Status   |
|-------------------|-------------------------------------|----------------|--|------------|----------|
| KeepaliveResponse | XML fragment (FlowID element)       | -              | -  | 1          | Required |
|                   |                                     | xmlns          | Name space identifier (URI) of Pucc Streaming Control Protocol | 1          | Required |
| FlowID            | ID for discriminating flow uniquely | -              | -  | 1          | Required |

The KeepaliveResponse message is mapped by the Pucc core protocol as follows.

Table4.6-5 Mapping of KeepaliveResponse message for Pucc Core Protocol

| Element name | Element value                            | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|--|----------------|---|------------|----------|
| Core         | XML fragment                             | -              | -   | 1          | Required |
|              |  | xmlns          | Name space identifier (URI) of Pucc Core Protocol | 1          | Required |
| ComType      | “Unicast”                                | -              | -   | 1          | Required |
| MsgID        | Message ID                               | -              | -   | 1          | Required |
| ReplyID      | Element value of MsgID of Setup message  | -              | -   | 1          | Required |
| MsgType      | “Response”                               | -              | -   | 1          | Required |
| CommunityID  | Community ID                             | -              | -   | 1          | Option   |
| Source       | Source node ID                           | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)            | -              | -   | 1          | Required |
| Target       | Destination node ID                      | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)             | -              | -   | 1          | Option   |
| Route        | XML fragment (Route element)             | -              | -   | 1          | Required |
|              |  | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                               | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (KeepaliveResponse element) | -              | -   | 1          | Required |
|              |  | protocol       | Protocol ID of Pucc Streaming Control Protocol    | 1          | Required |

The following is an example of the KeepaliveResponse message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:14:10Z@975870cd-f9bb-4305-1234-e65e29f6031</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>
  <MsgType>Response</MsgType>
  <CommunityID>urn:ED:Community:Pucc</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6031</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>

```

```

<TraceRoute>
  <Route Node=""975870cd-f9bb-4305-1234-e65e29f6031""/>
</TraceRoute>
<MsgBody protocol=""http://www.pucc.jp/2012/03/p2p_stream_ctrl"">
  <KeepaliveResponse xmlns="" http://www.pucc.jp/2012/03/p2p_stream_ctrl"">
    <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
  </KeepaliveResponse>
</MsgBody>
</Core>

```

Figure4.6-5 Ex:Mapping of KeepaliveResponse message

### 6.7. FlowInformationAdvertise Method

This method is used in order to advertise flow information from the Sender to the Receiver (multicast member or community member).

This method consists of the following messages.

- ◆ Advertise Type : FlowInformationAdvertise Message

Message exchange is performed by unicast. This message is transmitted and received among the following nodes hop by hop.

- Between the Sender and the Relay/Receiver

#### (1) In case of Receiver initiative streaming

The sequence of the FlowInformationAdvertise method in the case of Receiver initiative streaming is shown in the illustrated model as an example. (The arrow means the direction where streaming data flows.)

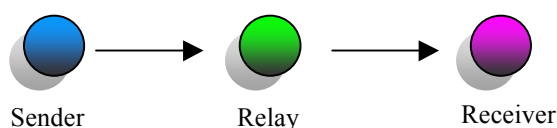


Figure4.7-1 Ex:Network model of FlowInformationAdvertise –Receiver initiative streaming

**PUCC Streaming Protocol**

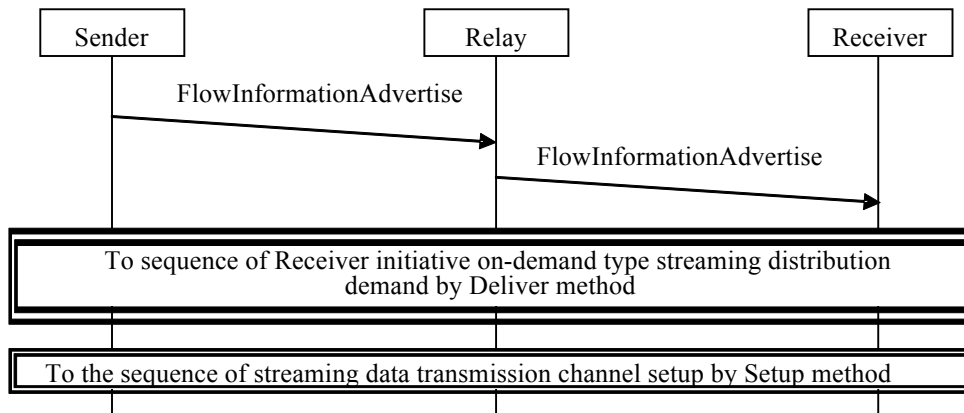


Figure4.7-2 Sequence of FlowInformationAdvertise method –Receiver initiative streaming

The Sender sets up the flow information announced publicly and transmits a FlowInformationAdvertise message. A flow ID which discriminates the flow, the name of the flow, the start/end time, the MIME type of media data, optional information of the Sender application, the flow classification, the directivity of the flow, band width information and so on is included in the flow information. An SDP description may be embedded in the flow information instead of the name of the flow, the start/end time, the MIME type of media data, optional information of the Sender application, band width information and so on. Note that any node that is hopped shall modify its SDP description appropriately since SDP is only applicable to single hops. Moreover, more than one set of flow information can also be set up.

The Receiver which has received the FlowInformationAdvertise message chooses the flow of which flow classification is the "Receiver". Furthermore, the Receiver requires the Receiver initiative streaming distribution by the Deliver method. After that, the Sender sets up a flow in the route to the Receiver by Setup method, and starts streaming.

**(2) In case of Sender-initiated streaming**

The sequence of the Abort method in the case of Sender initiative streaming is shown in the illustrated model as an example. (The arrow means the direction where streaming data flows.)

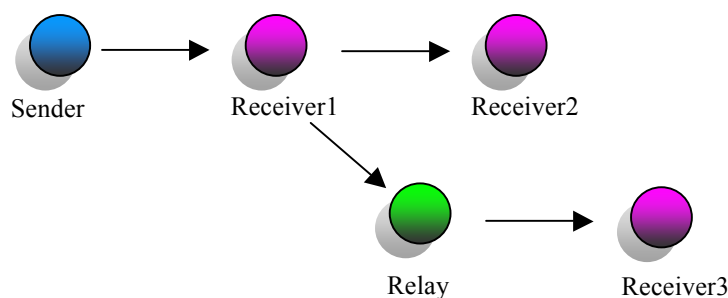


Figure4.7-3 Ex:Network model of FlowInformationAdvertise method –Sender initiative streaming

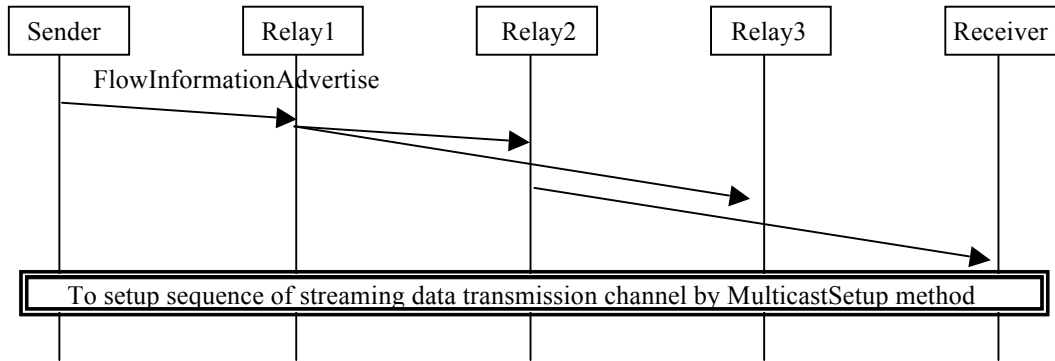


Figure4.7-4 Sequence of FlowInformationAdvertise method –initiative streaming

The Sender sets up the flow information announced publicly and transmits a FlowInformationAdvertise message. A flow ID which identifies the flow, the name of the flow, the start/end time, the MIME type of media data, optional information of the Sender application, the flow classification, the directivity of the flow, band width information and so on is included in the flow information. An SDP description may be embedded in the flow information instead of the name of the flow, the start/end time, the MIME type of media data, optional information of the Sender application, band width information and so on. Note that as any node that is hopped shall modify its SDP description appropriately since SDP is only applicable to single hops. Moreover, more than one set of flow information can also be set up.

The Receiver which has received the FlowInformationAdvertise message chooses the flow which flow classification is the "Sender". Furthermore, the Receiver sets up a flow by the MulticastSetup method, and starts the Sender initiative type streaming.



**Pucc Streaming Protocol**

The parameter of the FlowInformationAdvertise message is defined as follows.

Table4.7-1 FlowInformationAdvertise message

| Element name             | Element value   | Attribute name | Attribute value  | Occurrence | Status   |
|--------------------------|---|----------------|--|------------|----------|
| FlowInformationAdvertise | XML fragment (Flow element, TraceRoute element)   | -              | -  | 1          | Required |
|                          |   | xmlns          | Name space identifier (URI) of Pucc Streaming Control Protocol                         | 1          | Required |
| Flow                     | XML fragment (Name element, Description element, Sender element, Bandwidth element, ActiveTime element, Repeat element, MediaData element, Application element, Url element, Sdp element) | -              | -  | Multiple   | Required |
|                          |   | ID             | ID for discriminating flow uniquely (<sequence number> <Data><Time>”@”<Sender nodeID>) | 1          | Required |
|                          |   | initiative     | Initiative node of setup start (“Sender”, “Receiver”)                                  | 1          | Option   |
|                          |   | directionality | Direction of flow (“UniDirectional”, “BiDirectional”)                                  | 1          | Option   |
| Name                     | Name of flow  | -              | -  | 1          | Option   |
| Description              | Description of flow   | -              | -  | 1          | Option   |
| Sender                   | Sender node ID  | -              | -  | 1          | Option   |
| Bandwidth                | Band information (numerical value[bps])   | -              | -  | 1          | Option   |
| ActiveTime               | XML fragment (Start element, End element)   | -              | -  | 1          | Option   |
| Start                    | Starting time of flow (ISO8601 form)  | -              | -  | 1          | Required |
| End                      | Ending time of flow (ISO8601 form)  | -              | -  | 1          | Required |
| Repeat                   | XML fragment (Interval element, Duration element)   | -              | -  | 1          | Option   |
| Interval                 | Time interval of repetition(Unit[ms])   | -              | -  | 1          | Required |
| Duration                 | Active duration(Unit[ms])   | -              | -  | 1          | Required |
| MediaData                | XML fragment (MimeType element)   | -              | -  | 1          | Option   |
| MimeType                 | MIME type of media data   | -              | -  | Multiple   | Required |
| Application              | XML fragment (Option element)   | -              | -  | 1          | Option   |
|                          |   | name           | Name of application which distributes media data                                       | 1          | Required |
| Option                   | Arbitrary characters  | -              | -  | 1          | Option   |
| Url                      | Url of streaming contents   | -              | -  | 1          | Option   |
| Sdp                      | SDP session description   | -              | -  | 1          | Option   |
| TraceRoute               | XML fragment (Route element)  | -              | -  | 1          | Option   |
| Route                    | XML fragment (Route element)  | -              | -  | 1          | Option   |
|                          |   | Node           | Sender node ID, Relay node ID or Receiver node ID                                      | 1          | Required |

**PUC Streaming Protocol**

The FlowInformationAdvertise message is mapped by the PUC core protocol as follows.

Table4.7-2 Mapping of FlowInformationAdvertise message for PUC Core Protocol

| Element name | Element value                                   | Attribute name | Attribute value                                  | Occurrence | Status   |
|--------------|---|----------------|--|------------|----------|
| Core         | XML fragment                                    | -              | -  | 1          | Required |
|              |   | xmlns          | Name space identifier (URI) of PUC Core Protocol | 1          | Required |
| ComType      | “Unicast”                                       | -              | -  | 1          | Required |
| MsgID        | Message ID                                      | -              | -  | 1          | Required |
| MsgType      | “Advertise”                                     | -              | -  | 1          | Required |
| CommunityID  | Community ID                                    | -              | -  | 1          | Option   |
| Source       | Source node ID                                  | -              | -  | 1          | Required |
| Destination  | XML fragment (Target element)                   | -              | -  | 1          | Required |
| Target       | Destination node ID                             | -              | -  | 1          | Required |
| TraceRoute   | XML fragment (Route element)                    | -              | -  | 1          | Required |
| Route        | XML fragment (Route element)                    | -              | -  | 1          | Required |
|              |   | Node           | Relay node ID                                    | 1          | Required |
| SessionID    | Session ID                                      | -              | -  | 1          | Option   |
| MsgBody      | XML fragment (FlowInformationAdvertise element) | -              | -  | 1          | Required |
|              |   | protocol       | Protocol ID of PUC Streaming Control Protocol    | 1          | Required |

The following is an example of FlowInformationAdvertise message.

```

<Core xmlns="http://www.puc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:23:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Advertise</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029" />
  </TraceRoute>
  <MsgBody protocol="http://www.puc.jp/2012/03//p2p_stream_ctrl">
    <FlowInformationAdvertise xmlns="http://www.puc.jp/2012/03//p2p_stream_ctrl">
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="UniDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
        <Repeat>
          <Interval>01:30.00</Interval>
          <Duration>15:00.00</Duration>
        </Repeat>
        <MediaData>
          <MimeType>video/x-ms-wmv</MimeType>
          <MimeType>audio/x-ms-wma</MimeType>
        </MediaData>
        <Application name="WindowsMediaService9"/>
      </Flow Information Advertise>
    </MsgBody>
  </Core>

```

**PUC Streaming Protocol**

```

</Flow>
<Flow ID="00002.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Receiver" directionality="UniDirectional">
  <Name>Video Delivery2</Name>
  <Description>Video Delivery2</Description>
  <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
  <Bandwidth>384000</Bandwidth>
  <ActiveTime>
    <Start>2004-01-20T22:00:00-09:00</Start>
    <End>2004-01-20T22:30:00-09:00</End>
  </ActiveTime>
  <Repeat>
    <Interval>90000</Interval>
    <Duration>900000</Duration>
  </Repeat>
  <MediaData>
    <MimeType>video/x-ms-wmv</MimeType>
    <MimeType>audio/x-ms-wma</MimeType>
  </MediaData>
  <Application name="WindowsMediaService9"/>
</Flow>
<TraceRoute>
  <Route Node="975870cd-f9bb-4305-1234-e65e29f6029"/>
</TraceRoute>
</FlowInformationAdvertise>
</MsgBody>
</Core>

```

Figure4.7-5 Ex: Mapping of FlowInformationAdvertise message

The following is an example of FlowInformationAdvertise message including SDP.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:23:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Advertise</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6030</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6031</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029">
      <Route Node="975870cd-f9bb-4305-1234-e65e29f6030"/>
    </Route>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <FlowInformationAdvertise xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
              initiative="Sender" directionality="UniDirectional">
        <Sdp>
          v=0
          o=ghost 2890844526 2890842807 IN IP4 192.168.11.3
          s=Video Delivery
          c=IN IP4 238.0.0.1
          t=3410417504 4415605104
          b=AS:77
          a=source-filter: incl IN IP4 192.168.11.3
          m=video: 4002 RTP/AVP 96
          b=TIAS:62000
          b=RR:0
          b=RS:600
          a=rtpmap:96 H264/90000
          a=fmtp:96 packetization-mode=1;
          a=framerate:10
          a=framesize:96 320-240

```

```

m=audio 4004 RTP/AVP 98
b=TIAS:15120
b=RR:0
b=RS:600
a=rtpmap:98 mpeg4-generic/48000 2
a=fmtp:98 streamtype=5; profile-level-id=16; mode=AAC-hbr;
</sdp>
<Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
</Flow>
<TraceRoute>
  <Route Node="975870cd-f9bb-4305-1234-e65e29f6029">
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6030"/>
  </Route>
</TraceRoute>
</FlowInformationAdvertise>
</MsgBody>
</Core>

```

Figure4.7-6 Ex: Mapping of FlowInformationAdvertise message including SDP

The TraceRoute element under the FlowInformationAdvertise element indicates a route that the FlowInformationAdvertise message passes through. Every node on the FlowInformationAdvertise message route adds its own node ID information to this element. The TraceRoute element includes Node IDs of source node and destination node.

## 6.8. MulticastSetup Method

The Multicast method is used in order to set up the streaming data transmission channel from the Receiver node to the Sender node. A route is set up in the section from the Receiver node (multicast member or community member) to the Sender node.

This method consists of the following messages.

- ◆ Request Type : MulticastSetup Message
- ◆ Response Type : MulticastSetupResponse Message

Message exchange is performed by single-hop unicast. This message is transmitted and received among the following nodes.

- Between the Receiver and the Relay/Receiver
- Between the Relay and the Sender/Relay/Receiver
- Between the Receiver and the Sender/Relay/Receiver

The sequence of the MulticastSetup method is shown in the illustrated model as an example. (The arrow means the direction where streaming data flows.)

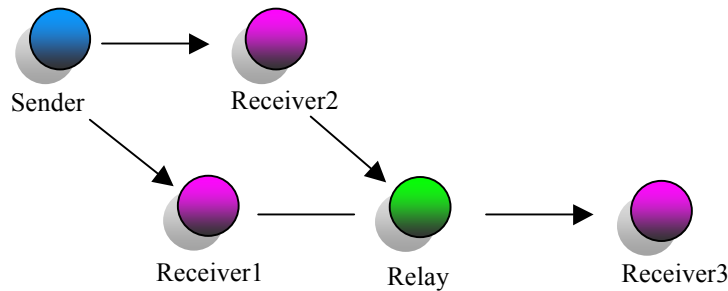


Figure4.8-1 Ex:Network model of MulticastSetup method

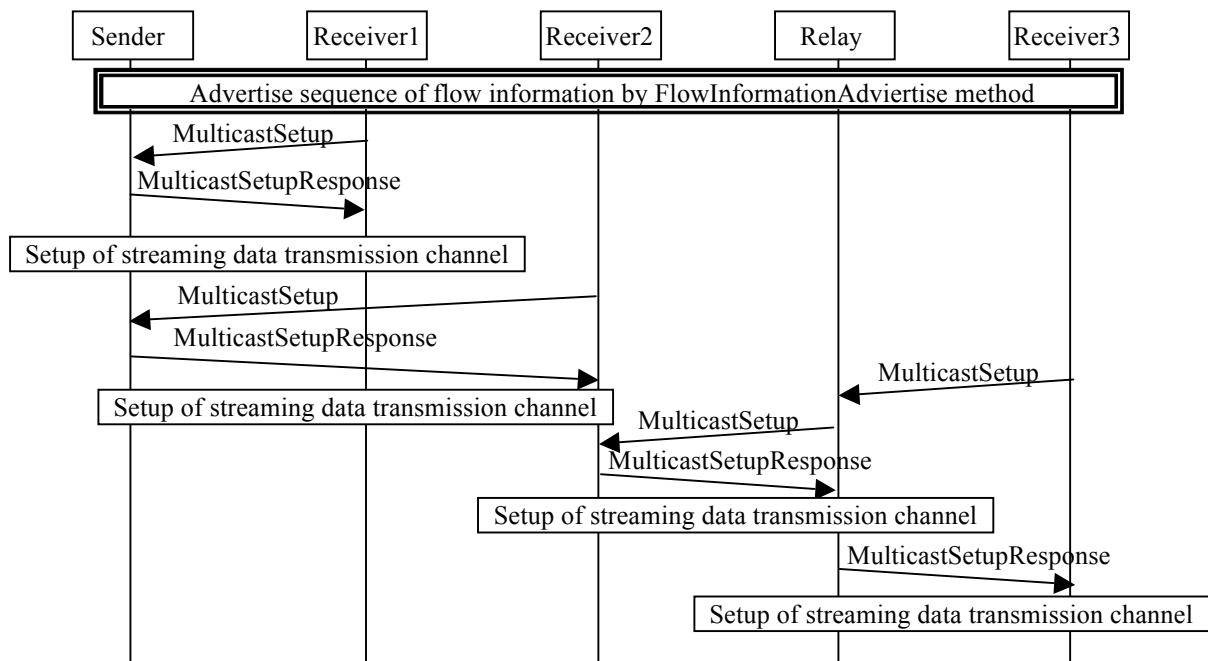


Figure4.8-2 Sequence of MulticastSetup method

The Sender advertises flow information by the FlowInformationAdvertise method. Each Receiver which has received flow information sets up a route in reverse order to the route by which it has received the FlowInformationAdvertise message. Furthermore, the Receiver transmits a MulticastSetup message. A flow ID which identifies the flow, the name of the flow, the start/end time, the MIME type of media data, optional information of the Sender's application, the flow classification, the directivity of the flow, bandwidth information and so on is included in flow information. The Receiver allocates a streaming data transmission channel according to the content of the flow information. The address for transmission (the waiting for streaming data) is set to the MulticastSetup message.

The node which has received the MulticastSetup message allocates a streaming data transmission channel according to the content of the flow information and returns the MulticastSetup message to the Sender. The address for transmission (the waiting port for streaming data) and success (or failure) of the

allocation is set to the MulticastSetupResponse message. Furthermore, this node transmits the MulticastSetup message to the following upstream node according to the route set as the message. A streaming data transmission channel is established by performing exchange of the MulticastSetup / MulticastSetupResponse messages. The node which has received the MulticastSetup message immediately replies with a MulticastSetupResponse message without transmitting the MulticastSetup message to the following node when a transmission channel cannot be allocated.

The node which has received the MulticastSetup message immediately replies with a MulticastSetupResponse message without transmitting the MulticastSetup message to the node when the route is set up by the same flow ID between the following nodes. When another route set up with the the same flow ID already exists, the node returns a MulticastSetupResponse message after the setting end.

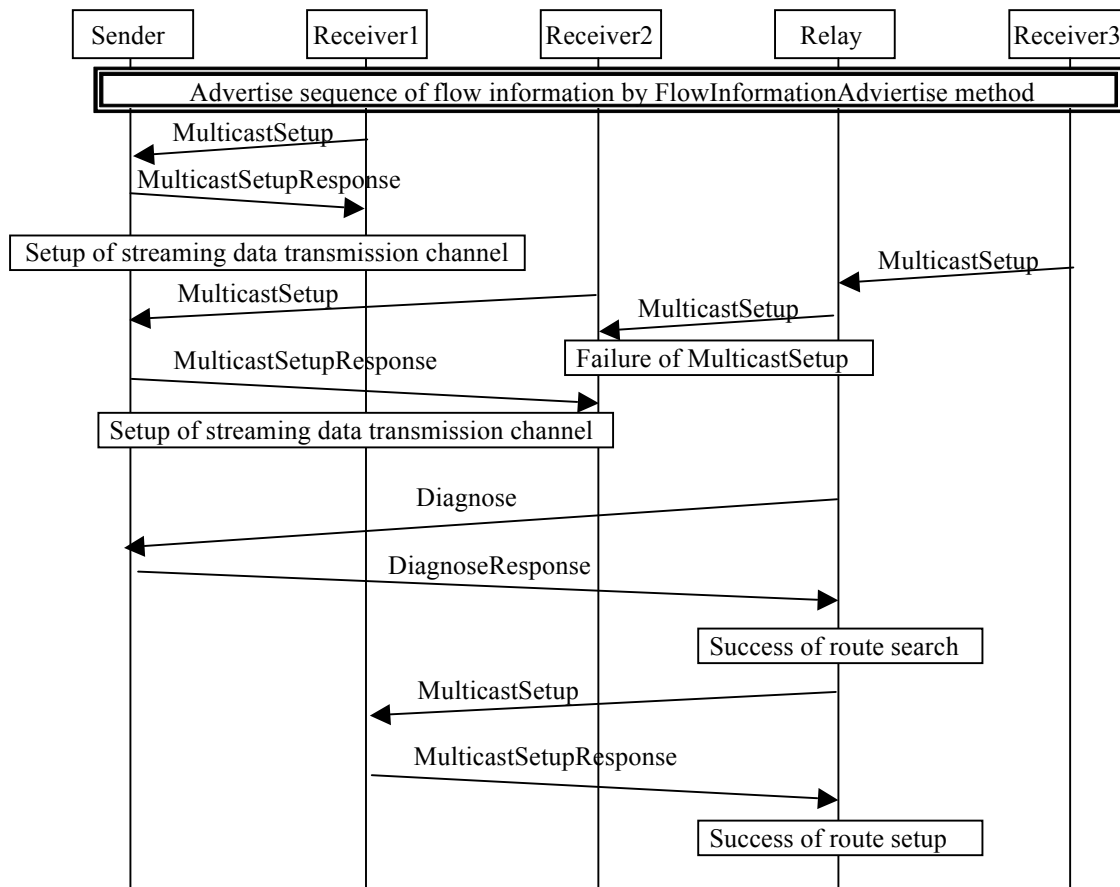


Figure4.8-3 Sequence of MulticastSetup method –Failure of MulticastSetup

***PUCC Streaming Protocol***

When the MulticastSetup goes wrong among arbitrary nodes, the node which has detected the failure tries to set up a flow using the new route to the Receiver. In this case, the MulticastSetup message is set up as follows.

- Flow element : Same flow ID
- Source element : Self-node
- Destination element : New route

The "reset" parameter is added to the Setup message aiming at a re-setup of the route. When another route set up with the same flow ID already exists, the node returns a MulticastSetupResponse message.

When the search or setup of a route goes wrong, the node which has detected the failure (a detection node) performs the following processing.

- A detection node : Relay node

The Relay node transmits an Abort message to the adjacent node in the Receiver's direction.

- A detection node : Sender (the end of MulticastSetup)

The Sender processes nothing.

- A detection node : Receiver (the start of MulticastSetup)

The Receiver stops the flow setup and notifies the user's application of the failure in the flow setup.

The parameter of the MulticastSetup message is defined as follows.

Table4.8-1 MulticastSetup message(1/2)

| Element name   | Element value   | Attribute name | Attribute value   | Occurrence | Element name |
|----------------|---|----------------|---|------------|--------------|
| MulticastSetup | XML fragment<br>(Source element, Destination element, Flow element, RequestedMedium element, TransmissionMedium element)  | -              | -   | 1          | Required     |
|                |   | xmlns          | Name space identifier (URI) of PUCC Streaming Control Protocol            | 1          | Required     |
|                |   | type           | Classification of setup "initial", "reset"                                | 1          | Option       |
| Source         | Node ID of node which starts MulticastSetup   | -              | -   | 1          | Required     |
| Destination    | XML fragment (Route element, Target element)  | -              | -   | 1          | Required     |
| Route          | XML fragment (Route element, Target element)  | -              | -   | 1          | Required     |
|                |   | Node           | Node ID of Route node (Relay/Receiver)                                    | 1          | Required     |
| Target         | Sender node ID  | -              | -   | 1          | Required     |
| Flow           | XML fragment (Name element, Description element, Sender element, Bandwidth element, ActiveTime element, Repeat element, MediaData element, Application element, Url element, Sdp element) | -              | -   | 1          | Required     |
|                |   | ID             | ID of unique identifier (<sequence number> <Data><Time>@"<Sender nodeID>) | 1          | Required     |
|                |   | initiative     | Initiative node of setup start "Sender", "Receiver"                       | 1          | Option       |
|                |   | directionality | Direction of flow "UniDirectional", "BiDirectional"                       | 1          | Option       |
| Name           | Name of flow  | -              | -   | 1          | Option       |
| Description    | Description of flow   | -              | -   | 1          | Option       |
| Sender         | Sender node ID  | -              | -   | 1          | Option       |
| Bandwidth      | Band information (numerical value[bps])   | -              | -   | 1          | Option       |
| ActiveTime     | XML fragment (Start element, End element)   | -              | -   | 1          | Option       |
| Start          | Starting time of flow (ISO8601 form)  | -              | -   | 1          | Required     |
| End            | Ending time of flow (ISO8601 form)  | -              | -   | 1          | Required     |



Table4.8-1 MulticastSetup message(2/2)

| Element name       | Element value  | Attribute name | Attribute value  | Occurrence | Status   |
|--------------------|--|----------------|--|------------|----------|
| Repeat             | XML fragment (Interval element, Duration element)    | -              | -  | 1          | Option   |
| Interval           | Time interval of repetition (Unit[ms])               | -              | -  | 1          | Required |
| Duration           | Active duration (Unit[ms])                           | -              | -  | 1          | Required |
| MediaData          | XML fragment (MimeType element)                      | -              | -  | 1          | Option   |
| MimeType           | MIME type of media data                              | -              | -  | Multiple   | Required |
| Application        | XML fragment (Option element)                        | -              | -  | 1          | Option   |
|                    |  | name           | Name of application which distributes media data   | 1          | Required |
| Option             | Arbitrary characters                                 | -              | -  | 1          | Option   |
| Url                | Url of streaming contents                            | -              | -  | 1          | Option   |
| Sdp                | SDP session description                              | -              | -  | 1          | Option   |
| RequestedMedium    | XML fragment (Address element, ResourceList element) | medium         | Classification of transport medium("RTP/AVP", "MMS", "CIP")                                | 1          | Required |
|                    |  | protocol       | Classification of transport protocol("UDP", "IEEE1394", "Bluetooth", "HTTP")               | 1          | Required |
|                    |  | type           | Classification of address("IPv4", "IPv6", "IPv4Multicast", "IEEE1394", "Bluetooth", "TCP") | 1          | Required |
| TransmissionMedium | XML fragment (Address element, ResourceList element) | medium         | Classification of transport medium ("RTP/AVP", "MMS", "CIP")                               | 1          | Required |
|                    |  | protocol       | Classification of transport protocol ("UDP", "IEEE1394", "Bluetooth", "HTTP",)             | 1          | Required |
|                    |  | type           | Classification of address ("IPv4", "IPv6", "IEEE1394", "Bluetooth", "TCP")                 | 1          | Required |
| Address            | Streaming data transmission address                  | -              | -  | 1          | Required |
| ResourceList       | XML fragment (Resource element)                      | -              | -  | 1          | Required |
| Resource           | Resources  | -              | -  | Multiple   | Required |
|                    |  | type           | Type of resource ("port", "channel")   | 1          | Required |
|                    |  | mimetype       | MIME type of media data  | 1          | Required |

**PUC Streaming Protocol**

The MulticastSetup message is mapped by the PUC core protocol as follows.

Table4.8-2 Mapping of MulticastSetup message for PUC Core Protocol

| Element name | Element value                         | Attribute name | Attribute value                                  | Occurrence | Status   |
|--------------|---------------------------------------|----------------|--|------------|----------|
| Core         | XML fragment                          | -              | -  | 1          | Required |
|              |                                       | xmlns          | Name space identifier (URI) of PUC Core Protocol | 1          | Required |
| ComType      | "Unicast"                             | -              | -  | 1          | Required |
| MsgID        | Message ID                            | -              | -  | 1          | Required |
| MsgType      | "Request"                             | -              | -  | 1          | Required |
| CommunityID  | Community ID                          | -              | -  | 1          | Option   |
| Source       | Source node ID                        | -              | -  | 1          | Required |
| Destination  | XML fragment (Target element)         | -              | -  | 1          | Required |
| Target       | Destination node ID                   | -              | -  | 1          | Required |
| TraceRoute   | XML fragment (Route element)          | -              | -  | 1          | Option   |
| Route        | XML fragment (Route element)          | -              | -  | 1          | Required |
|              |                                       | Node           | Relay node ID                                    | 1          | Required |
| SessionID    | Session ID                            | -              | -  | 1          | Option   |
| MsgBody      | XML fragment (MulticastSetup element) | -              | -  | 1          | Required |
|              |                                       | protocol       | Protocol ID of PUC Streaming Control Protocol    | 1          | Required |

The following is an example of MulticastSetup message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Request</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029" />
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <MulticastSetup xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl" type="initial">
      <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
      <Destination>
        <Route Node="975870cd-f9bb-4305-1234-e65e29f6030">
          <Target>975870cd-f9bb-4305-1234-e65e29f6031</Target>
        </Route>
      </Destination>
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="UniDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
      </Flow>
    </MulticastSetup>
  </MsgBody>
</Core>

```

**PUCC Streaming Protocol**

```

</ActiveTime>
<Repeat>
  <Interval>90000</Interval>
  <Duration>900000</Duration>
</Repeat>
<MediaData>
  <MimeType>video/x-ms-wmv</MimeType>
  <MimeType>audio/x-ms-wma</MimeType>
</MediaData>
  < Application name="WindowsMediaService9"/>
</Flow>
<TransmissionMedium medium="RTP/AVP" protocol="UDP" type="IPv4">
  <Address>192.168.0.1</Address>
  <ResourceList>
    <Resource type="port" mimetype="video/x-ms/wmv">9900</Resource>
    <Resource type="port" mimetype="audio/x-ms/wma">9902</Resource>
  </ResourceList>
</TransmissionMedium>
</MulticastSetup>
</MsgBody>
</Core>

```

Figure4.8-4 Ex:Mapping of MulticastSetup message

The parameter of the MulticastSetupResponse message is defined as follows.

Table4.8-3 MulticastSetupResponse message(1/2)

| Element name           | Element value   | Attribute name | Attribute value  | Occurrence | Status   |
|------------------------|---|----------------|--|------------|----------|
| MulticastSetupResponse | XML fragment (Result element, Reason element, Flow element)   | -              | -  | 1          | Required |
|                        |   | xmlns          | Name space identifier (URI) of PUCC Streaming Core Protocol                            | 1          | Required |
|                        |   | type           | Classification of setup "initial"(initialization), "reset"(resetting)                  | 1          | Option   |
| Result                 | "Success", "Failure"  | -              | -  | 1          | Required |
| Reason                 | "NotEnoughBandwidth", "UnsupportedTransport", "PortUnavailable", "InternalError"  | -              | -  | 1          | Option   |
| Flow                   | XML fragment (Name element, Description element, Sender element, Bandwidth element, ActiveTime element, Repeat element, MediaData element, Application element, Url element, Sdp element) | ID             | ID for discriminating flow uniquely (<sequence number> <Data><Time>"@"<Sender nodeID>) | 1          | Required |
|                        |   | initiative     | Initiative node of setup start "Sender", "Receiver"                                    | 1          | Option   |
|                        |   | directionality | Direction of flow "UniDirectional", "BiDirectional"                                    | 1          | Option   |
| Name                   | Name of flow  | -              | -  | 1          | Option   |
| Description            | Description of flow   | -              | -  | 1          | Option   |
| Sender                 | Sender node ID  | -              | -  | 1          | Option   |

Table4.8-3 MulticastSetupResponse message(2/2)

| Element name       | Element value  | Attribute name | Attribute value  | Occurrence | Status   |
|--------------------|--|----------------|--|------------|----------|
| Bandwidth          | Band information (numerical value[bps])              | -              | -  | 1          | Option   |
| ActiveTime         | XML fragment (Start element, End element)            | -              | -  | 1          | Option   |
| Start              | Starting time of flow (ISO8601 form)                 | -              | -  | 1          | Required |
| End                | Ending time of flow (ISO8601 form)                   | -              | -  | 1          | Required |
| Repeat             | XML fragment (Interval element, Duration element)    | -              | -  | 1          | Option   |
| Interval           | Time interval of repetition(Unit[ms])                | -              | -  | 1          | Required |
| Duration           | Active duration(Unit[ms])                            | -              | -  | 1          | Required |
| MediaData          | XML fragment (MimeType element)                      | -              | -  | 1          | Option   |
| MimeType           | MIME type of media data                              | -              | -  | Multiple   | Required |
| Application        | XML fragment (Option element)                        | -              | -  | 1          | Option   |
|                    |  | name           | Name of application which distributes media data   | 1          | Required |
| Option             | Arbitrary characters                                 | -              | -  | 1          | Option   |
| Url                | Url of streaming contents                            | -              | -  | 1          | Option   |
| Sdp                | SDP session description                              | -              | -  | 1          | Option   |
|                    |  | medium         | Classification of transport medium("RTP/A VP", "MMS", "CIP")                               | 1          | Required |
|                    |  | protocol       | Classification of transport protocol("UDP", "IEEE1394", "Bluetooth", "HTTP")               | 1          | Required |
| TransmissionMedium | XML fragment (Address element, ResourceList element) | type           | Classification of address("IPv4", "IPv6", "IPv4Multicast", "IEEE1394", "Bluetooth", "TCP") | 1          | Required |

Table4.8-4 Set value of Reason element of MulticastSetupResponse message

|   | Factor setting value | Summary   |
|---|----------------------|---|
| 1 | NotEnoughBandwidth   | Bandwidth required to transmit streaming data cannot be reserved. |
| 2 | UnsupportedTransport | The transport protocol is not supported.                          |
| 3 | PortUnavailable      | Port cannot be used   |
| 4 | InternalError        | Internal error happened.  |

**PUC Streaming Protocol**

The MulticastSetupResponse message is mapped by the PUC core protocol as follows.

Table4.8-5 Mapping of MulticastSetupResponse message for PUC Core Protocol

| Element name | Element value                                 | Attribute name | Attribute value                                  | Occurrence | Status   |
|--------------|---|----------------|--|------------|----------|
| Core         | XML fragment                                  | -              | -  | 1          | Required |
|              |   | xmlns          | Name space identifier (URI) of PUC Core Protocol | 1          | Required |
| ComType      | “Unicast”                                     | -              | -  | 1          | Required |
| MsgID        | Message ID                                    | -              | -  | 1          | Required |
| ReplyID      | Element value of MsgID of Setup message       | -              | -  | 1          | Required |
| MsgType      | “Response”                                    | -              | -  | 1          | Required |
| CommunityID  | Community ID                                  | -              | -  | 1          | Option   |
| Source       | Source node ID                                | -              | -  | 1          | Required |
| Destination  | XML fragment (Target element)                 | -              | -  | 1          | Required |
| Target       | Destination node ID                           | -              | -  | 1          | Required |
| TraceRoute   | XML fragment (Route element)                  | -              | -  | 1          | Option   |
| Route        | XML fragment (Route element)                  | -              | -  | 1          | Required |
|              |   | Node           | Relay node ID                                    | 1          | Required |
| SessionID    | Session ID                                    | -              | -  | 1          | Option   |
| MsgBody      | XML fragment (MulticastSetupResponse element) | -              | -  | 1          | Required |
|              |   | protocol       | Protocol ID of PUC Streaming Control Protocol    | 1          | Required |

The following is an example of the MulticastSetupResponse message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:14:10Z@975870cd-f9bb-4305-1234-e65e29f6031</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>
  <MsgType>Response</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6031</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6031"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <MulticastSetupResponse xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl" type="initial">
      <Result>Success</Result>
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="UniDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
        <Repeat>
          <Interval>90000</Interval>
          <Duration>900000</Duration>
        </Repeat>
        <MediaData>
          <MimeType>video/x-ms-wmv</MimeType>
        </MediaData>
      </Flow>
    </MulticastSetupResponse>
  </MsgBody>
</Core>

```

**PUC Streaming Protocol**

```

    <MimeType>audio/x-ms-wma</MimeType>
  </MediaData>
  < Application name="WindowsMediaService9"/>
</Flow>
</MulticastSetupResponse>
</MsgBody>
</Core>

```

Figure4.8-5 Ex1:Mapping of MulticastSetupResponse message

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:13:26Z@975870cd-f9bb-4305-1234-e65e29f6030</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>
  <MsgType>Response</MsgType>
  <CommunityID>urn:ED:Community:PUC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6030</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6029</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6030"/>
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <MulticastSetupResponse xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl" type="reset">
      <Result>Failure</Result>
      <Reason>NotEnoughBandwidth</Reason>
      <Flow ID="00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029"
        initiative="Sender" directionality="UniDirectional">
        <Name>Video Delivery</Name>
        <Description>Video Delivery</Description>
        <Sender>975870cd-f9bb-4305-1234-e65e29f6029</Sender>
        <Bandwidth>384000</Bandwidth>
        <ActiveTime>
          <Start>2004-01-20T21:00:00-09:00</Start>
          <End>2004-01-20T22:00:00-09:00</End>
        </ActiveTime>
        <Repeat>
          <Interval>90000</Interval>
          <Duration>900000</Duration>
        </Repeat>
        <MediaData>
          <MimeType>video/x-ms-wmv</MimeType>
          <MimeType>audio/x-ms-wma</MimeType>
        </MediaData>
        < Application name="WindowsMediaService9"/>
      </Flow>
    </MulticastSetupResponse>
  </MsgBody>
</Core>

```

Figure4.8-6 Ex2:Mapping of MulticastSetupResponse message

## 6.9. MulticastKeepalive Method

The MulticastKeepalive method is used in order to maintain the flow under streaming data transmission in PUCC multicast (broadcast) streaming. Each node on the route of the flow exchanges messages with the adjacent nodes which have transmitted the MulticastSetup message

This method consists of the following messages.

- ◆ Request Type : MulticastKeepalive Message
- ◆ Response Type : MulticastKeepaliveResponse Message

Message exchange is performed by single-hop unicast. This message is transmitted and received among the following nodes.

- Between Receiver and Relay/Receiver
- Between Relay and Sender/Relay/Receiver
- Between Receiver and Sender/Relay/Receiver

The sequence of the MulticastKeepalive method is shown for in illustrated model as an example. (The arrow means the direction where streaming data flows.)

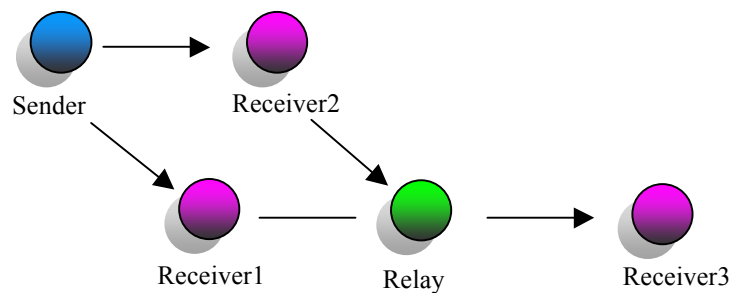


Figure4.9-1 Ex:Network model of MulticastKeepalive method

**PUCC Streaming Protocol**

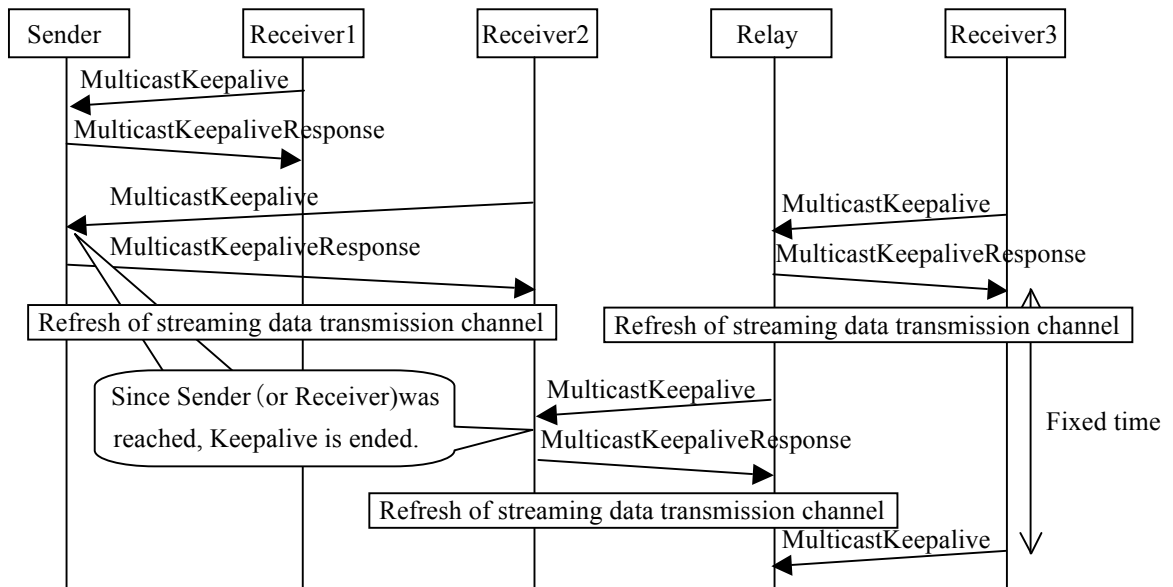


Figure4.9-2 Sequence of MulticastKeepalive method –Success of MulticastKeepalive

Each node which has set up the flow transmits the MulticastKeepalive message to an upstream adjacent node periodically. The Flow ID is set to the MulticastKeepalive message. The node which has received the MulticastKeepalive message immediately returns the MulticastKeepaliveResponse message. The node which has received the MulticastKeepaliveResponse message resets the soft state timer of the streaming data transmission channel among downstream adjacent nodes. All the nodes on a route maintain the flow by refreshing the transmission channel among adjacent nodes periodically.



**PUCC Streaming Protocol**

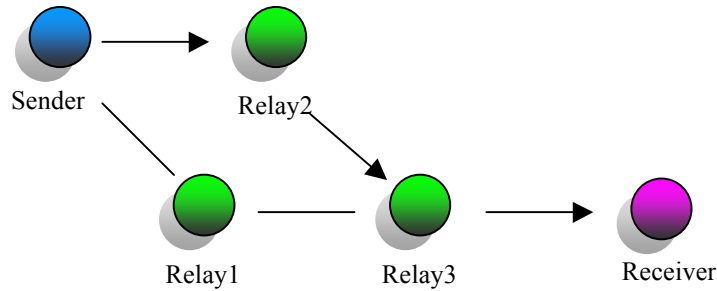


Figure4.9-3 Ex:Network model of MulticastKeepalive method –Failure of MulticastKeepalive

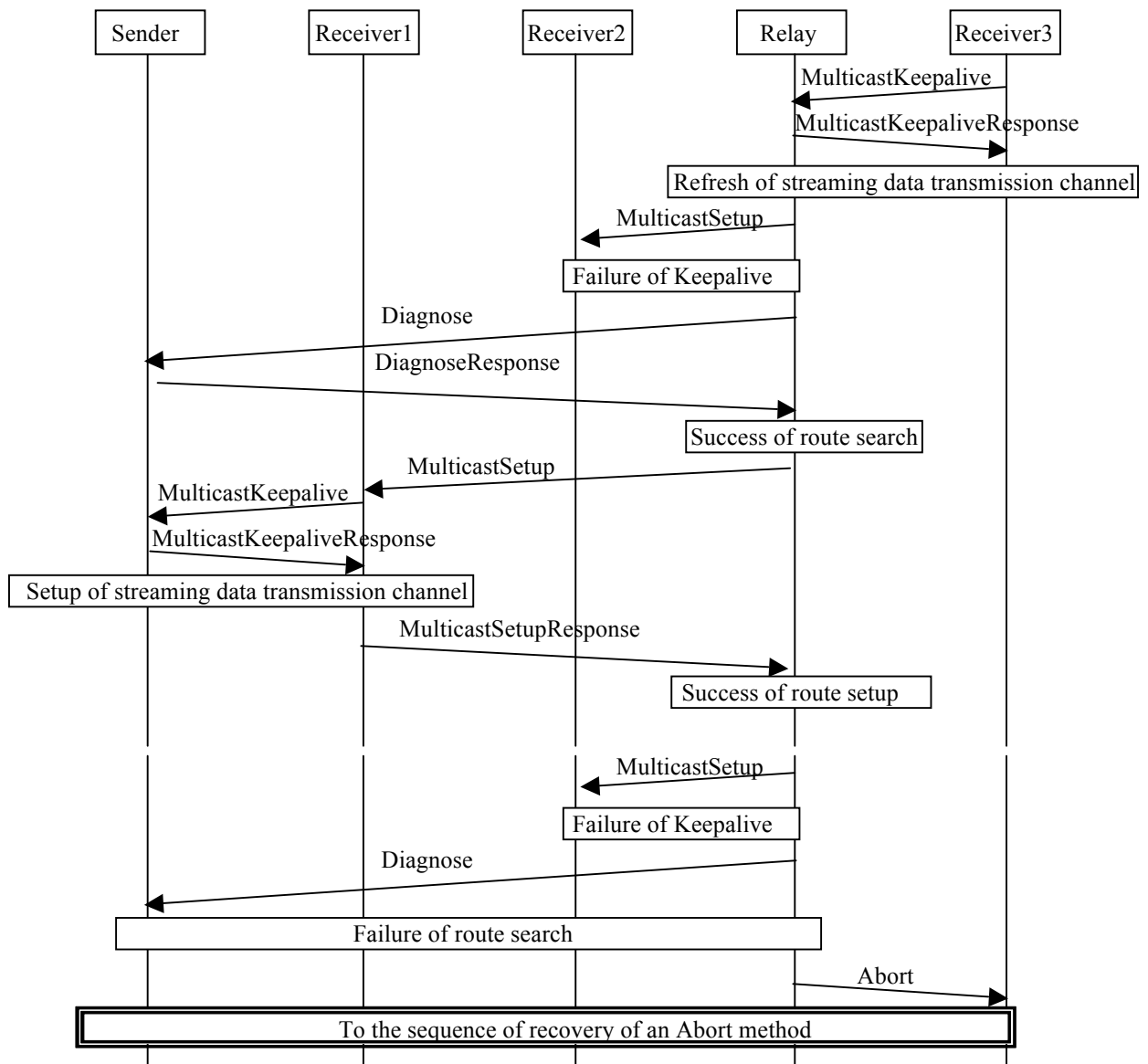


Figure4.9-4 Sequence of MulticastKeepalive method –Failure pf MulticastKeepalive

***PUCC Streaming Protocol***

When the MulticastKeepalive goes wrong among the arbitrary nodes on the route, the node which has detected the failure tries to set up a flow, using a new route to the Sender. In this case, the MulticastSetup message is set up as follows.

- Flow element : Same flow ID
- Source element : Self-node
- Destination element : New route

When another route set up with the same flow ID already exists, the node returns a MulticastSetupResponse message. When the search or setup of a route goes wrong, the node which has detected the failure performs the following processing.

- The node which has detected the failure : Relay node

The Relay node transmits an Abort message to the adjacent node in the Receiver's direction.

- The node which has detected the failure : Sender (the end of MulticastSetup)

The Receiver processes nothing.

- The node which has detected the failure : Receiver (the start of Receiver)

The Sender stops a flow setup and notifies the user's application of the failure.

The parameter of the MulticastKeepalive message is defined as follows.

Table4.9-1 MulticastKeepalive message

| Element name       | Element value                       | Attribute name | Attribute value  | Occurrence | Status   |
|--------------------|-------------------------------------|----------------|--|------------|----------|
| MulticastKeepalive | XML fragment (FlowID element)       | -              | -  | 1          | Required |
|                    |                                     | xmlns          | Name space identifier (URI) of PUCS Streaming Control Protocol | 1          | Required |
| FlowID             | ID for discriminating flow uniquely | -              | -  | 1          | Required |

**PUCC Streaming Protocol**

The MulticastKeepalive message is mapped by the PUCC core protocol as follows.

Table4.9-2 Mapping of MulticastKeepalive message for PUCC Core Protocol

| Element name | Element value                             | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|---|----------------|---|------------|----------|
| Core         | XML fragment                              | -              | -   | 1          | Required |
|              |   | xmlns          | Name space identifier (URI) of PUCC Core Protocol | 1          | Required |
| ComType      | “Unicast”                                 | -              | -   | 1          | Required |
| MsgID        | Message ID                                | -              | -   | 1          | Required |
| MsgType      | “Request”                                 | -              | -   | 1          | Required |
| CommunityID  | Community ID                              | -              | -   | 1          | Option   |
| Source       | Source node ID                            | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)             | -              | -   | 1          | Required |
| Target       | Destination node ID                       | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)              | -              | -   | 1          | Option   |
|              |   | -              | -   | 1          | Required |
| Route        | XML fragment (Route element)              | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID                                | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (MulticastKeepalive element) | -              | -   | 1          | Required |
|              |   | protocol       | Protocol ID of PUCC Streaming Control Protocol    | 1          | Required |

The following is an example of the MulticastKeepalive message.

```

<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</MsgID>
  <MsgType>Request</MsgType>
  <CommunityID>urn:ED:Community:PUCC</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6029</Source>
  <Destination>
    <Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
  </Destination>
  <TraceRoute>
    <Route Node="975870cd-f9bb-4305-1234-e65e29f6029" />
  </TraceRoute>
  <MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <MulticastKeepalive xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
      <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
    </MulticastKeepalive>
  </MsgBody>
</Core>

```

Figure4.9-5 Ex:Mapping of MulticastKeepalive message

The parameter of the MulticastKeepaliveResponse message is defined as follows.

Table4.9-3 MulticastKeepaliveResponse message

| Element name               | Element value                       | Attribute name | Attribute value  | Occurrence | Status   |
|----------------------------|-------------------------------------|----------------|--|------------|----------|
| MulticastKeepaliveResponse | XML fragment (FlowID element)       | -              | -  | 1          | Required |
|                            |                                     | xmlns          | Name space identifier (URI) of Pucc Streaming Control Protocol | 1          | Required |
| FlowID                     | ID for discriminating flow uniquely | -              | -  | 1          | Required |

The MulticastKeepaliveResponse message is mapped by the Pucc core protocol as follows.

Table4.9-5 Mapping of MulticastKeepaliveResponse message for Pucc Core Protocol

| Element name | Element value                                     | Attribute name | Attribute value                                   | Occurrence | Status   |
|--------------|---|----------------|---|------------|----------|
| Core         | XML fragment                                      | -              | -   | 1          | Required |
|              |   | xmlns          | Name space identifier (URI) of Pucc Core Protocol | 1          | Required |
| ComType      | “Unicast”   | -              | -   | 1          | Required |
| MsgID        | Message ID  | -              | -   | 1          | Required |
| ReplyID      | Element value of MsgID of Setup message           | -              | -   | 1          | Required |
| MsgType      | “Response”  | -              | -   | 1          | Required |
| CommunityID  | Community ID                                      | -              | -   | 1          | Option   |
| Source       | Source node ID                                    | -              | -   | 1          | Required |
| Destination  | XML fragment (Target element)                     | -              | -   | 1          | Required |
| Target       | Destination node ID                               | -              | -   | 1          | Required |
| TraceRoute   | XML fragment (Route element)                      | -              | -   | 1          | Option   |
| Route        | XML fragment (Route element)                      | -              | -   | 1          | Required |
|              |   | Node           | Relay node ID                                     | 1          | Required |
| SessionID    | Session ID  | -              | -   | 1          | Option   |
| MsgBody      | XML fragment (MulticastKeepaliveResponse element) | -              | -   | 1          | Required |
|              |   | protocol       | Protocol ID of Pucc Streaming Control Protocol    | 1          | Required |

The following is an example of the MulticastKeepaliveResponse message.

```
<Core xmlns="http://www.pucc.jp/2012/03//p2p_core">
  <ComType>Unicast</ComType>
  <MsgID>12346.2003-12-14T12:14:10Z@975870cd-f9bb-4305-1234-e65e29f6031</MsgID>
  <ReplyID>12345.2003-12-14T12:13:24Z@975870cd-f9bb-4305-1234-e65e29f6029</ReplyID>
  <MsgType>Response</MsgType>
  <CommunityID>urn:ED:Community:Pucc</CommunityID>
  <Source>975870cd-f9bb-4305-1234-e65e29f6031</Source>
  <Destination>
```

***PUCC Streaming Protocol***

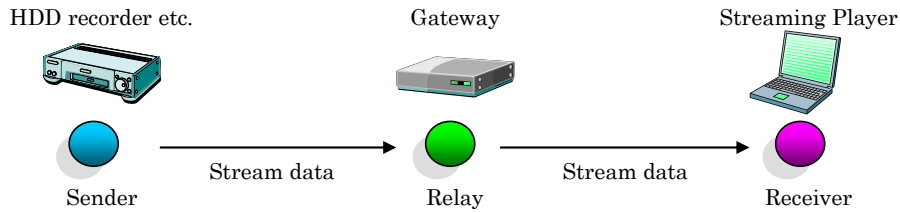
```
<Target>975870cd-f9bb-4305-1234-e65e29f6030</Target>
</Destination>
<TraceRoute>
  <Route Node="975870cd-f9bb-4305-1234-e65e29f6031"/>
</TraceRoute>
<MsgBody protocol="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
  <MulticastKeepaliveResponse xmlns="http://www.pucc.jp/2012/03//p2p_stream_ctrl">
    <FlowID>00001.2003-12-14T12:13:24@975870cd-f9bb-4305-1234-e65e29f6029</FlowID>
  </MulticastKeepaliveResponse>
</MsgBody>
</Core>
```

Figure4.9-6 Ex: Mapping of MulticastKeepaliveResponse message

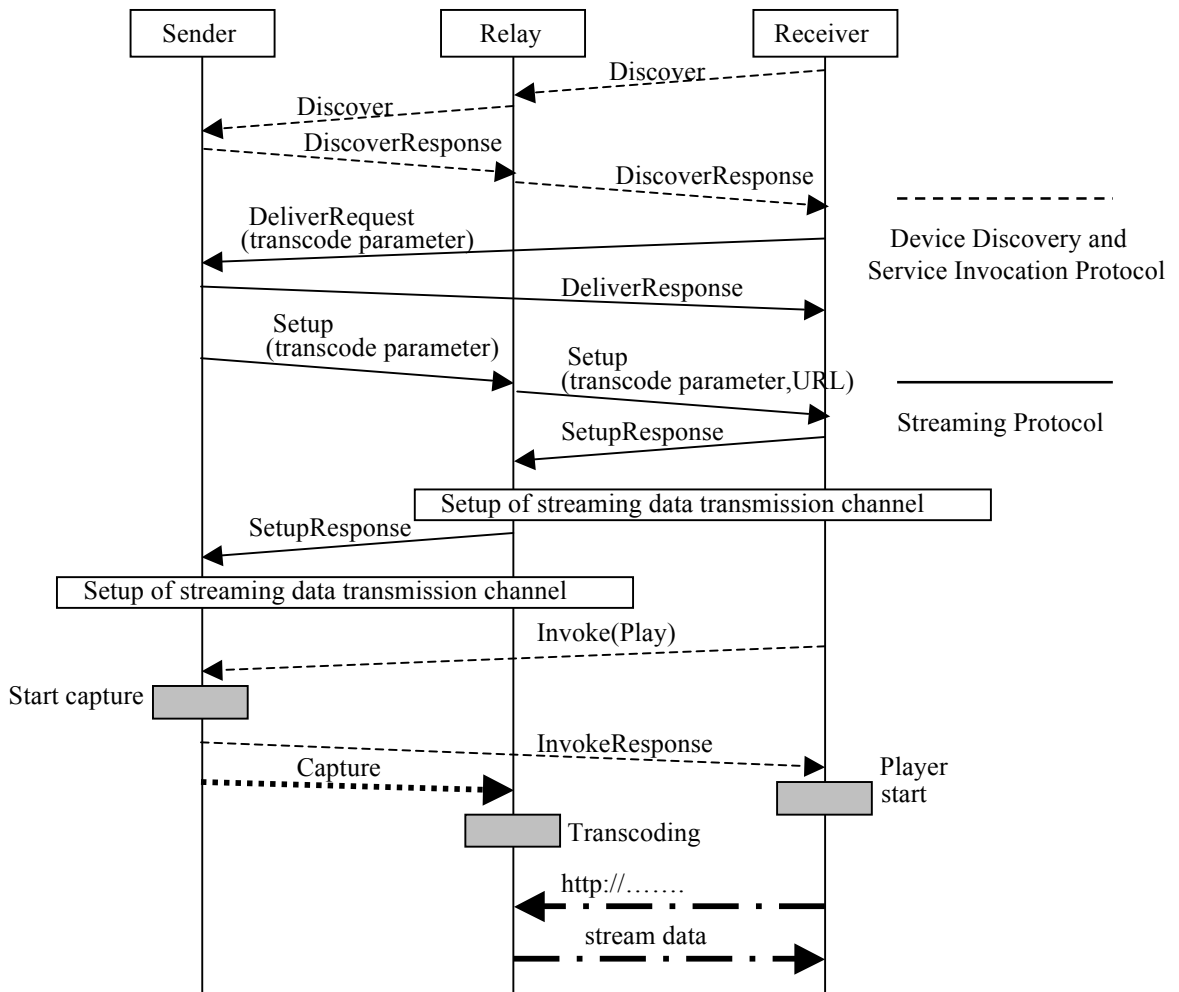
## Appendix A. Version History

| Document number         | Date           | Note        |
|-------------------------|----------------|-------------|
| PUCC Streaming Protocol | 30 Sep,2007    | Version 1.0 |
| PUCC Streaming Protocol | 30 ,Nov 2009   | Version 2.0 |
| PUCC Streaming Protocol | 22 ,March 2012 | Version 3.0 |
|                         |                |             |
|                         |                |             |
|                         |                |             |

**Appendix B. Example of Device Discovery and PUCC Streaming Sequence**

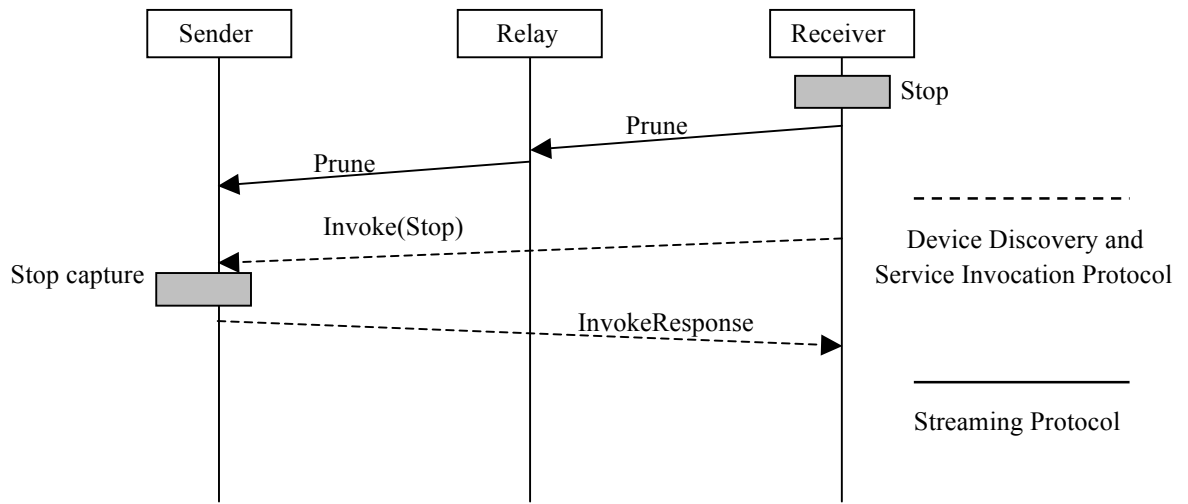


FigureB-1 Network model



FigureB-2 Sequence of Control and Streaming (start)

**PUCC Streaming Protocol**



FigureB-3 Sequence of Control and Streaming (stop)



|   |  |             |
|---|--|-------------|
|  |  | Page73 (76) |
| <b><i>PUCC Streaming Protocol</i></b>   |  |             |

## **Appendix C: Namespace Definitions**

Namespaces are defined with URI.

*[http://www.pucc.jp/2012/03/stream\\_ctrl](http://www.pucc.jp/2012/03/stream_ctrl)*

xmlns attribute of MsgBody element uses it in PUCC Streaming Control Protocol.

|   |  |             |
|---|--|-------------|
|  |  | Page74 (76) |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |
| <b>P2P Universal Computing Consortium</b>   |  |             |

## Appendix D: How to Set an SDP Session Description in Flow Information

Signaling protocols such as SIP and RTSP describe in an SDP (Session Description Protocol) session description format the information equal to what P2P Universal Computing Consortium Streaming Protocol specifies as flow information. Setting an SDP session description in flow information allows applications working with SIP, RTSP, etc. to simplify their session description processing by sharing it across them.

[Note]

1. When the Sender and the Receiver form a multi-hop connection, any hopped node shall modify its setting values appropriately. Unlike P2P Universal Computing Consortium, SDP(Session Description Protocol) is only applicable to single hops.

[How to set an SDP session description]

1. Set an SDP session description in the Sdp element in the Flow element.
2. Set the SDP session description in a text format specified by [SDP].
3. Use UTF-8 for character encoding.
4. In accordance with [XML], escape in a character reference format any letters that cannot be included in an XML format such as '<' and '>'.

[Priority in Setting]

1. The items set in [SDP] may also be set in the RequestedMedium element or the TransportMedium element of the Setup message, the SetupResponse message, or the MulticastSetupResponse message. When both values do not match, the value of either the RequestedMedium element or the TransportMedium element has priority.

## **Appendix E: Transport binding of the content transfer protocol messages over the Bluetooth(SPP)**

This section describes the transport binding of the content transfer protocol messages over Bluetooth(SPP). The specification of the transport binding of the PUCC protocol messages over Bluetooth(SPP) is defined in the PUCC Basic Protocol specification. See the chapter about Bluetooth Transport(SPP) in the PUCC Basic Protocol specification.

Content transfer protocol messages are encapsulated by the following frame when the Content Transfer Protocol is used over Bluetooth(SPP). The encapsulation format of the Content Transfer Protocol is informative.

**TableE-1: The format of the frame for binding of Content Transfer Protocols over Bluetooth(SPP)**

|         | Description  | Status   |
|---------|--|----------|
| Header  | There are 2 kinds of header.<br>A) Frame of normal message<br><b>CFRM</b> <SP> <b>Connectiotype</b> <SP> <b>FrameNo</b> (- <b>SeqNo/WholePacketCounts</b> )<SP> <b>Size</b> <SP> <b>DestNodeID</b> <SP> <b>SrcNodeID</b> ( <SP> <b>CommunityID</b> ) <CR><LF><br>B) Frame in response to a frame error<br><b>FRMERR</b> <CR><LF>   | Required |
|         | <b>Connectiotype</b><br>The Connectiotype parameter is used to designate a connection type.<br><b>0: keep-alive</b> (Transport connection not disconnected after this frame. When content transfer protocols are over OBEX, this value must be set.)<br><b>1: single</b> (Disconnect transport connection after this frame.)   | Required |
|         | <b>FrameNo</b><br>The FrameNo parameter is sequential number used to distinguish frames. This parameter is set at the default value at 0 and final value is 2147483647. If a message is divided, "-SeqNo/WholePacketCounts" is given to the end of FrameNo. The frame is specified by changing only the part of "- SeqNo/WholePacketCounts". The position of a frame is specified by "-SeqNo/WholePacketCounts". | Required |
|         | <b>Size</b><br>The Size parameter is used to designate size of payload (without header and trailer) by the octet number.   | Required |
|         | <b>DestNodeID</b><br>The DestNodeID parameter is used to designate destination node ID. If destination node ID is not fixed, "*" is specified.   | Required |
|         | <b>SrcNodeID</b><br>The SrcNodeID parameter is used to designate source node ID.   | Required |
|         | <b>CommunityID</b><br>The CommunityID parameter is used to distinguish which community the message belongs to.   | Optional |
| Payload | A) Frame of normal message<br><b>(MIME entity header)</b> <CR><LF> [ <b>Content</b> ]<br>The MIME entity header is optional. The Payload has one Content.<br>B) Frame to response to a frame error<br><b>[Header of CFRM]</b><br>The Payload has one header of CFRM which is regarded as error.  | Required |
| Trailer | <b>FRMEND</b> <CR><LF>   | Required |


|   |  |             |
|---|--|-------------|
| <br>P2P Universal Computing Consortium |  | Page76 (76) |
| <b><i>PUCC Streaming Protocol</i></b>   |  |             |

Figure E-1 The following shows a sample CFRM frame.

```
CFRM 0 0-1/3 702 965841a8-aa56-8746-b456c458450 113541a8-bb56-8746-b456c457581 urn:pucc:community:X
// content
FRMEND
```

**Figure E-1 A sample of CFRM frame**