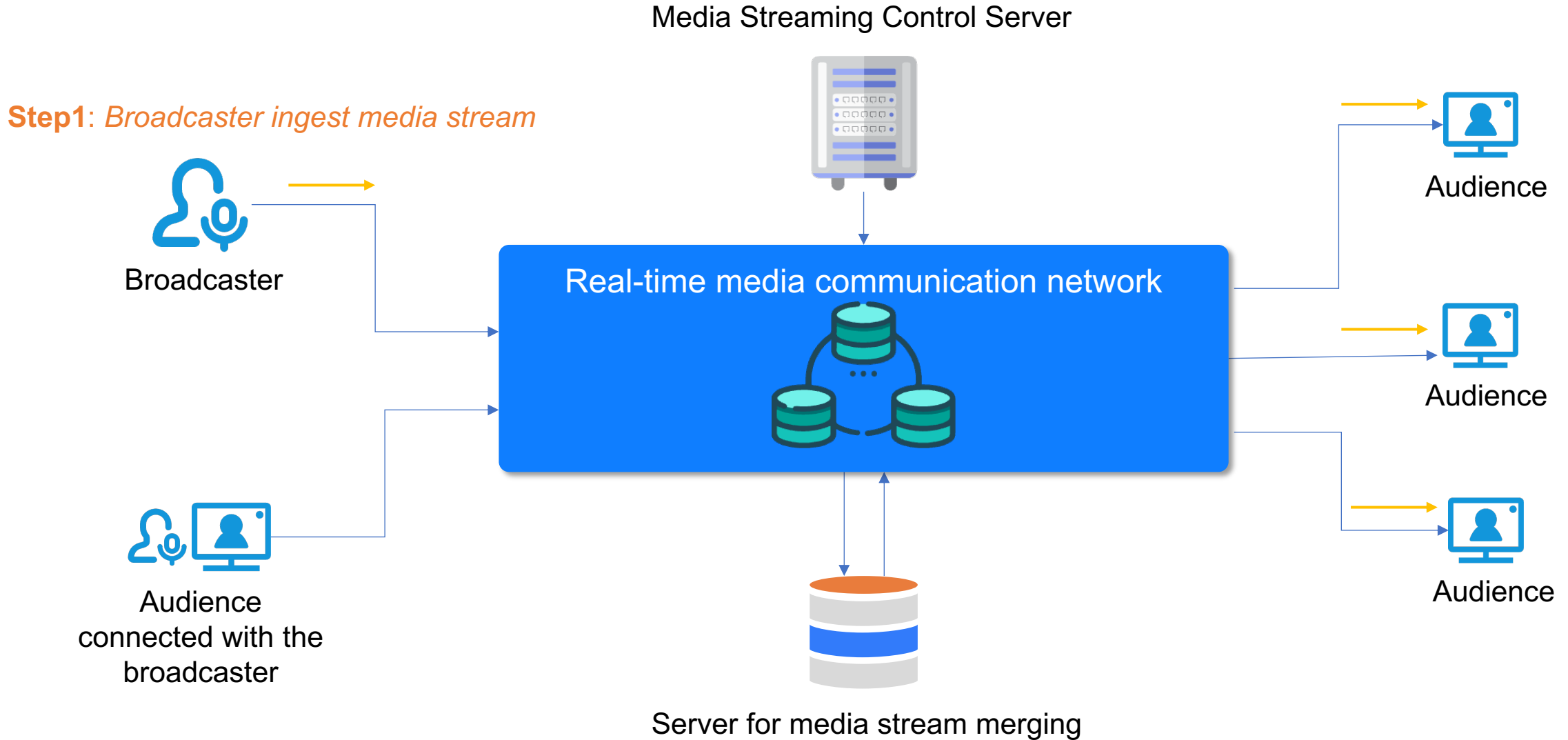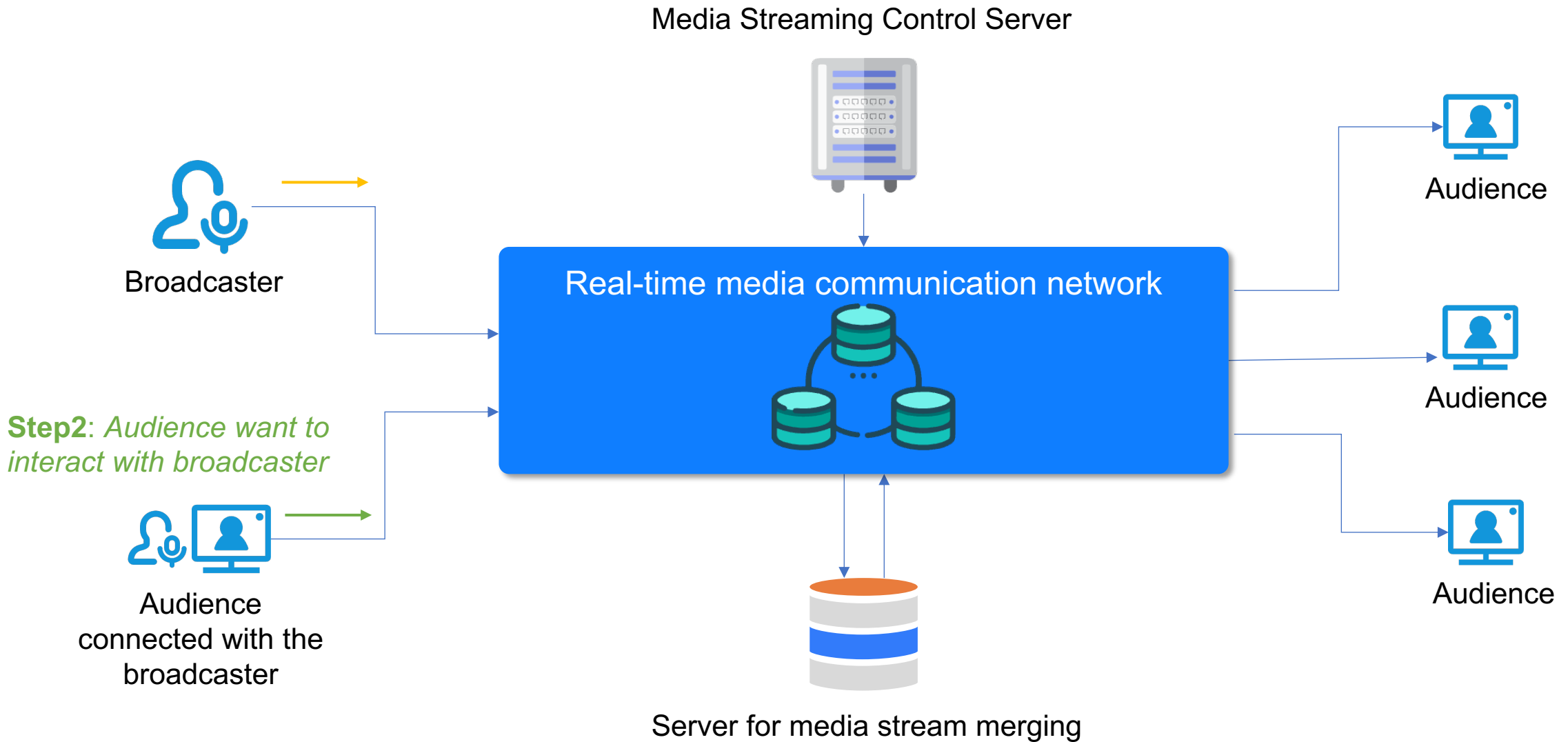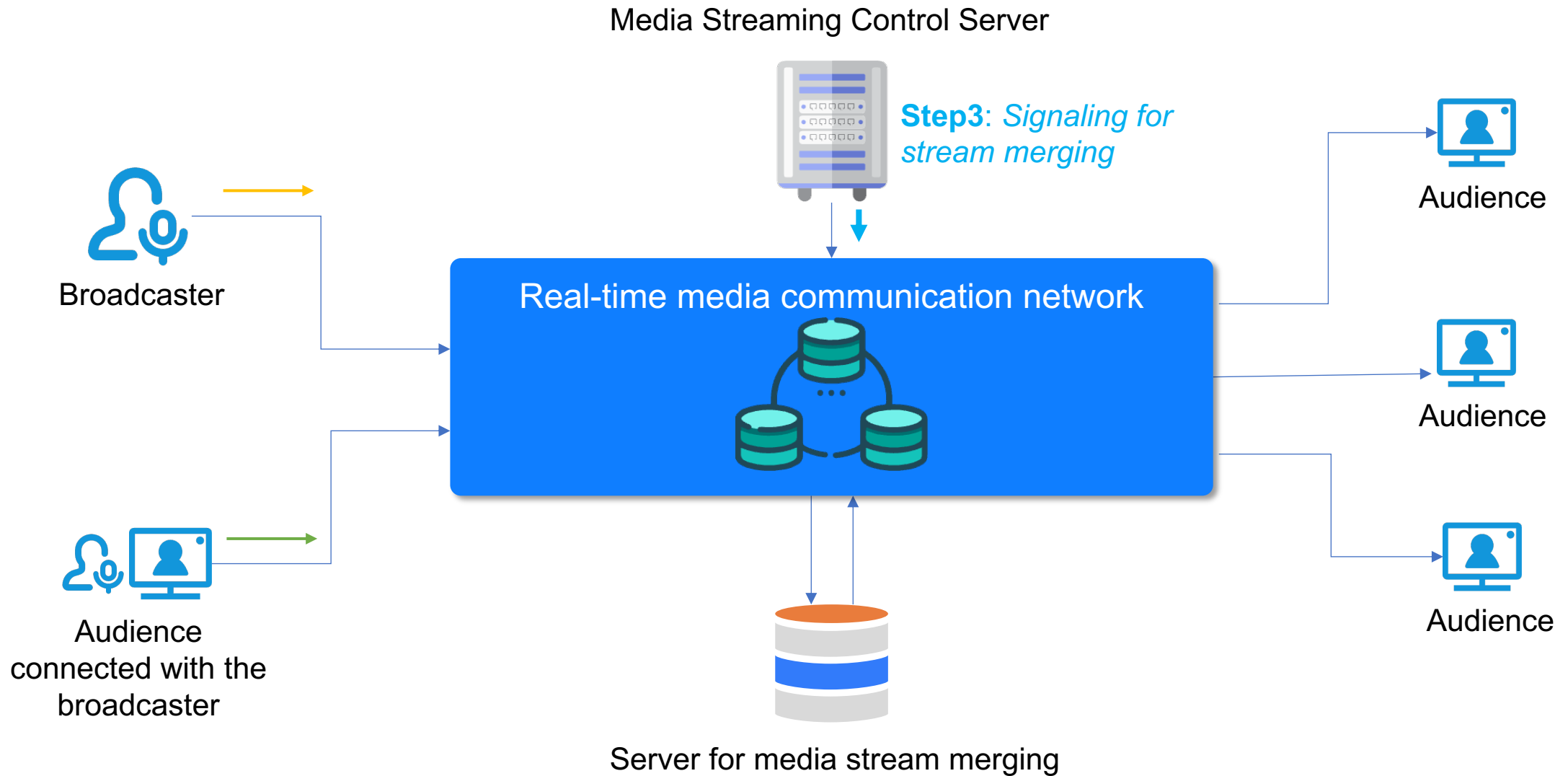# *WebRTC-HTTP Interactive Signaling Protocol*

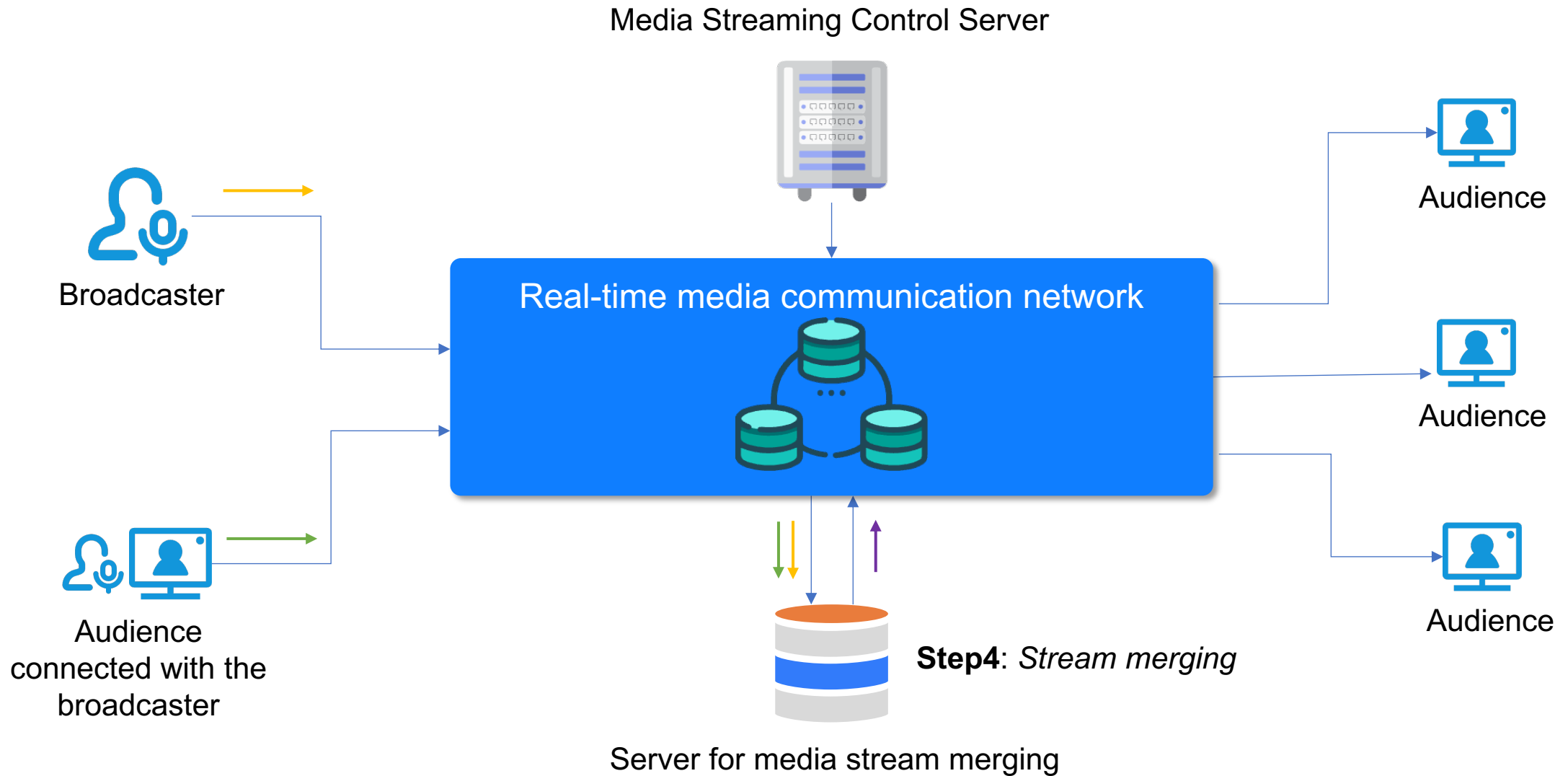https://www.ietf.org/archive/id/draft-liu-webrtc-http-interactive-protocol-00.txt

Use Case: Interactive live-webcast

Media Streaming Control Server

Step1: *Broadcaster ingest media stream*

Broadcaster

Audience connected with the broadcaster

Real-time media communication network

Server for media stream merging

Audience

Audience

Audience

Media Streaming Control Server

Broadcaster

Step2: *Audience want to interact with broadcaster*

Audience connected with the broadcaster

Real-time media communication network

Server for media stream merging

Audience

Audience

Audience

3

Media Streaming Control Server

Real-time media communication network

Broadcaster

Audience connected with the broadcaster

Step4: *Stream merging*

Server for media stream merging

Audience

Audience

Audience

5

Media Streaming Control Server

Step5: *Signaling for stream switching*

Broadcaster

Real-time media communication network

Audience

Audience

Audience

Audience connected with the broadcaster

Server for media stream merging

6

Media Streaming Control Server

Real-time media communication network

Broadcaster

Audience connected with the broadcaster

Step6: *Stream switch*

Server for media stream merging
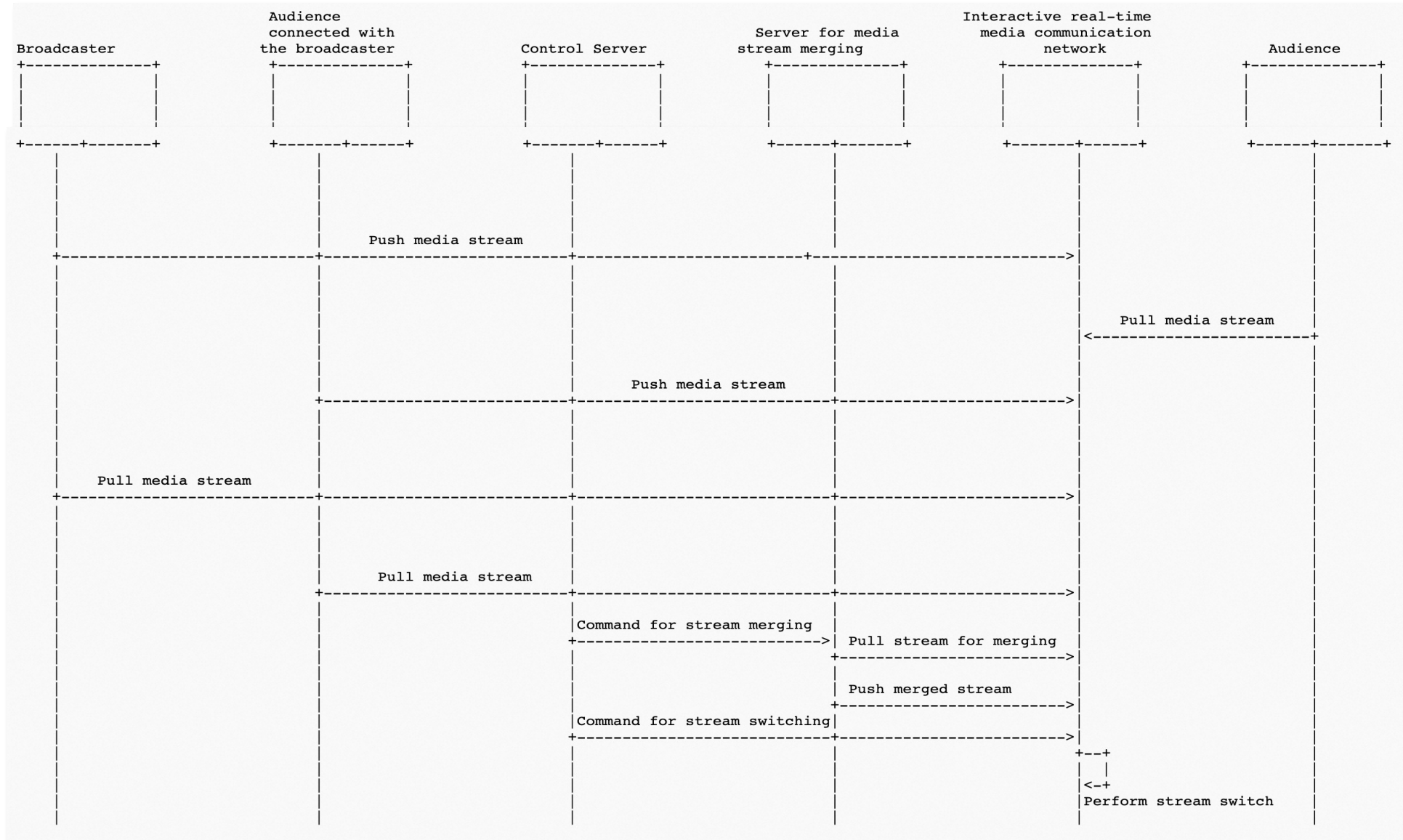
Audience

Audience

Audience

Signaling Procedure

Signaling Specification(1/4): Merging Signaling

```
Merging Message:
{

 POST /whisp/merging/endpoint HTTP/1.1
 Host: whisp.example.com
 Content-Type: application/json
 Content-Length:
 {

 "main media": {
    "media ID":[
      "amsid":[
        "rts audio"
      ],
      "vmsid":[
        "rts video"
      ]

    ],
    "URL":"http://demo.example.com/liveapp****/liveStream****1",
 }

 "secondary media": {
    "media ID":[
      "amsid":[
        "rts audio"
      ],
      "vmsid":[
        "rts video"
      ]

    ],
    "URL":"http://demo.example.com/liveapp****/liveStream****2",
 }

  "merge template": {
    "merge template id"[
       "01"
    ]
  }
 }
}

                    Figure 4: Merging signaling message
```

Signaling Specification(2/4): Switching Signaling

```
Switching Message:
{

  POST /whisp/switching/endpoint HTTP/1.1
  Host: whisp.example.com
  Content-Type: application/json
  Content-Length:
  {

   "source media": {
     "media ID":[
       "amsid":[
         "rts audio"
       ],
       "vmsid":[
         "rts video"
       ]

     ],
     "URL":"http://demo.example.com/liveapp****/liveStream****1",
  }


   "destination media": {
     "media ID":[
       "amsid":[
         "rts audio"
       ],
       "vmsid":[
         "rts video"
       ]

     ],
     "URL":"http://demo.example.com/liveapp****/liveStream****2",
  }
 }
}
```

Figure 5: Switching signaling message

Signaling Specification(3/4): Grabbing Signaling

```
Grabbing Message:
{

  POST /whisp/grabbing/endpoint HTTP/1.1
  Host: whisp.example.com
  Content-Type: application/json
  Content-Length:
  {


   "new media": {
     "media ID":[
       "amsid":[
         "rts audio"
       ],
       "vmsid":[
         "rts video"
       ]


    ],
    "URL":"http://demo.example.com/liveapp****/liveStream****",
   }
  error_code,
  }
}
```

Figure 6: Grabbing signaling message

```
+========+======================+
| Reason | Code                 |
+========+======================+
|    0x0 | Dropped by Mobility  |
+--------+----------------------+
|    0x1 | Proactive dropping   |
+--------+----------------------+
|    0x2 | Passive dropping     |
+--------+----------------------+
```

Table 2: Error code for
grabbing signaling message

Signaling Specification(4/4): Pulling Signaling

```
Pulling Message:
{

 POST /whisp/pulling/endpoint HTTP/1.1
 Host: whisp.example.com
 Content-Type: application/json
 Content-Length:
 {


  "media": {
    "URL":"http://demo.example.com/liveapp****/liveStream****",
 }
 }
}

            Figure 7: Pulling signaling message
```

Comments?