Protocols for Multiparty Multimedia Sessions

By: Chunyan Fu, PhD, Ericsson Canada
Fatna Belqasmi, PhD, Ericsson Canada

- Part I: introduction, signaling and media control protocols (Nov. 16, by Chunyan)
- Part II: floor control protocols, putting together and case study (Nov. 23, by Fatna)

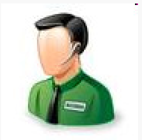
Part I: Introduction, signaling and media control protocols

- Introduction
  - What is multiparty multimedia session
  - Technical components
- Signaling protocols
  - H.323
  - SIP
- Media control protocols
  - Megaco (H.248)
  - SIP based protocols

- Introduction
  - What is multiparty multimedia session
  - How to implement
  - Protocols involved
  - Classifications

Multiparty multimedia session

- The conversational exchange of multimedia content between several parties
  - About multimedia
    - Audio, video, data, messaging



  - About participants
    - Any one who wants to participates the conference

How – thinking from a real life case

- When organizing a conference or a meeting, what to do?

Deciding topics, participants, time, agenda and booking a conf room → Policy control

Inviting participants and getting their confirmation to attend → Signaling

Starting the conference: let people seat down in the room and prepare the projector, microphone, player → Media control

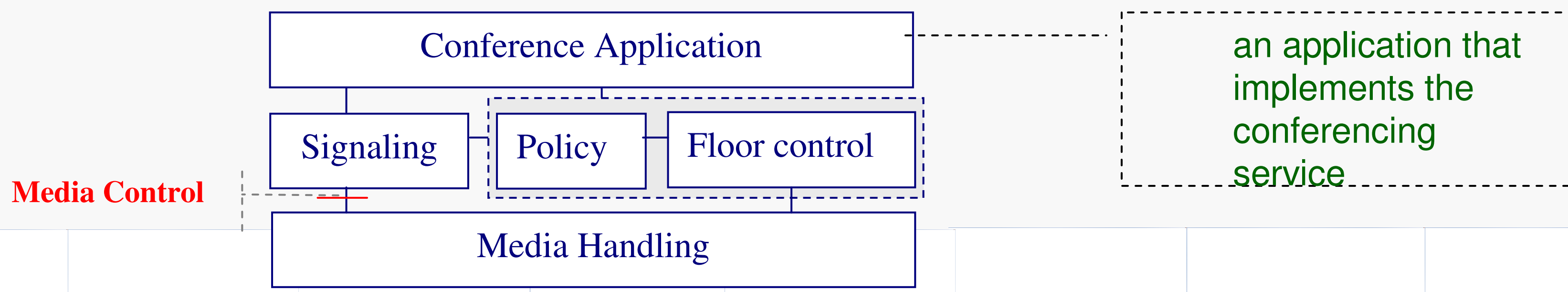
During a conference:

-- talking, discussing; presenting, playing a video to everybody, translating → Media handling

-- being a chair and deciding who can talk next → Floor control

How – technical components

- Signaling
  - Session establishment, modification and termination
  - Capability negotiation
- Media
  - Media handling: media transmission, mixing, trans-coding
  - Media control: stands when there is a separation of signaling and media mixing entities
- Conference control
  - Conference policy: conference arrangement, admission control, participant management, voting
  - Floor control: allows users of share resources such as video and audio without access conflicts.



Protocols involved

- Signaling
  - H.323, SIP (Session Initiation Protocol)
- Media
  - Media control: Megaco (Media Gateway Control protocol), SIP based media control – NetAnn/SIP MSCML (Media Server Control Markup Language), SIP media control channel framework
  - Media transport: RTP/RTCP, SRTP
- Conference control
  - Policy control: CPCP (conference policy control protocol), XCAP
  - Floor control: BFCP (Binary Floor Control Protocol), TBCP (Talk Burst Control Protocol)
    - Floor server control: FSCML (Floor Server Control Markup Language)

Classifications

- Open/close
- Pre-arranged/ad hoc
- With/without sub-conferencing (i.e. sidebar)
- With/without floor control
- Topology: centralized, distributed, hybrid

- Signaling protocols
  - ITU-T: H.323
    - Basic
    - Conferencing models
    - Scenarios
  - IETF: SIP
    - Basic
    - Conferencing models
    - Scenarios
  - H.323 vs. SIP

What is H.323 ?

- ITU-T standard for the transmission of real-time audio, video, and data communications over packet-based networks.
- Defines components, protocols and procedures
- Can be applied for multiparty multimedia session

H.323 Network Components (1)

- Terminal can either be a personal computer (PC) or a stand-alone device
  - Running an H.323 stack
  - Running the multimedia applications

- Gatekeeper (optional) is the brain of H.323
  - Addressing
  - Authorization and authentication of terminals and gateways
  - Bandwidth management; accounting
  - Billing and charging
  - Call-routing

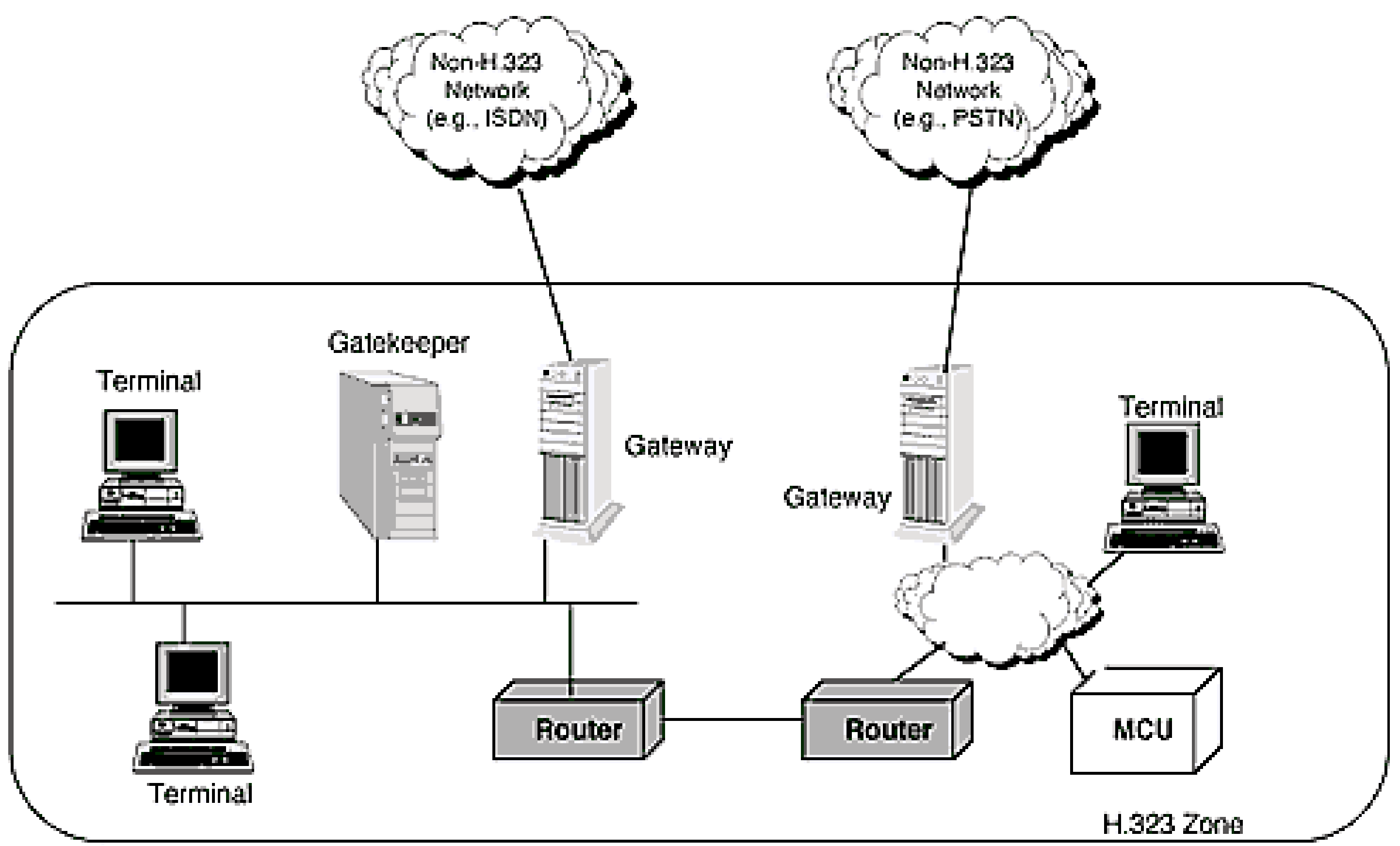
H.323 Network Component (2)

- Gateway (optional) provides connectivity between an H.323 network and a non–H.323 network (e.g. PSTN)
  - Translating protocols
  - Converting media format

- Multipoint control unit (optional) provides conferences of three or more H.323 terminals
  - MC: multipoint controller handles control signaling
  - MP: multipoint processor handles media mixing (optional)

H.323 Zone

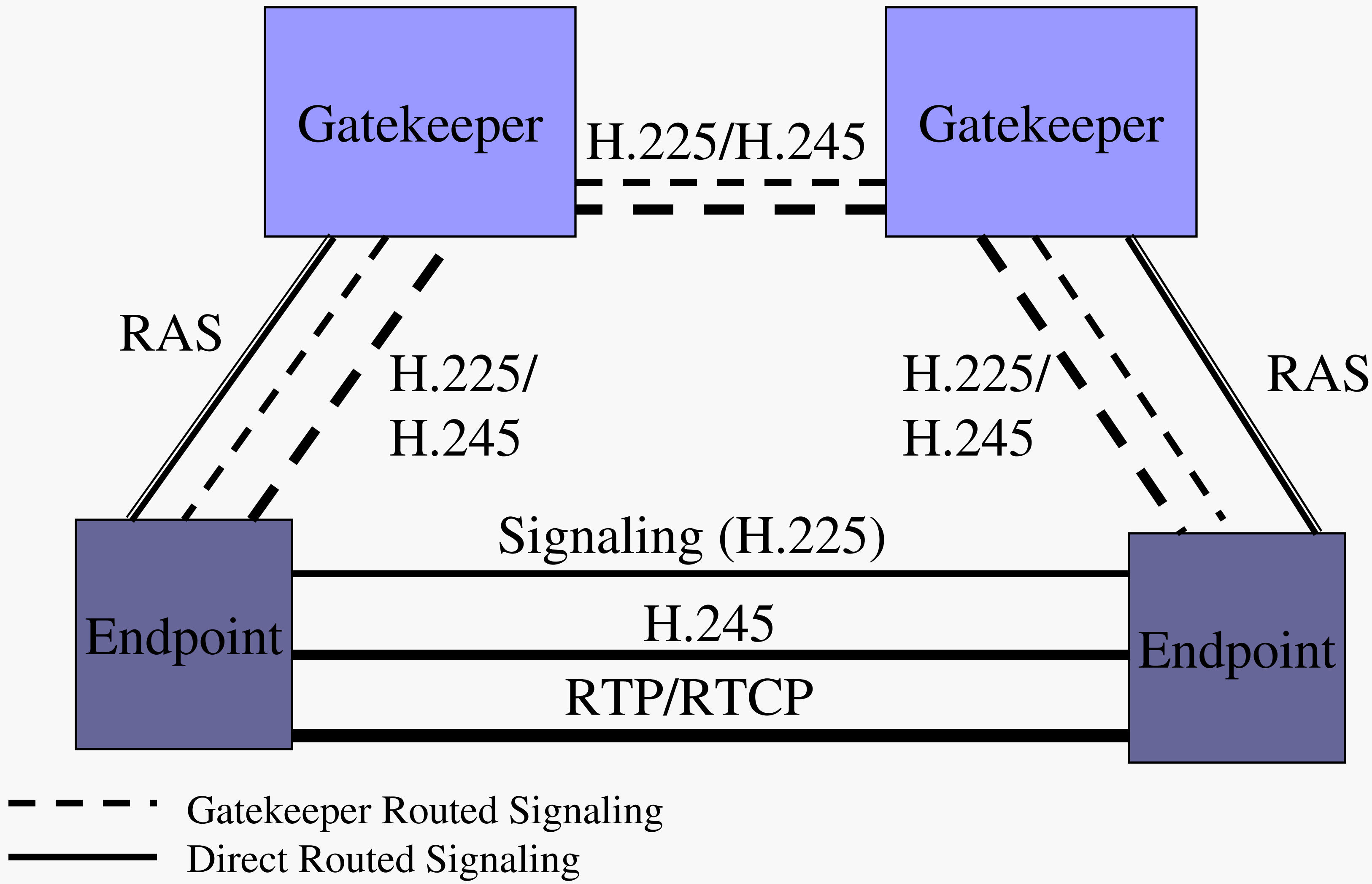
- Terminals, gateways, MCUs managed by one gatekeeper



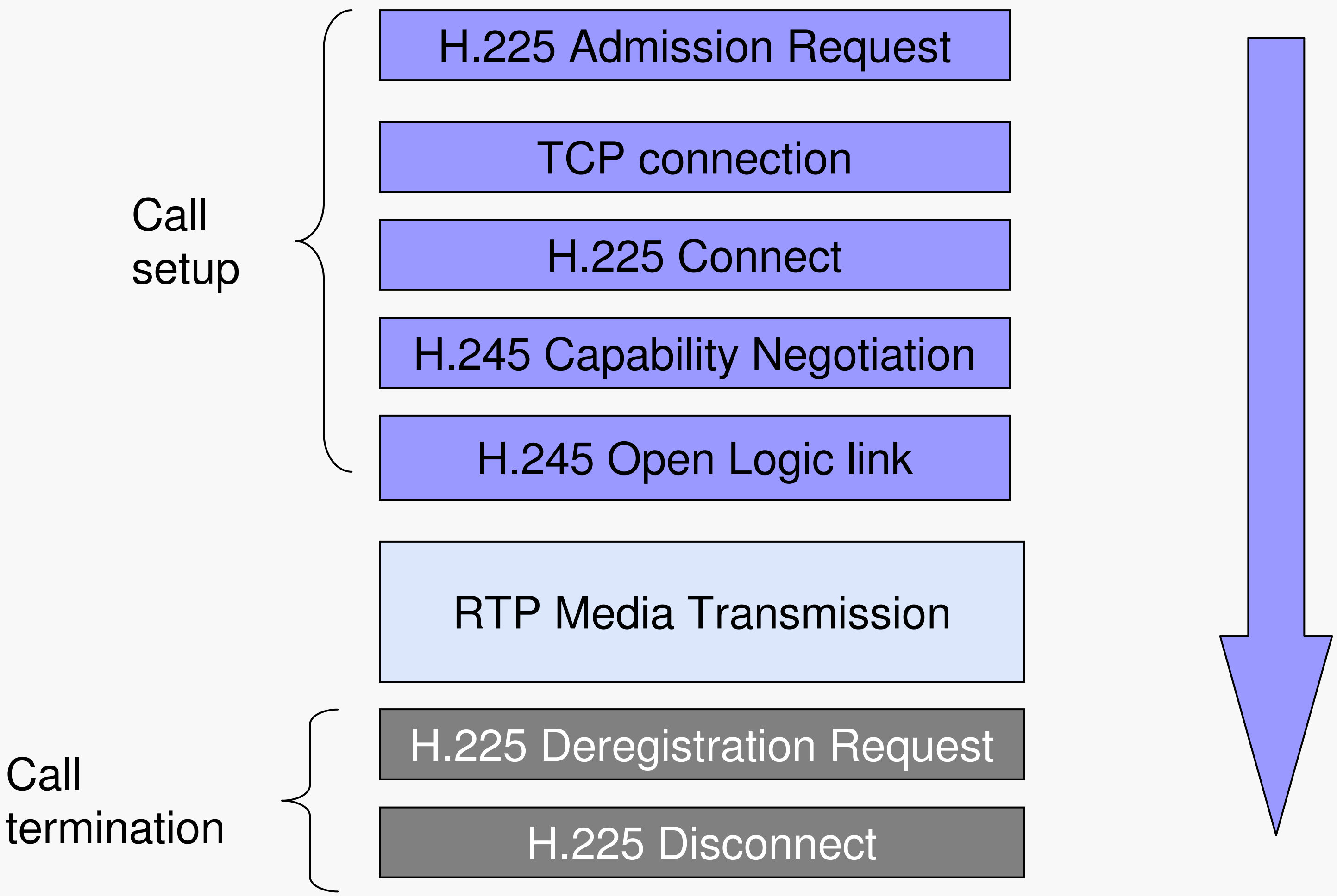
H.323 protocols

- H.225 call signaling
  - Call setup, termination
- H.225 RAS: registration, admission, and status
  - gatekeeper discovery (GRQ)
  - endpoint registration
  - endpoint location
  - admission control
- H.245 control signaling
  - capability negotiation
  - open logical channel
- Media transmission: RTP/RTCP

H.323 protocols

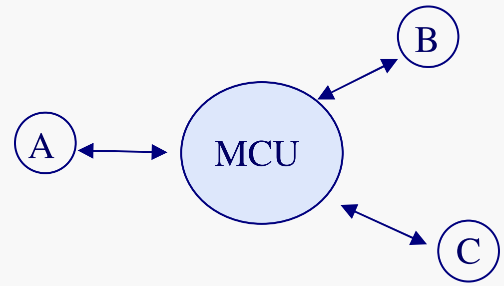


H.323 basic procedure



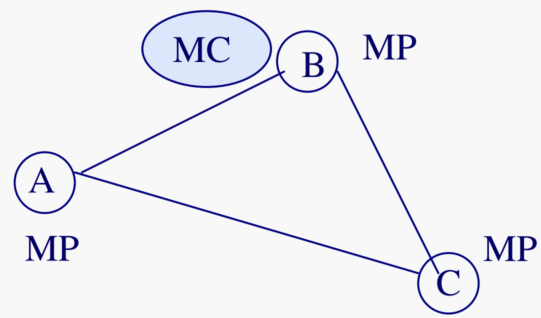
H.323 conferencing models

- Centralized Conference
  - MCU is required
  - All terminals send media and control signals to MCU in point to point fashion.



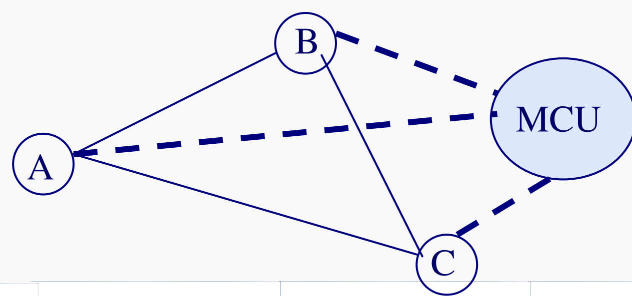
Centralized Conference

- Decentralized Conference
  - Terminals multicast media to others
  - Control signaling is still in MCU



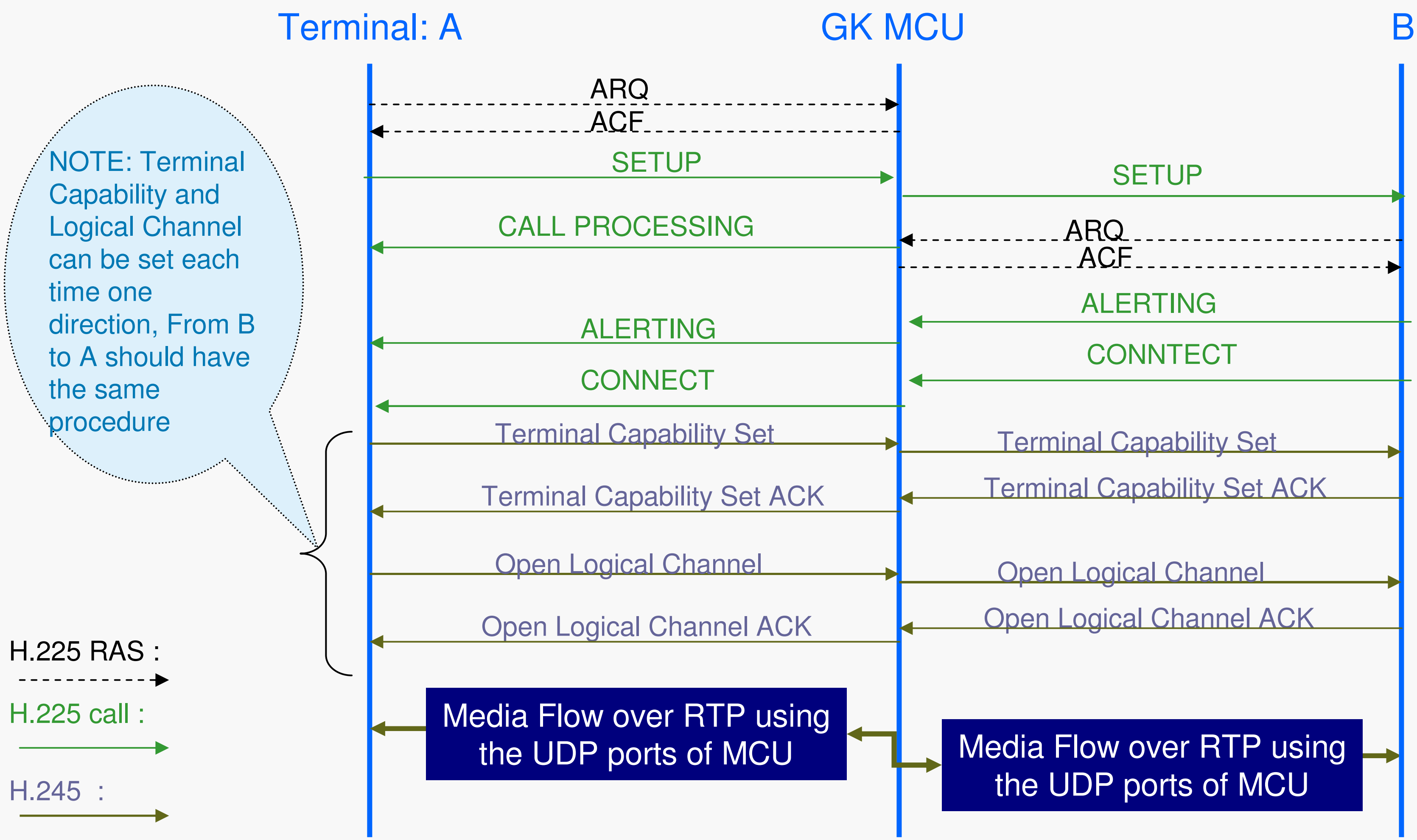
Distributed Conference

- Hybrid Conference
  - Control signaling and audio processing is done by MCU
  - Video using multicast

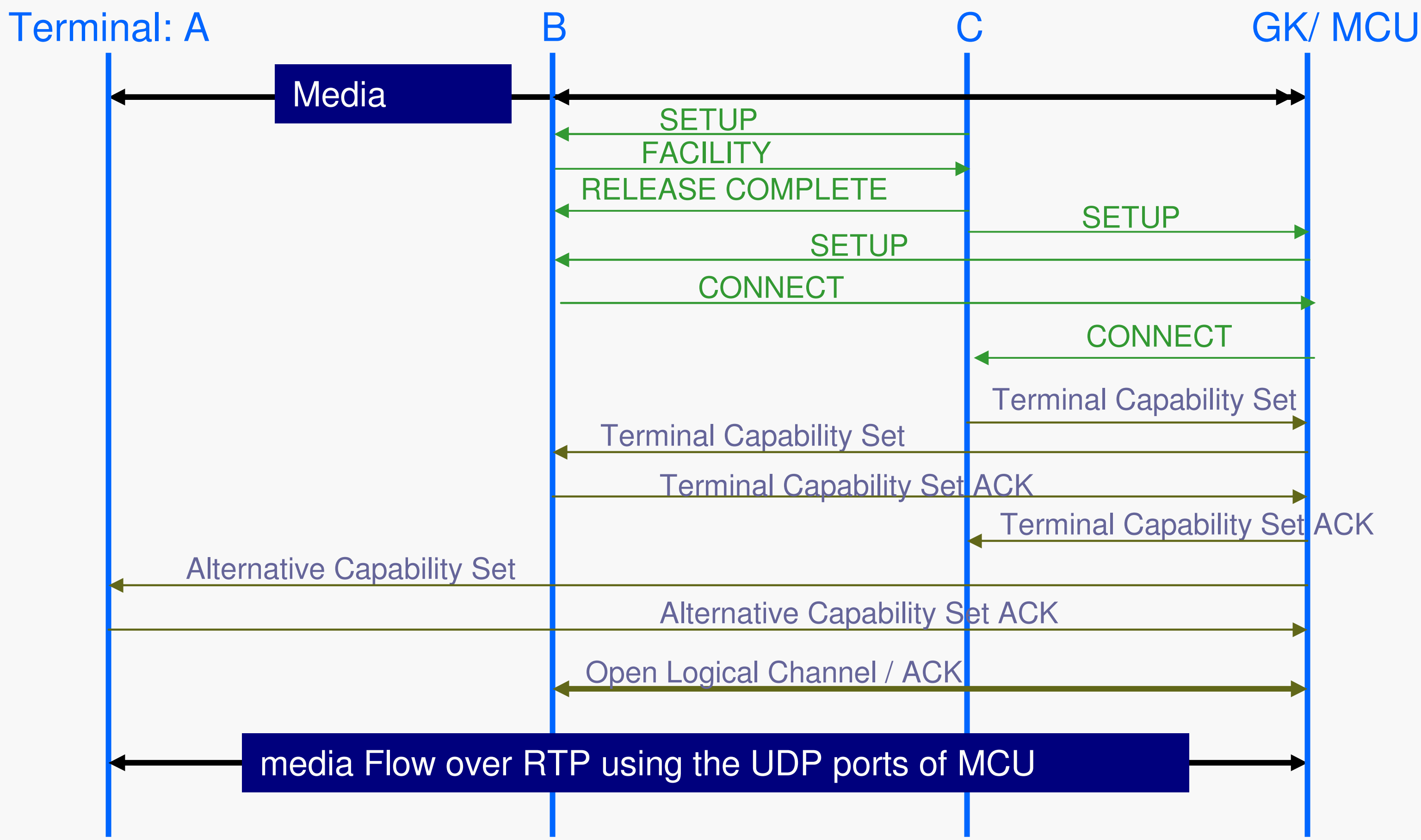


Hybrid: Centralized audio, distributed video

H.323 conference initiation example



H.323 participant join example



What is SIP

- IETF standard to set up, modify and terminate media sessions over IP, operating in a request/response model
- Reuse Internet addressing (URLs, DNS, proxies), reuse HTTP coding, text based application-layer protocol
- A basic line protocol and extensions
- It can be applied for multiparty multimedia session control

SIP entities

- User Agents
  - UAC: originates request
  - UAS: processes requests and sends response
- SIP Servers
  - Registrar: registration of user's contact addresses
  - Proxy: decides next hop and forwards request
  - Redirect: sends address of next hop back to client
  - Location server: provides user location details

SIP requests

- From baseline (RFC 3261)
  - INVITE initiate a session
  - ACK confirm final response
  - BYE terminate a session
  - CANCEL cancel an ongoing session
  - OPTIONS features support by other side
  - REGISTER register with location service
- Most relevant extensions
  - INFO mid-call information (RFC 2976)
  - SUBSCRIBE subscribe to event (RFC 3265)
  - NOTIFY notify subscribers
  - REFER ask recipient to issue SIP request (RFC 3515)
  - PUBLISH publish event and presence state (RFC3903)
  - PRACK provisional acknowledgement (RFC 3262)

SIP responses

SIP Responses defined as HTTP-style

	Description	Examples
1xx	Informational – Request received, continuing to process request.	180 Ringing 181 Call is Being Forwarded
2xx	Success – Action was successfully received, understood and accepted.	200 OK 202 Accepted
3xx	Redirection – Further action needs to be taken in order to complete the request.	300 Multiple Choices 302 Moved Temporarily
4xx	Client Error – Request contains bad syntax or cannot be fulfilled at this server.	401 Unauthorized 408 Request Timeout
5xx	Server Error – Server failed to fulfill an apparently valid request.	503 Service Unavailable 505 Version Not Suported
6xx	Global Failure – Request is invalid at any server.	600 Busy Everywhere 603 Decline

SIP headers - partial list

Header	Description	Examples
From	Required field containing the originating SIP URI, may also has display name	From: SIP: Alice@example.com From: SIP: +1-514-2345678@gateway.com f: SIP: Bob@192.168.1.100
To	Required field containing the recipient SIP URL. May contain a display name.	To: SIP: Alice@example.com To: SIP: +1-514-2345678@gateway.com t: SIP: Bob@192.168.1.100
Call-ID	Used to uniquely identify a particular session or registration messages. Should have randomness to ensure overall global uniqueness.	Call-ID: 1@mars.brooks.net Call-ID: Jan-01-1999-1510- 1@server.mci.com i: 31415926535@uunet.com
Contact	Alternative SIP URL for more direct message routing.	Contact: W. Riker, Acting Captain <riker@starfleet.gov> Contact: room203@hotel.com; expires=3600 m: admin@mci.com
Content-Length	Octet count in message body.	Content-Length: 285
Content-Type	Content type of message body	Content-Type: application/sdp c: application/h.323
CSeq	Command Sequence number – used to distinguish different requests during the same session.	CSeq: 1 INVITE CSeq: 1000 INVITE CSeq: 4325 BYE CSeq: 1 REGISTER
Via	Used to show the path taken by a request	Via: SIP/2.0/UDP sip.mfs.com Via: SIP/2.0/TCP uunet.com v: SIP/2.0/UDP 192.168.1.1
Max-Forwards	Count by decrease when pass a hop, if reach 0, return 483 too many hops error	Max-Forwards: 70

SIP dialog

- Dialog (call leg): to facilitate the session management
  - From
  - To
  - Call-ID
- For a response:
  - Via, From, To, Call-ID, and CSeq are copied exactly from Request.
  - To and From are NOT swapped
- Sequence number (Cseq) increases when a new request sent within a dialog

SIP message body

- Message body can be any…
- Most implementations:
  - SDP - Session Description Protocol
    - Used to specify info about a multi-media session.
    - SDP fields have a required order
    - Offer/Answer mode
  - XML
    - Used for different purposes, e.g. messaging, presence information
    - The formats are defined by other standards, e.g. PIDF (RFC 3863)

SDP examples

SDP Example 1

v=0
o=alice +1-514-555-1212 IN IP4 example.com
s=Let's Talk
t=0 0
c=IN IP4 101.64.4.1
m=audio 49170 RTP/AVP 0 3

SDP Example 2

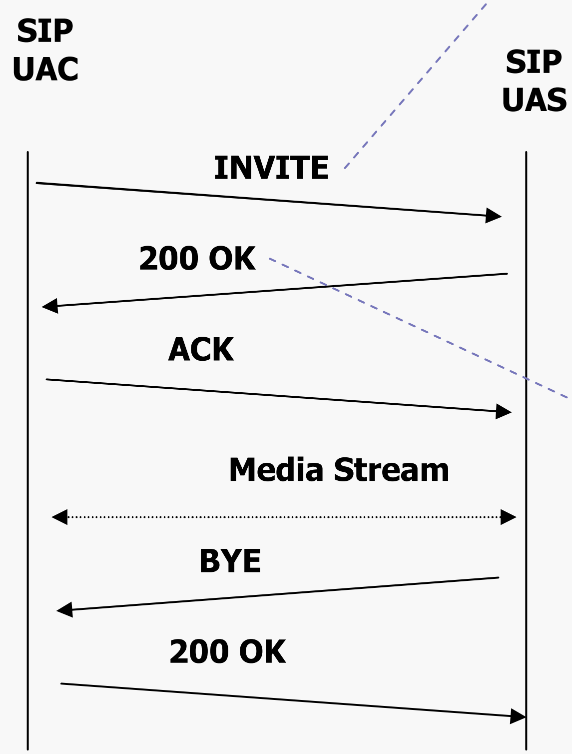
v=0
o=bob 124333 67895 IN IP4 example1.com
s=Yes!
t=0 0
c=IN IP4 101.234.2.1
m=audio 3456 RTP/AVP 0

Field	Descripton
Version	v=0
Origin	o=<username> <session id> <version> <network type> <address type> <address>
Session Name	s=<session name>
Times	t=<start time> <stop time>
Connection Data	c=<network type> <address type> <connection address>
Media	m=<media> <port> <transport> <media format list>

An XML example

```
PUBLISH sip:example.com SIP/2.0
Event: presence
CSeq: 1 PUBLISH
From: "Alice"<sip:alice@example.com>;tag=1138
To: "Alice"<sip:alice@example.com>
…
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<presence entity="pres:alice@example.com"
   xmlns="urn:ietf:params:xml:ns:pidf">
    <tuple id="1">
        <status>
            <basic>open</basic>
        </status>
        <contact
   priority="0.800000000000000044408920985006261616945266723
   6328125">sip:alice@example.com</contact>
        <timestamp>2009-06-03T09:31:18.689-
   00:05</timestamp>
        <timed-status until="2009-06-03T10:31:18.677-00:05"
   from="2009-06-03T09:31:18.677-00:05"/>
</tuple>
</presence>
```

SIP example



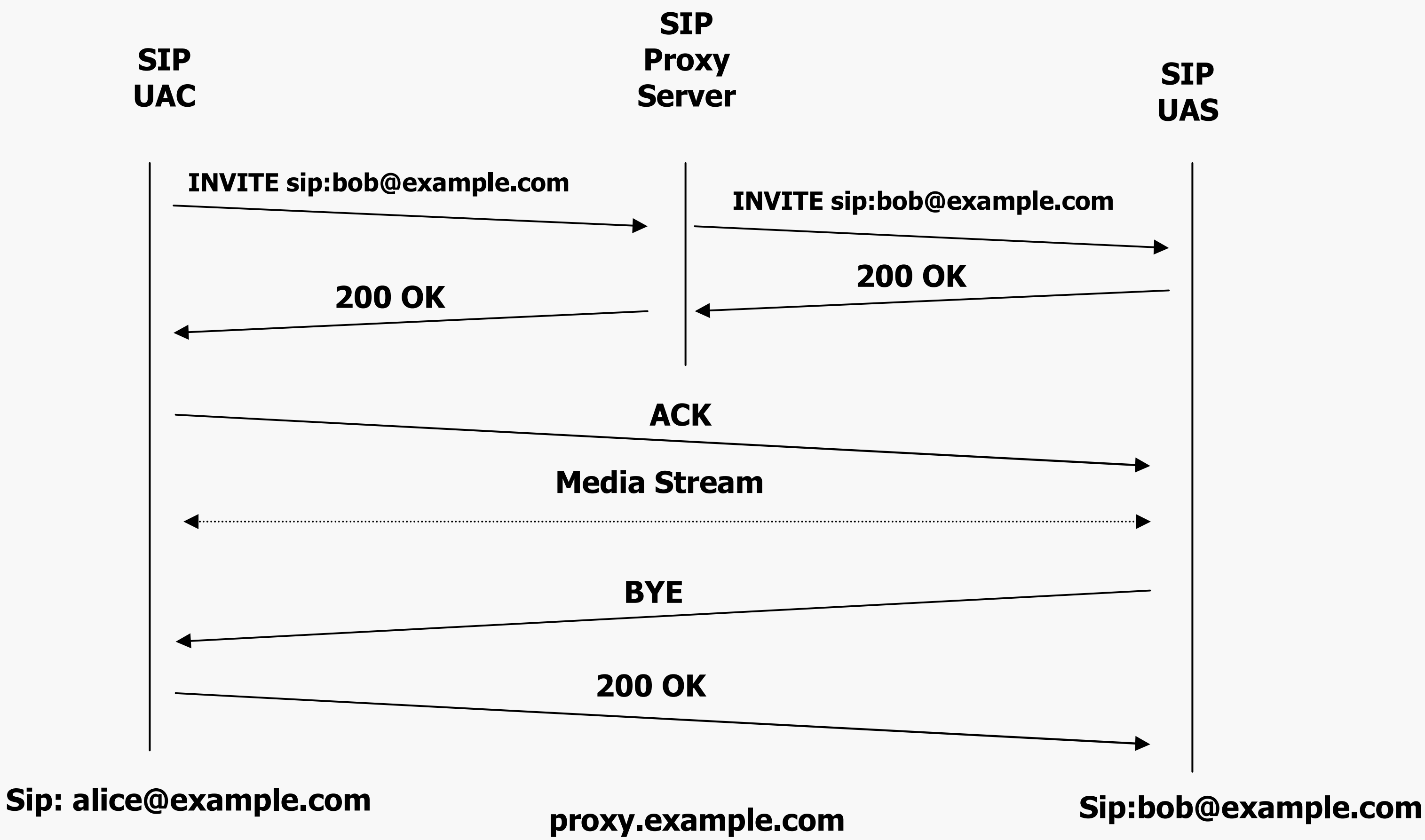

```
INVITE sip:bob@example.com SIP/2.0
Via: SIP/2.0/UDP example.com:5060
From: Alice<sip:alice@example.com>
To: Bob<sip:bob@example.com>
Call-ID: 314159@example.com
CSeq: 1 INVITE
Contact: sip:alice@example.com
Content-Type: application/sdp
Content-Length: 124

v=0
o=alice 5462346 332134 IN IP4
s=Let's Talk
t=0 0
c=IN IP4 10.64.1.1
m=audio 49170 RTP/AVP 0 3
```

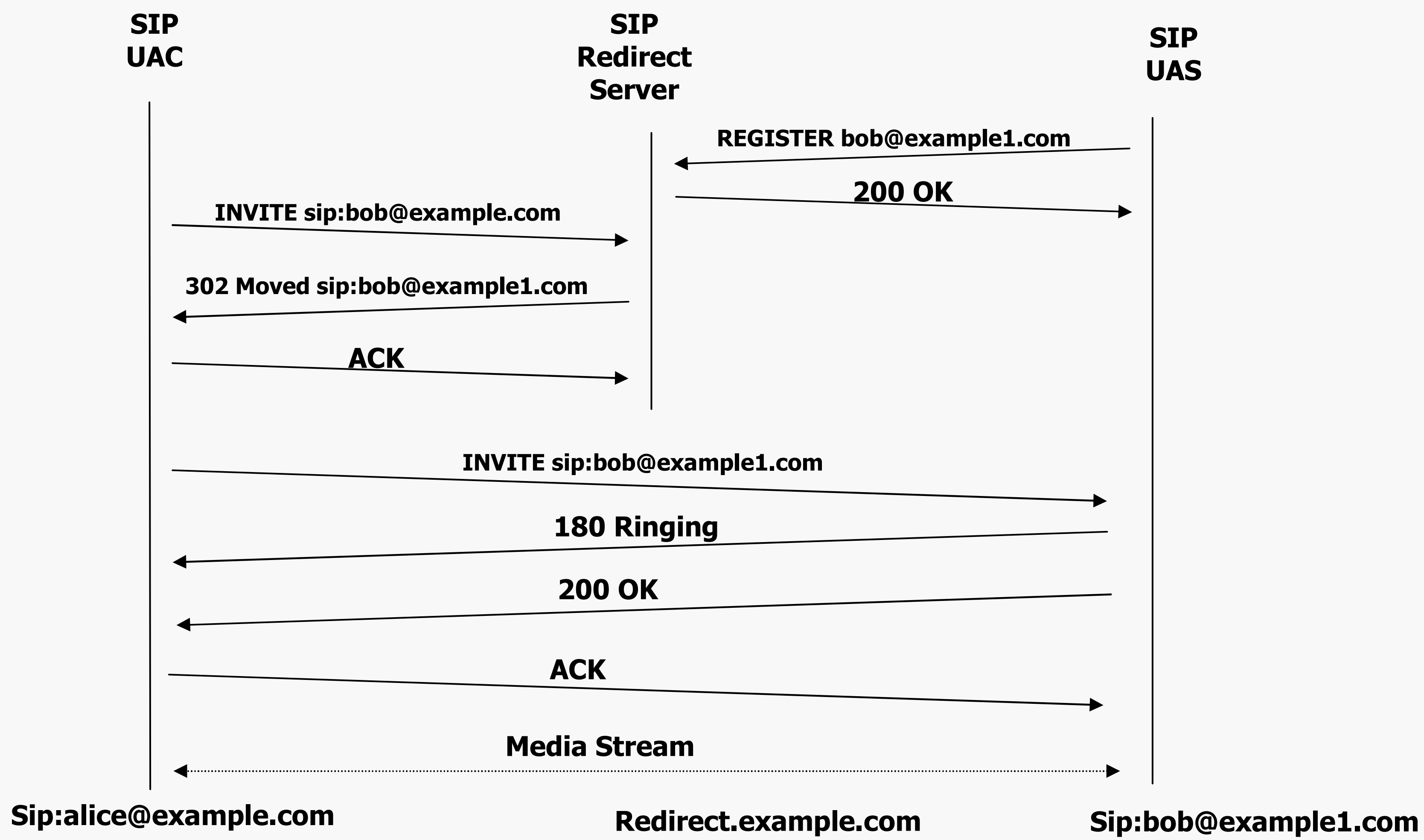
```
SIP/2.0 200 OK
Via: SIP/2.0/UDP example.com
From: Alice<sip:alice@example.com>
To: Bob<sip:bob@example.com>
Call-ID: 314159@example.com
CSeq: 1 INVITE
Contact: sip:bob@example.com
Content-Type: application/sdp
Content-Length: 107

v=0
o=bob 124333 67895 IN IP4
s=Yes!
t=0 0
c=IN IP4 11.234.2.1
m=audio 3456 RTP/AVP 0
```

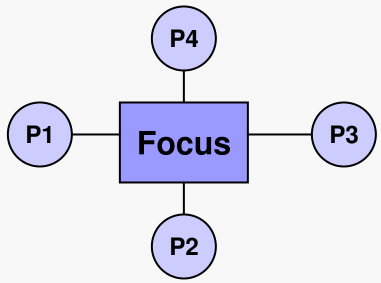
SIP proxy server example



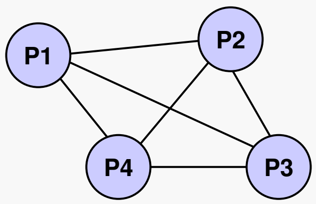
SIP redirect server example



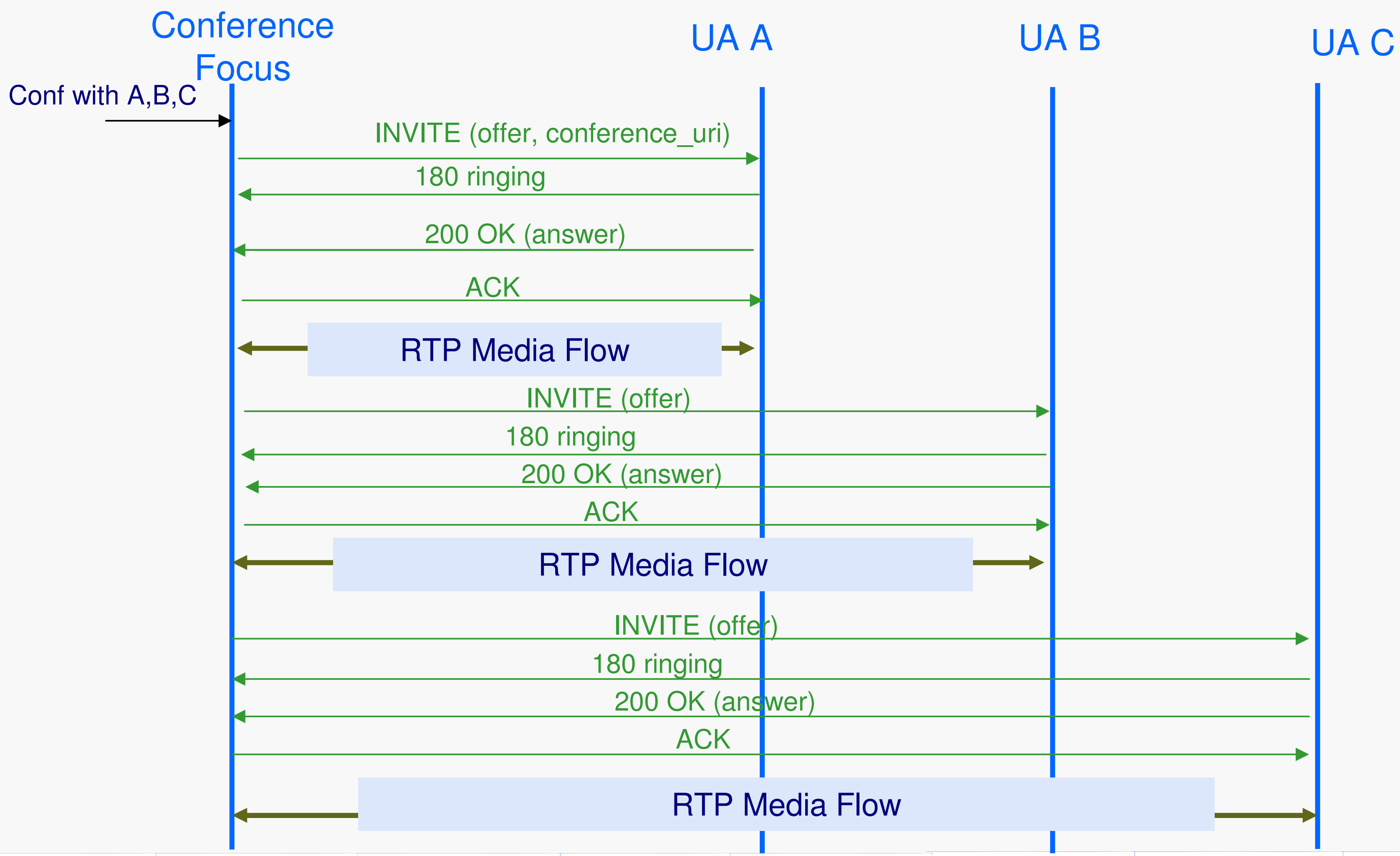
SIP conferencing models



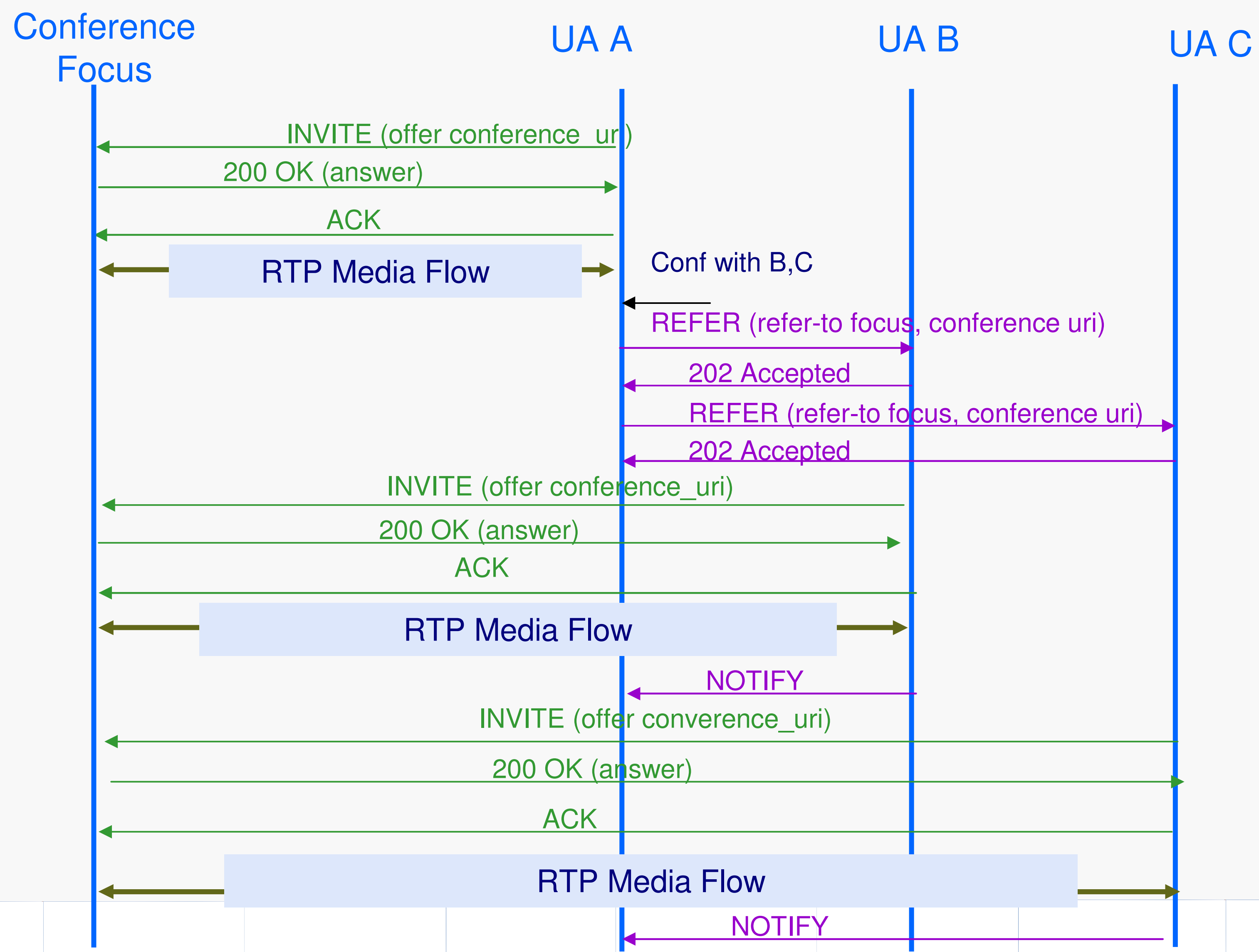
- Tightly coupled conference
  - Dial-In Conference
    - End point invite conference server which handle the media mixing
  - Dial-Out Conference
    - Server invite all the parties into a conference
  - Ad-hoc Centralized Conference
    - Two party setup conference directly, other party added through a conference server
- Loosely coupled
  - central signaling with multicast media
- Fully distributed



SIP conference example – dial out



SIP conference example – dial in



H.323 vs. SIP

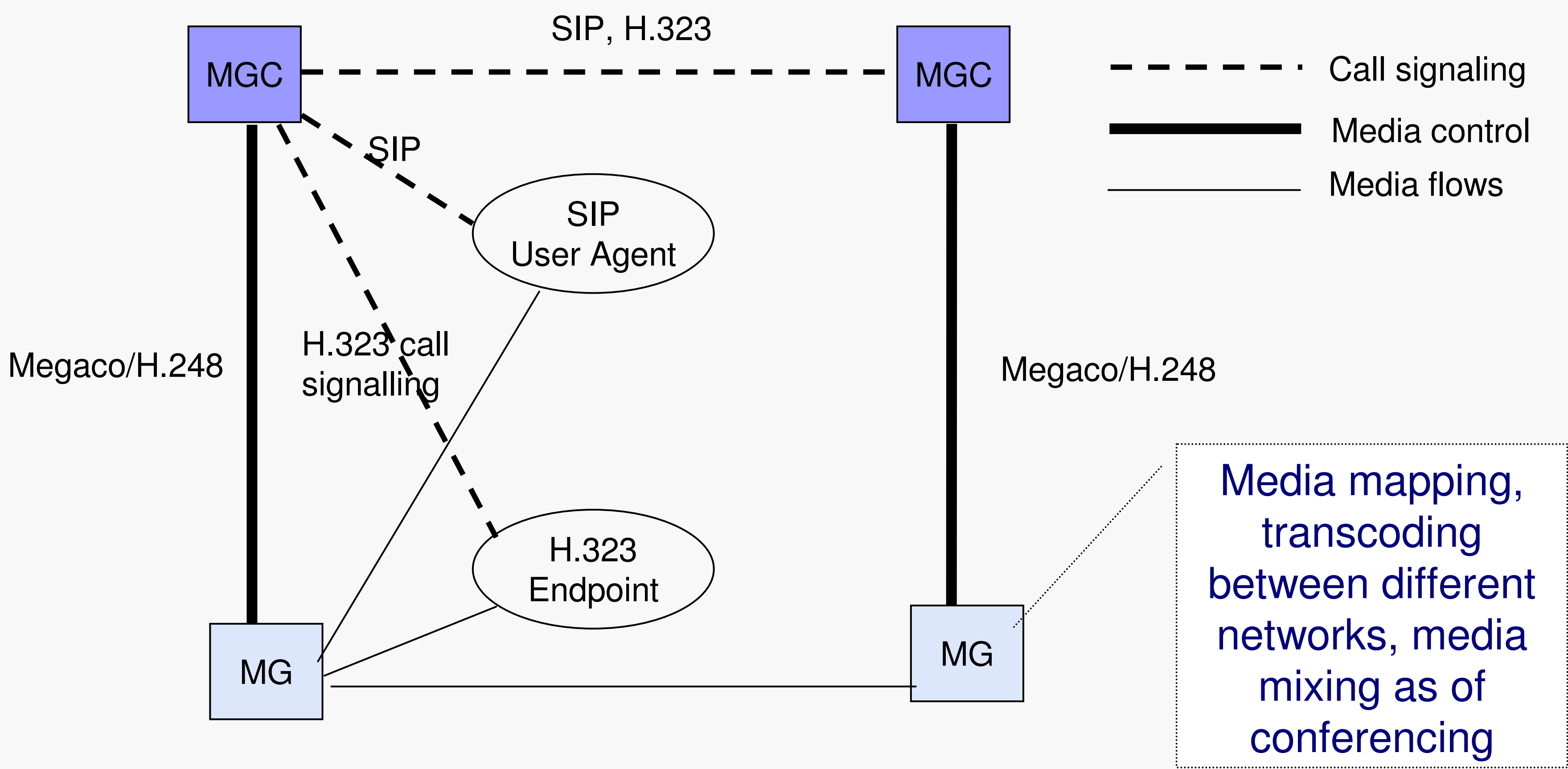
- Different people has different views, but what's in common
  - H.323 is matured and better for video conferencing, with less interoperability issues and lots of deployment
    - With concept of call, linked tightly to telephony services, call, conferencing…
  - SIP is a keep growing standard, being chosen as the core signaling protocol of 3GPP IP Multimedia Subsystem, being today's developers choice
    - With concept of session, linked tightly to internet services, presence, messaging, the standard for conferencing is still ongoing and will reach there

- Media control protocols
  - H.248/Megaco
    - Basic concept
    - Conference control
  - SIP Based Media Control
    - NetAnn (RFC 4240)
    - MSCML
    - SIP media control channel framework
  - Megaco vs. SIP based media control

What is Megaco/H.248

- Megaco/H.248 is an implementation of the Media Gateway Control architecture (defined by RFC 2805 requirements) for controlling Media Gateways on IP networks and PSTN.
- The result of collaboration IETF (RFC 3015) and ITU-T (H.248) groups
- A separation of call control and media conversion
- A master/slave protocol

Megaco components and signaling protocols



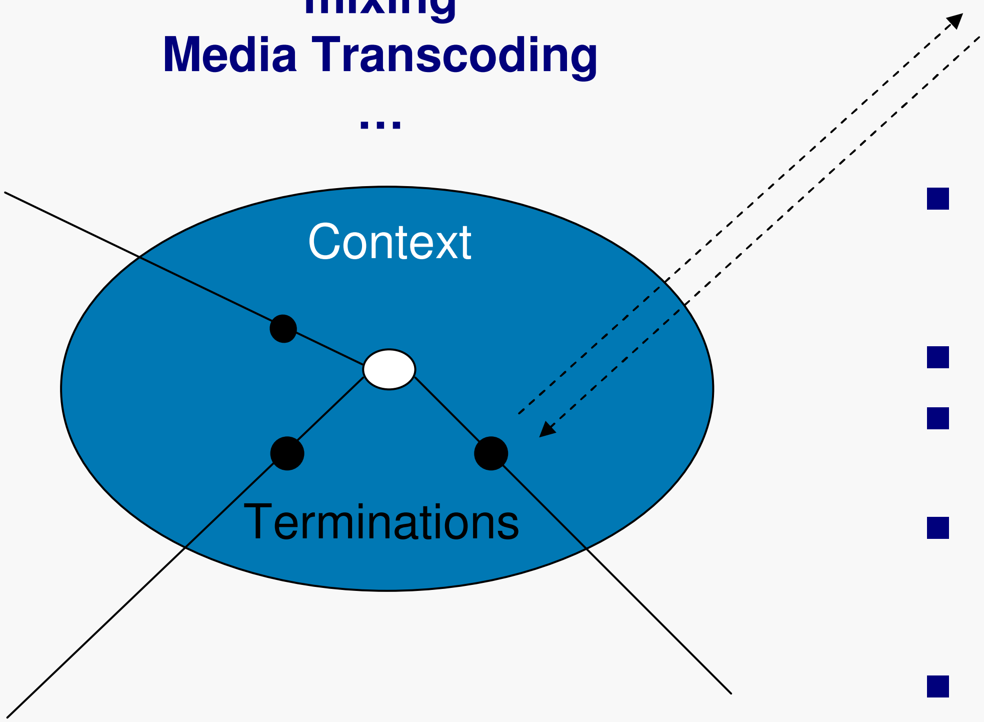
Megaco concepts

- Connection model
  - terminations, streams, and the context
- Termination properties
  - descriptors
- Context properties
- Message structure
  - transactions, actions, and commands
- Events and signals

Megaco functions

Media connection, mixing
Media Transcoding
...

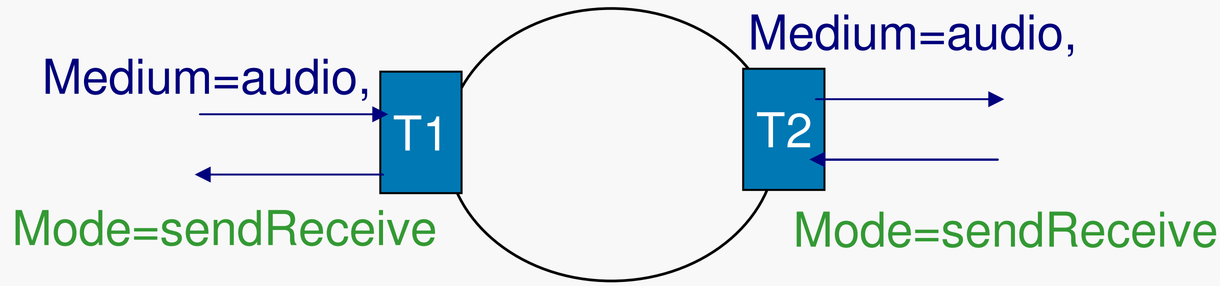
Events and Signals



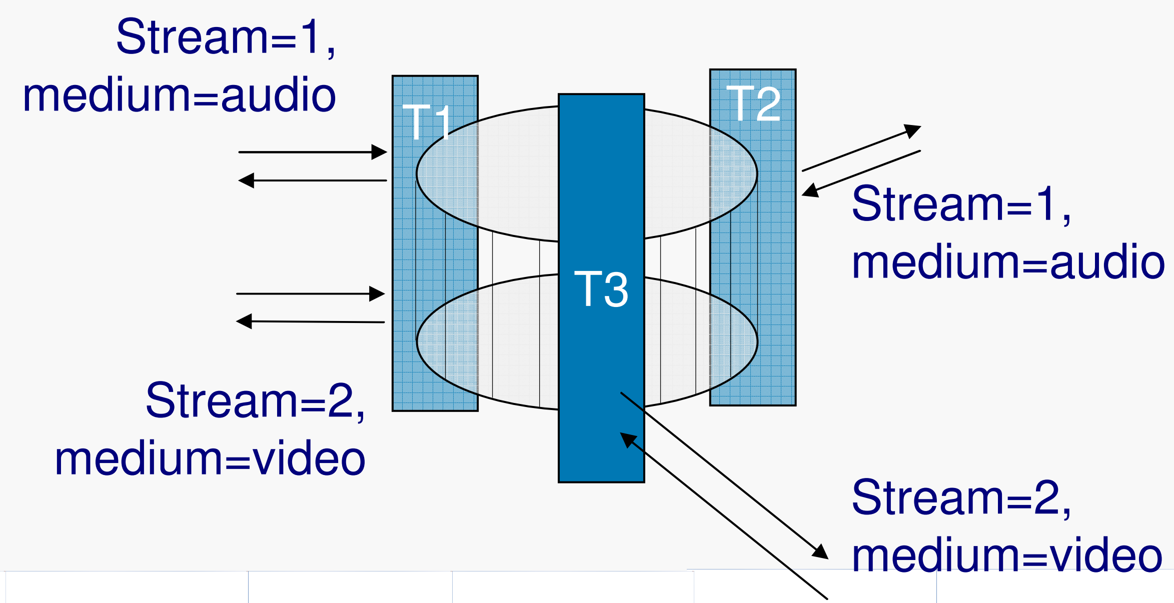
- Media connections achieved by placing two or more terminations into a common context
- Context viewed as mixing bridge
- Termination = source or sink of media flows
- Transport, medium, encoding/decoding specified per stream at each termination
- Flows specified by stream
- Context support multiple media streams

Megaco context examples

- Basic call



- Multimedia



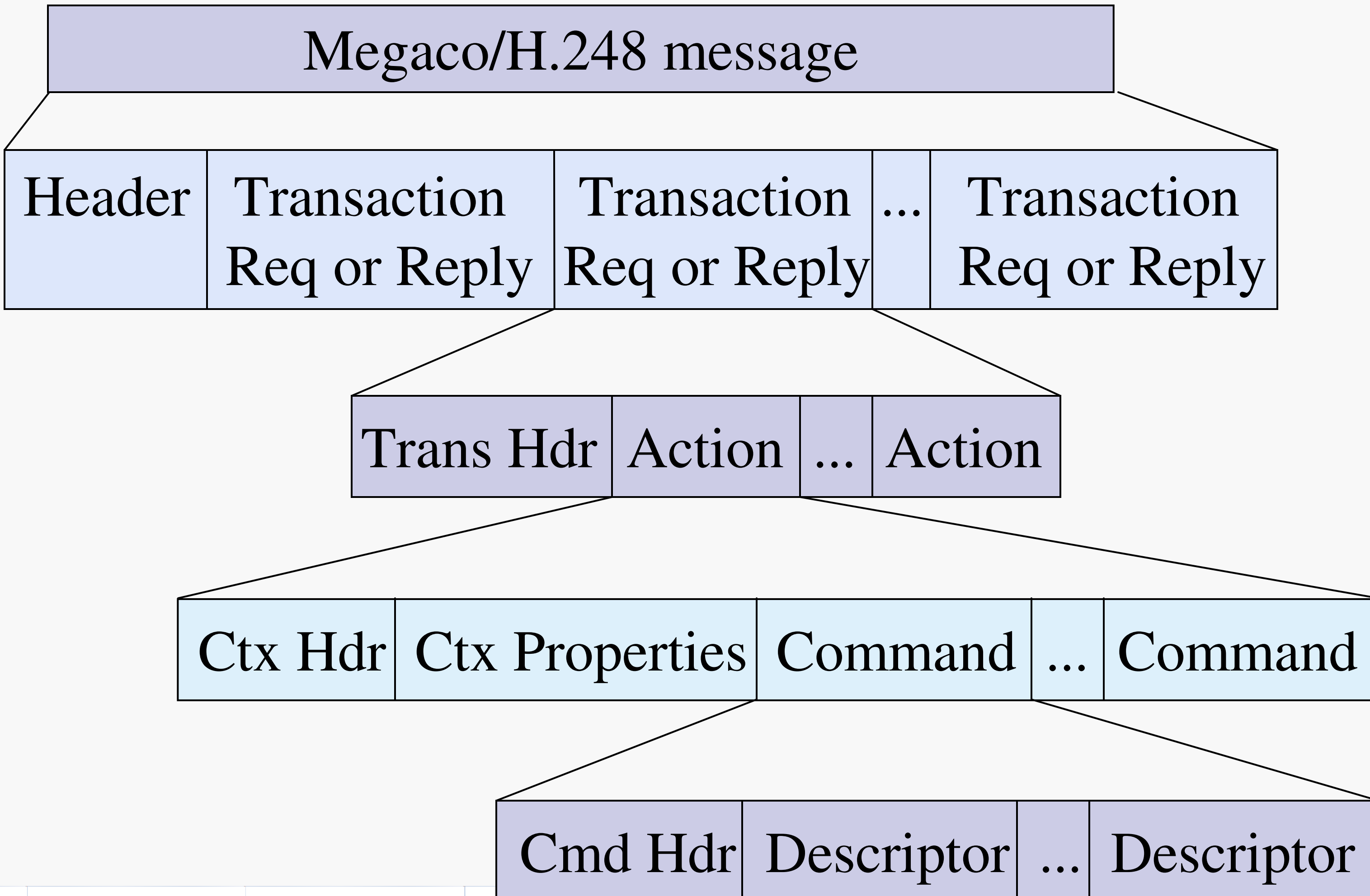
Megaco termination property

- Properties of terminations are organized syntactically into descriptors
  - basic ones are Termination State, Media, Events, and Signals descriptors
    - Media descriptor actually composed of other descriptors: Stream descriptors, which in turn contain LocalControl, Local, and Remote descriptors
- Default property values can be configured in the MG

Megaco context properties

- Three properties so far specified for a context
  - topology descriptor allows detailed specification of connectivity between individual pairs of terminations
  - priority flag can guide MG's allocation of scarce resources
  - emergency flag can indicate contexts which must be maintained and restored in the event of failures
- Null context introduced as a convention where persistent terminations are held when they are not in a real context
  - When terminations are returned to the null context, they take on their configured default property values.
- ROOT termination represents the MG itself

Megaco message structure



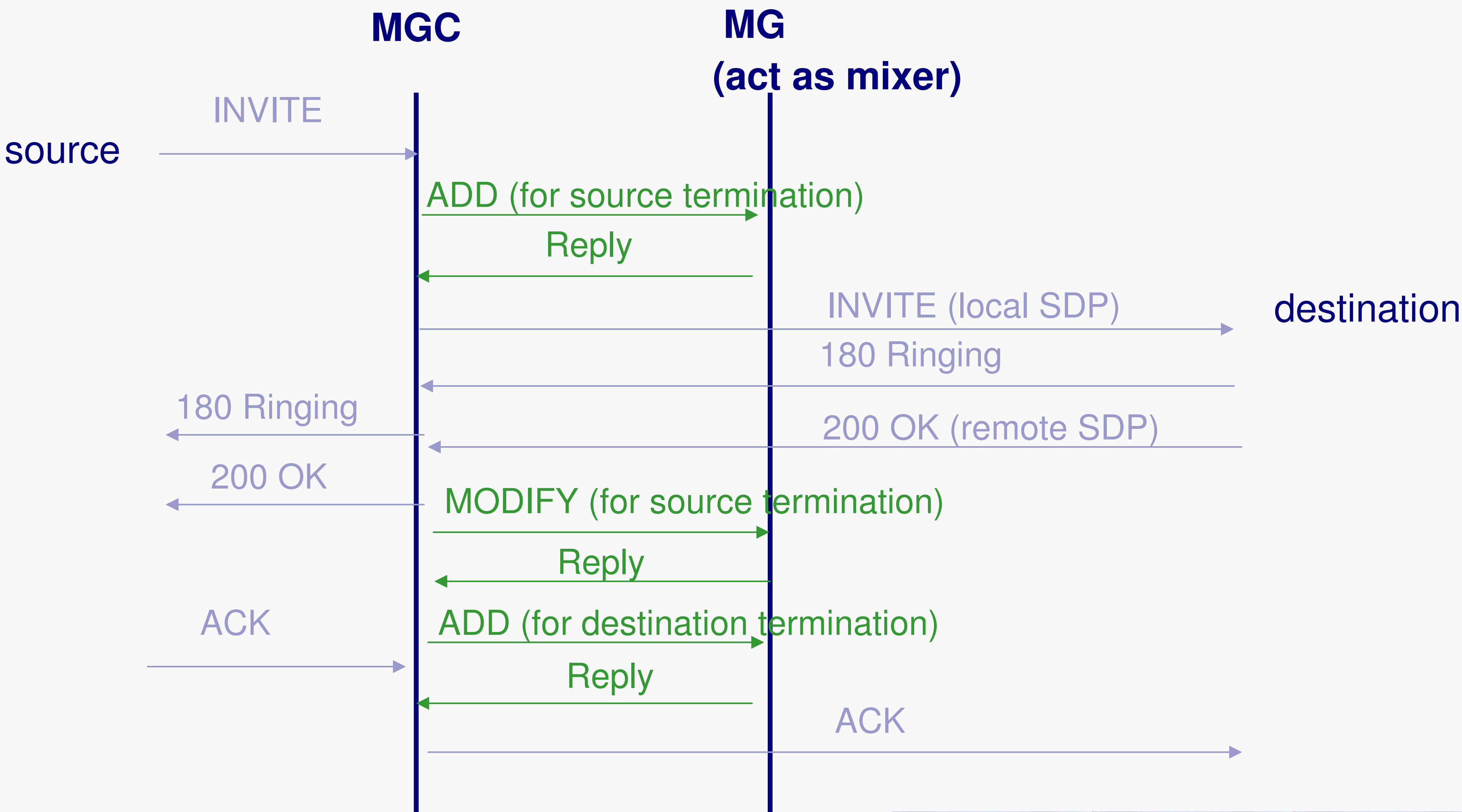
Megaco commands

- Megaco/H.248 provides the following commands
  - For termination manipulation
    - Add, Subtract, Move, Modify
  - For event reporting
    - Notify
  - For management
    - AuditCapability, AuditValue, ServiceChange

Events and signals

- Events are detected at the MG and reported to the MGC
  - MGC controls what events it wants to learn about at any given time
  - sets the termination Events descriptor
- Signals cause things to happen on terminations
  - play a tone, display text, ...
  - Specified in the Signals descriptor for a termination
  - MGC can specify duration of signal ahead of time or signal can play until explicitly stopped
  - Signals stop playing when any event is detected unless MGC says otherwise.

Megaco message flow example



- SIP based media control protocols
  - NetAnn (RFC 4240)
  - MSCML (RFC 5022)

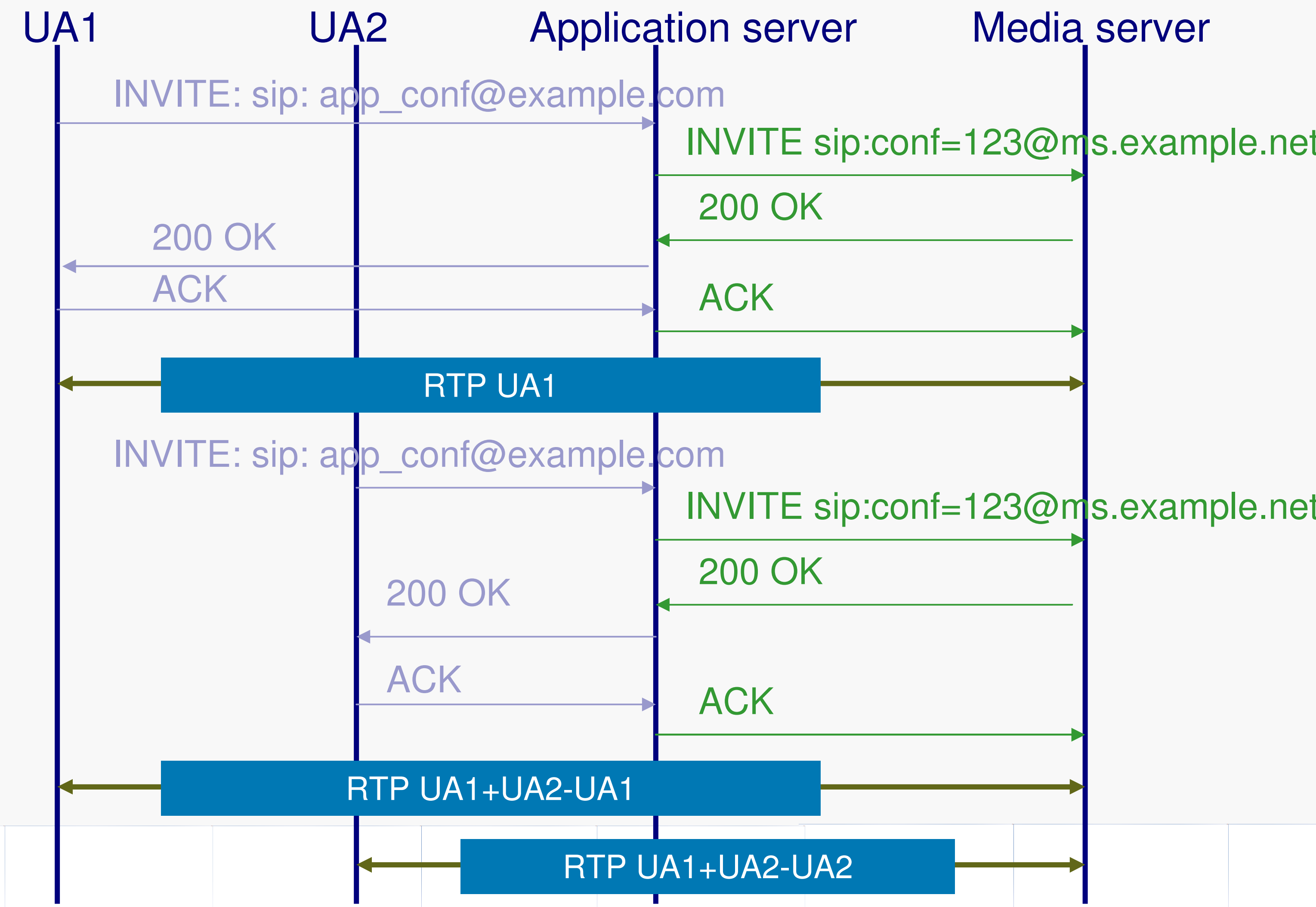
What is NetAnn

- Defined by IETF RFC 4240, provide basic media control service for applications in SIP based network
- The control is between media server (MS) and application server (AS)
- Predecessor to MSCML.
- Basic announcements, simpler conference model, doesn’t provide for mid-call requests and responses

Basic concept

- Use SIP request URI as service indicator
  - Can pass SIP servers without being treated as a ‘bad’ request
- Use SIP INVITE message
  - Examples
    - INVITE sip:annc@ms2.example.net; play=http://audio.example.net/allcircuitsbusy.g711
    - INVITE sip:conf=123@ms.example.net

NetAnn sequence example - conferencing



What is MSCML

- Defined initially by RFC 4722, replaced by RFC 5022
- Same as NetAnn, provides services to users at an application level, services specified in user part of SIP Request URI, control between AS and MS
- Provide IVR and advanced (compared to NetAnn) conference service, as well as fax
- Command oriented, request/response protocol

Basic concept

- There are three type of MSCML message, request, response, notification

```
<?xml version="1.0" encoding="utf-8"?>
  <MediaServerControl version="1.0">
    <request>
      ... request body ...
    </request>
  </MediaServerControl>
```

```
<?xml version="1.0" encoding="utf-8"?>
  <MediaServerControl version="1.0">
    <response>
      ... response body ...
    </response>
  </MediaServerControl>
```

- MSCML messages are located in the body of SIP Request messages. Each SIP request can only embed on MSCML message
  - SIP request messages: INVITE, INFO
  - ‘conf’ and ‘ivr’ in SIP request URI specify the message type

MSCML main commands

- Main requests
  - Conference related
    - <configure_conf>
    - <configure_leg>
    - <configure_team>
  - IVR related
    - <play>
    - <playcollect>
    - <prompt>
    - <playrecord>
    - <stop>
  - Event/signal (within a dialog)
    - <subscribe>
    - <notification>
    - <signal>
- Response
  - ID: optional
  - Request Type: e.g. <play>
  - Code: 2XX, 4XX, 5XX
  - Text: human readable

MSCML conference management

- Configure_conference is mandatory: creating a control leg for conference

```
<?xml version="1.0" encoding="utf-8"?>
  <MediaServerControl version="1.0">
    <request>
      <configure_conference reservedtalkers="120"
        reserveconfmedia="yes"/>
    </request>
  </MediaServerControl>
```

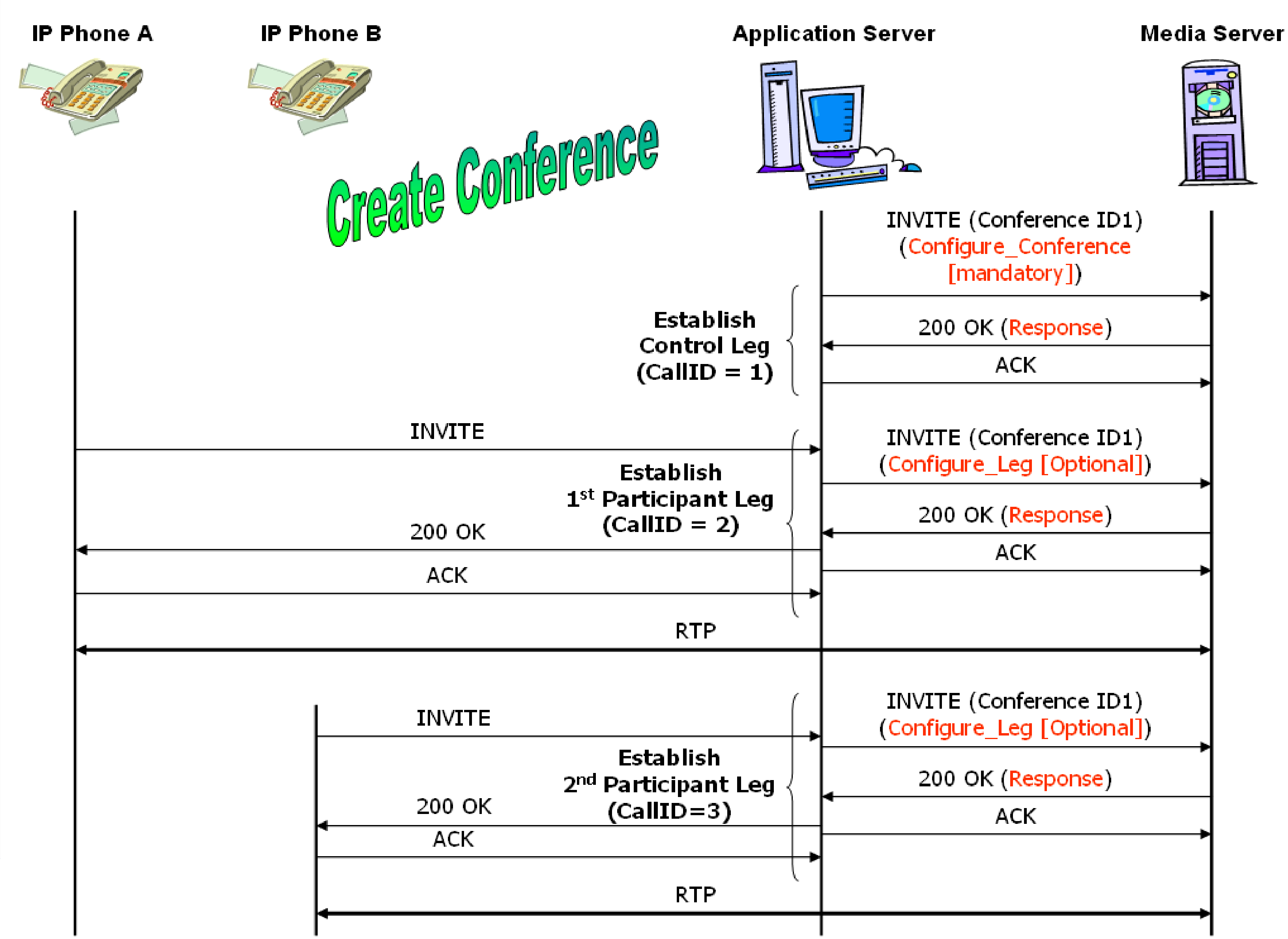
- Configure_leg is a control leg for a dialog. It can configure the dialog’s media mode

```
<?xml version="1.0" encoding="utf-8"?>
  <MediaServerControl version="1.0">
    <request>
      <configure_leg mixmode="mute"/>
    </request>
  </MediaServerControl>
```

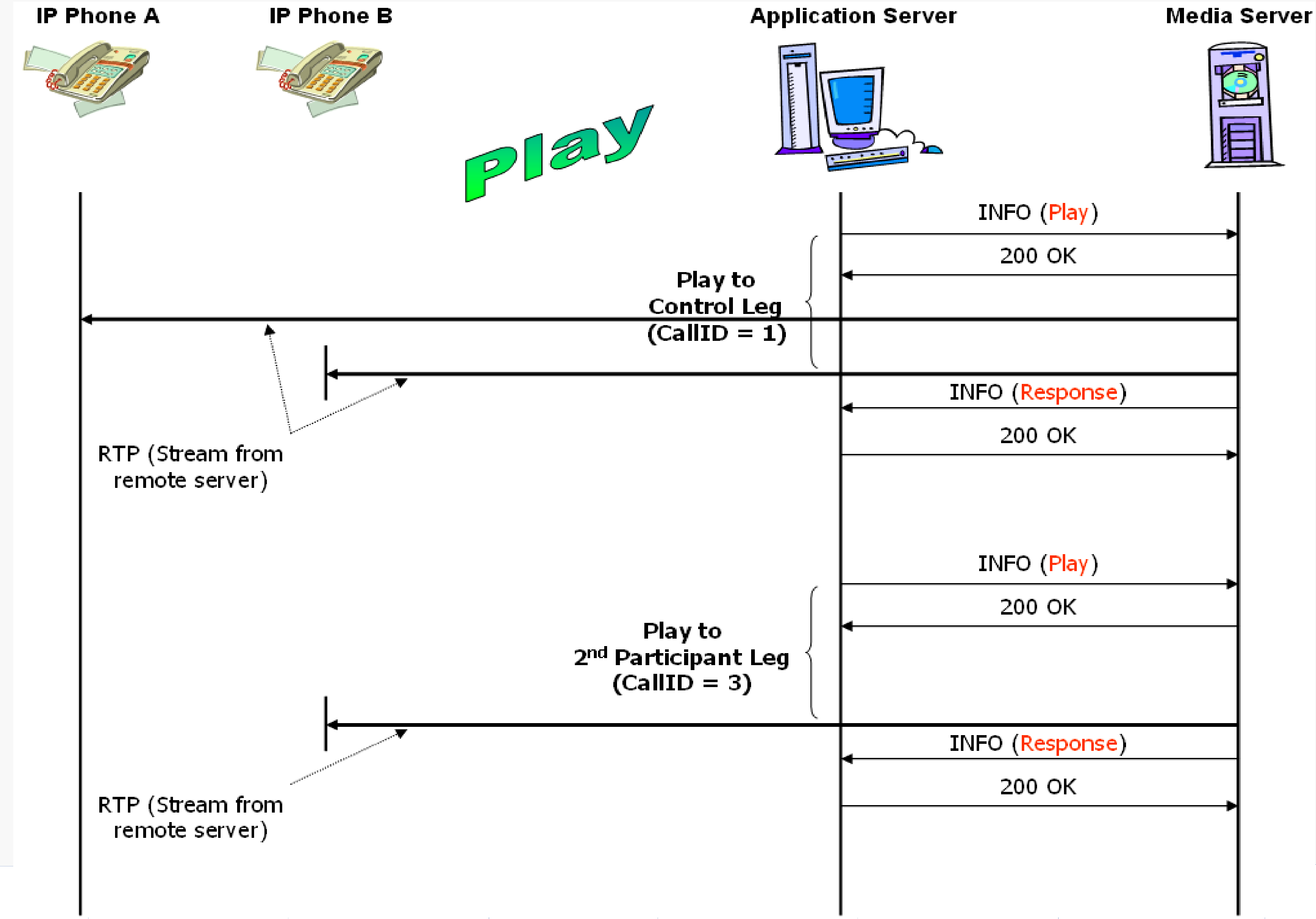
- Can play a prompt to a conference or to a specific leg
- Conference terminates by sending a BYE to conference control leg
- BYE to a leg will just remove a participant

```
<?xml version="1.0" encoding="utf-8"?>
  <MediaServerControl version="1.0">
    <request>
      <play>
        <prompt>
          <audio
url="http://prompts.example.net/en_US/welcome.au"/>
        </prompt>
      </play>
    </request>
  </MediaServerControl>
```

MSCML conference example – Create



MSCML conference example – play a prompt



Megaco vs. SIP based media control

	Megaco	SIP based media control protocols	
Protocols/ standards	MGCP, H.248	RFC 4240 (NetAnn)	RFC 4722 (MSCML)
mode	Master/slave: A MG controlled by one MGC a time	Peer-to-peer: MS controlled by many application at the same time	
organization	ITU-T (applied by 3GPP)	IETF (applied by 3GPP IMS conferencing architecture)	
focus	Any media service	Simple conference, announcement, prompt	Conference, IVR
Develop and Deployment	Complex and Need time	Easy and can be fast, but may have more interoperability issues	
code	Binary/text	Text	Text (XML)
Interested by	Telecom	Internet and Telecom	
Status	comprehensive	Under development	

References

- SIP:
  - *Henning Schulzrinne and Jonathan Rosenberg*, "The Session Initiation Protocol: Internet-Centric Signaling" IEEE Communications Magazine, Oct 2000
  - RFC 3261
  - SIP Conference model: RFC 4353
- H.323:
  - *Hong Liu and Mouchtaris, P. , "Voice over IP signaling: H.323 and beyond"* , IEEE Communications Magazine, Oct 2000
  - *Markku Korpi and Vineet Kumar,* "Supplementary Services in the H.323 IP Telephony Network", IEEE Communications Magazine, July 1999
- Megaco:
  - *Tom Taylor, "*Megaco/H.248: A New Standard for Media Gateway Control*",* , IEEE Communications Magazine, Oct 2000
- NetAnn: RFC 4240
- MSCML: RFC 5022

Questions