Applying Real-time Multimedia Conferencing Techniques to the Web

Mark Handley

UCL and ISI

History

- There are RFCs on live audio on the internet dating back to 1973, and two IETF working groups in this area (AVT and MMUSIC).
- In 1992, for the first time, live audio from the IETF meeting was multicast internationally across the Internet
- Since then, the Mbone has grown to ~3700 subnetworks on 5 continents.

Standards
(Audio Video Transport)

- The *Real-time Transport Protocol* (RTP) is currently at IETF Proposed Standard stage after 3 years of development.
- RTP payload formats are defined for many standard encodings and some experimental ones.
- Work is in progress on defining combined IP/UDP/RTP header compression for slow links.

Internet Drafts
(IETF Multiparty Multimedia Session Control WG)

SDP - Session Decription Protocol

SAP - Session Announcement Protocol

SIP - Session Invitation Protocol

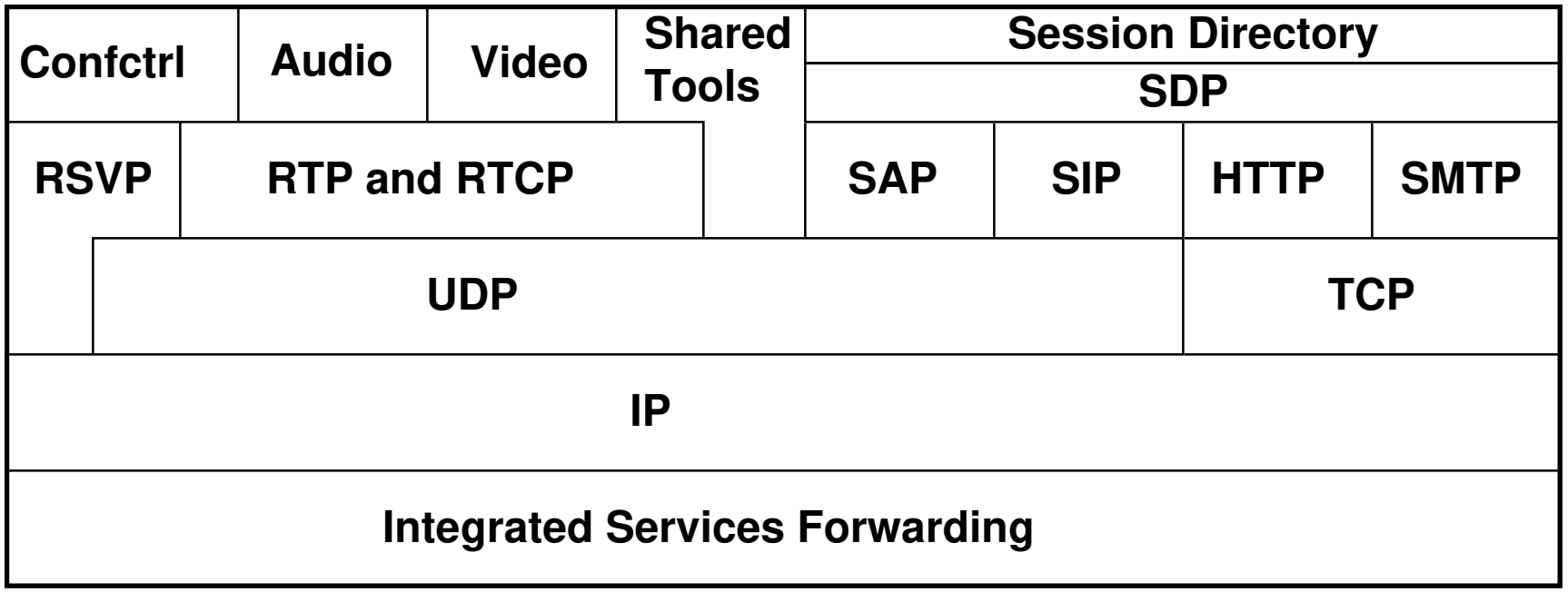
SCIP - Simple Conference Invitation Protocol

SCCP - Simple Conference Control Protocol

all these are available from

ftp://ftp.isi.edu/confctrl/docs/

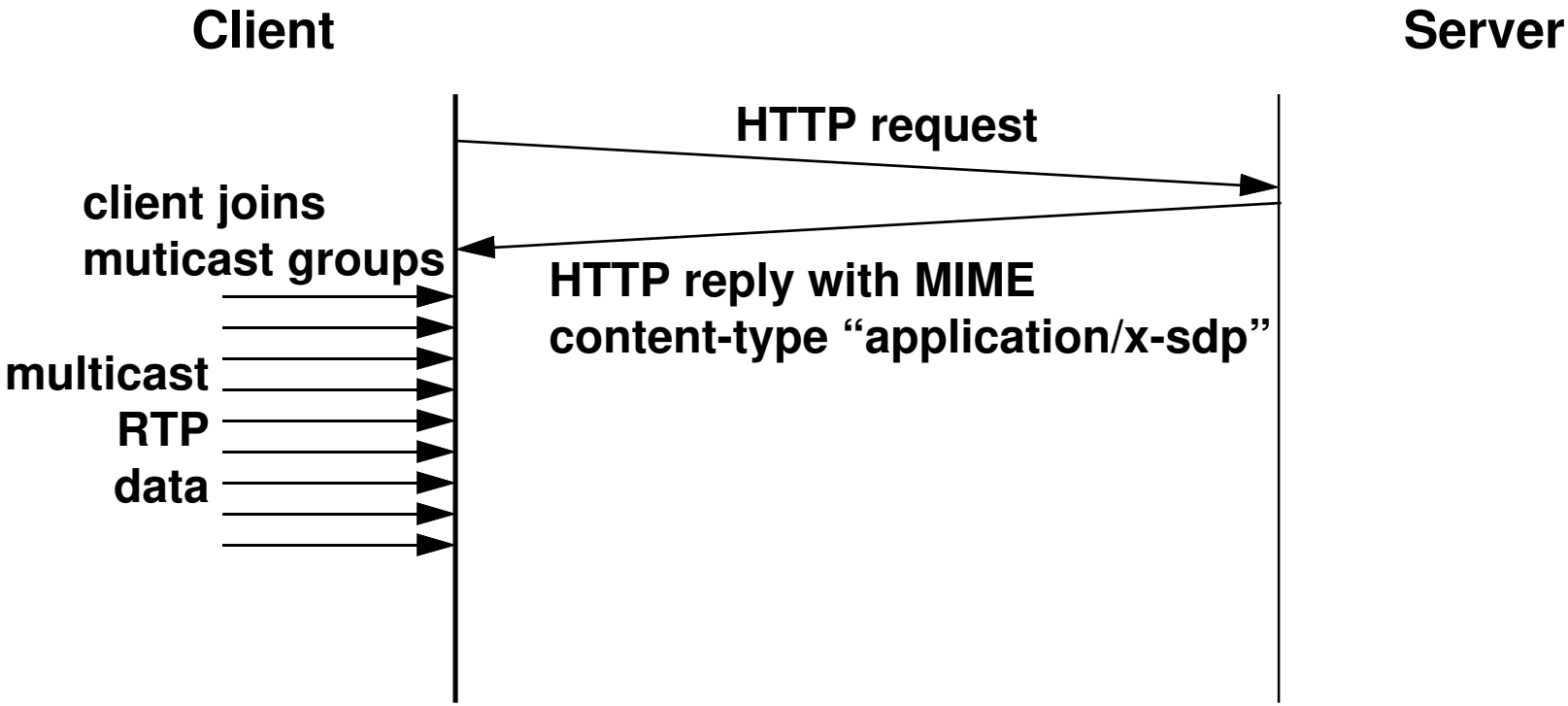
The Multimedia Conferencing Stack



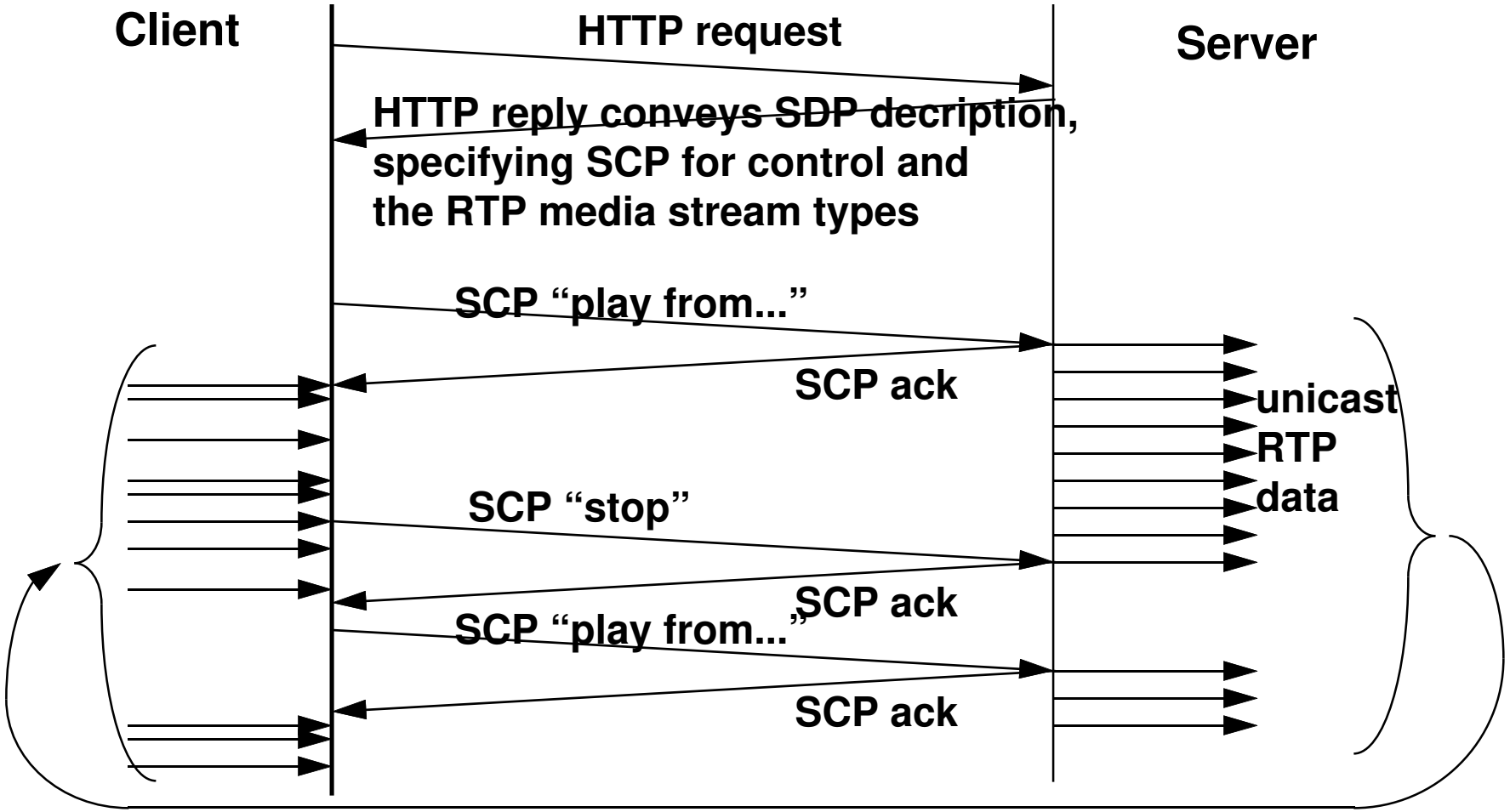
Applying this to the Web

- For some purposes, we already have sufficent protocols defined to integrate real-time media with the web.
- The main missing part is a control protocol for controlling multimedia servers (stop, play, rewind, record, etc)
- We also need some HTML extensions to integrate these media steams into documents.

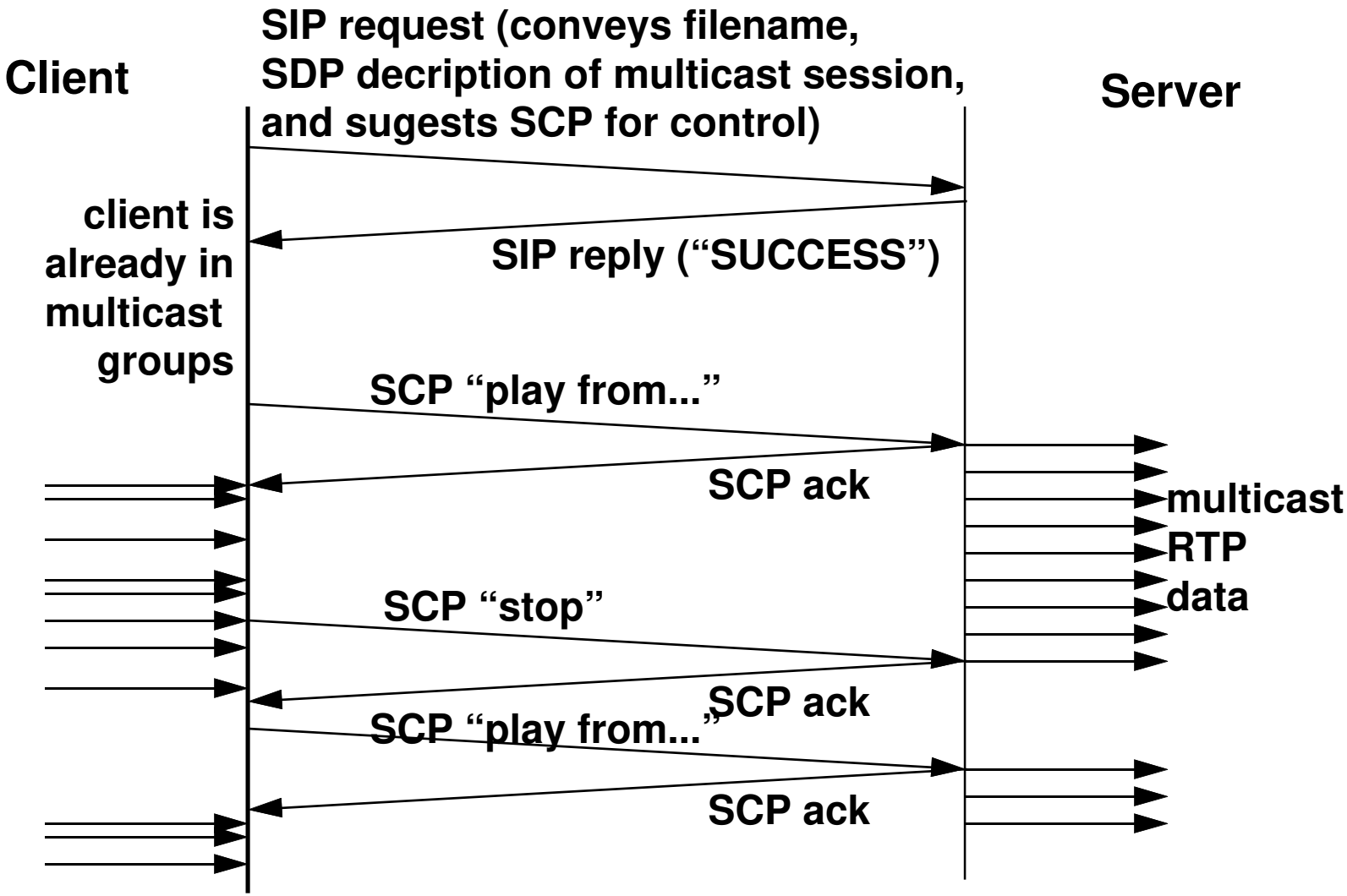
Joining a live Multimedia Session from the Web



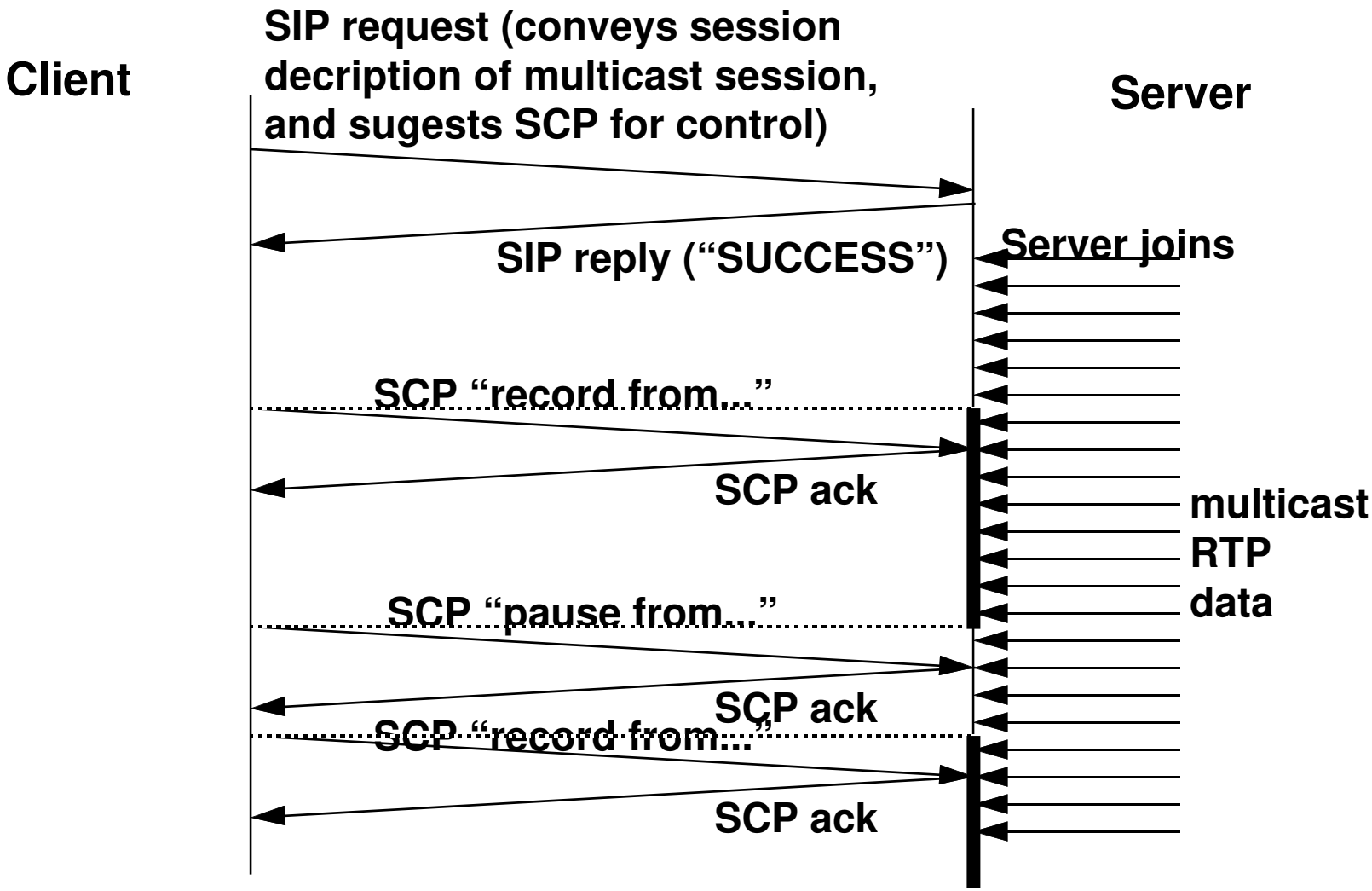
Playback of Recorded Session to a Single Web Client



Playback of Recorded Session into an existing Multicast Session



Recording of a Multicast Session



Embedding Media Streams in HTML

Disclaimer - don't worry about the precise syntax, it's the general idea I'm trying to get across!!!

Embedding Media Streams in HTML

```
<HR>
<AUDIO href=http://www.x.uk/sess17.sdp#audio>
Take a look at this cool web seminar:<BR>
<VIDEO href=http://www.x.uk/sess17.sdp#video><BR>
<WHITEBOARD
  href=http://www.x.uk/sess17.sdp#whiteboard>
<HR>
```

General idea:

- **One *multi*-media URL referenced by several embedded media.**
- **Synchronised playback of several different media streams.**

Embedded Decoders (II)

```
<TABLE> <TR> <TD>
  Mark Handley <BR>
  <VIDEO
    href=http://www/sess3.sdp#video/mjh@128.16.64.4>
</TD> <TD>
  Jon Crowcroft <BR>
  <VIDEO
     href=http://www/sess3.sdp#video/jon@128.16.64.16>
</TD> </TABLE>
```

- There may be multiple video streams coming from a single multimedia session (unicast or multicast)
- Need to de-multiplex by RTP CNAME...

Bad Ideas

- **Using TCP for everything**
- **Using TCP for anything :–)**
  - **especially real-time or real-time control streams**
- **Multicast URLs**
- **Multimedia URLs**
- **Monomedia URLs**
  - **for all these, you need an extra level of indirection such as provided by MIME or SDP.**
- **Bundling different media into a single data stream.**
  - **you lose the ability for the network and the OS to treat some streams preferentially.**
  - **you lose modularity of encoders/decoders.**

Good Ideas

- RTP
- IP/UDP/RTP header compression
- Layered encodings
- Piggybacked lower quality redundancy
- Timeslice striping across packets, and receiver interpolation
- Multicast
- Caches

Problems

- **Many**
- **Especially congestion control with realtime streams.**
  - **Layered encodings may help significantly**
- **Caching realtime streams**

Conclusions

RTP

Everything else is “work-in-progress”, so not concluded!