Practical speak subscription handling in multicast conferences

Alessandro Falaschi

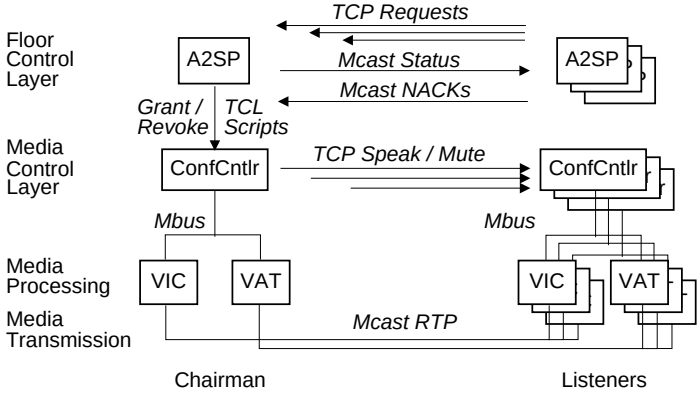
Dept. Info-Com, University of Roma "La Sapienza" - Via Eudossiana 18, 00184 Roma - Italy

aalef@infocom.uniroma1.it - Tel. +39 06 44585482 Fax + 39 06 4873300

Background Despite of the ease of announce and join loosely-coupled multicast conferences, their widespread adoption is hindered probably also because a simple mechanism for speaker turn over, analogous to the deeply-rooted social behavior of raising an hand, being noted by the chair, and then invited to speak, is missing. Several proposal have been made [1, 2, 4, 3] about the proper way to implement conference control protocols, often referred to as the *floor control* issue, and applications have been produced [6, 8, 5, 7], even comprising evolute features such as resource reservation. However, a simple solution to the problem of conference chairing is still missing. Actually, something [9] seems to move again in the SIP context, and sounds promising. In the meanwhile, existing solutions use either H.323 (that is unicast) technology [10], or are bounded to some particular scope [11].

Practical Solution We have chosen to develop a new component (*Ask to Speak*, or A2SP for short) which performs the *floor control* function of a multicast conference, and relies on existing tools (VIC [12] and VAT [13] for media *processing*, CONFCNTLR [14] for media *control*, SDR [12] for session *announcement*), to arrange the whole architecture.

A2SP collects requests to speak, and disseminates (by multicast) *status information* such as the enqueued candidate speakers list, and the speaker identity. These informations are published by an SRM-like [16] protocol, achieving consistency among participants. Granting of the floor is delegated to CNFCNTLR, which performs *remote control* of the media processing tools, and activates the participant's transmission.



The use of A2SP in a conference is announced by SDR, which is enriched by specific plug-in files, which either specify the existence of an A2SP control type, or specify how to launch media tools when an A2SP attribute if present in the SDP received by SDR. In such a way, if VIC/VAT are to be controlled by A2SP, then an ad hoc TCL script is loaded at their start, which redefine some procedures, e.g. disabling the transmit buttons at the listener side.

Collection of requests By joining to the conference, participants start VIC, VAT and A2SP, and the latter launches CONFCNTLR. After that, A2SP checks if it is running on the same host which originated the SDP, in this way deducting whether it is chair or participant, and differentiates its behavior accordingly. Listeners so know where to locate the chair, and use TCP connections for sending him *requests* to speak, summarizing the question subject. The chairman may approve the requests, or reject them.

Dissemination of the status Approved requests build up in a *shared state* of enqueued requests to speak, which is sent by multicast to the whole listeners group in incremental steps. Participants can refrain from requiring to speak, if anybody has already submitted the same matter. The shared state messages leave the chair as a semi-periodical *Heart-Beat* (HB) sequence, time-stamped by an incrementing sequence number. If no information has to be sent, the HB rate decays, and messages only contains the sequence number of the last transmitted status message, allowing listeners to detect missing packet conditions. In this latter case, listeners try to issue a multicast retransmission request (NACK), according to the SRM [16] paradigm: the NACK is delayed by a random amount of time, and it is suppressed if someone else makes the same request, thus avoiding implosion phenomena. At NACK reception, the chair retransmits the missing data in the next HB message, synchronizing participants, and allowing for late joiners.

Granting the floor At some time, the chair begins to grant requests from participants, one by one. He sends (by multicast) a speaker name, picked up from the authors shown in the approved requests list, which *highlights* on the A2SP window at listener side. The chair A2SP then *sends* a TCL script to its instance of CONFCNTLR, asking to *switch on* the media tools at the remote speaker side, by means of MBUS [15] commands. After that a participant has spoken, the chair sends a new multicast message which *cancels* the speaker name from the A2SP listener windows, and asks CONFCNTLR to *switch off* the remote media tools.

Usability The main speaker of the conference may operate from the same chairman computer, or (probably) can be located at a different place. The chair A2SP can ask CONFCNTLR to activate media transmission also for hosts which do not show in the approved requests list, thus providing the flexibility of granting the floor to the main speaker, as well as to other people. Finally, people who asked to speak can retract their requests, before the chair gives them voice.

Experimentation Usage of the A2SP tool is being experimented on the GARR Italian research network, and is proposed for use in the context of Teledoctorate courses held within the CNIT network [17].

Extensions As the experimentation progresses, new features will be added to A2SP. A TO DO list could be:

- a voting feature, letting listeners to anonymously express their degree of understanding;
- allowance for more that one speaker at time;
- building of an RTCP cache at the chair, to be used for indicating the name of people invited to speak, without a previous request by them;
- auto pilot feature, for granting fixed time to all the enqueued requests in sequence;
- thorough testing against different releases and derivations of the media tools.

References

[1] Schooler, E.M., "Case Study: Multimedia Conference Control in a Packet-switched Teleconferencing System", Journal of Internetworking: Research and Experience, 4(2), pp 99-120, June 1993

[2] Dommel, H.P., Garcia-Luna-Aceves, J.J, "Floor Control for Multimedia Conferencing and Collaboration", Multimedia systems (ACM/Springer), Vol. 5, N0. 1, January 1997

[3] Bormann, C., Kutscher, D., Ott, J., Trossen, D., "Simple Conference Control Protocol – Service Specification", IETF Draft, February 2001

[4] Handley, M., Wakeman, I., Crowcroft, J., " The Conference Control Channel Protocol (CCCP): A Scalable Base for Building Conference Control Applications", ACM SIGCOMM 95, New York, August 1995

[5] Malpani, R., Rowe, L.A., "Floor Control for Large-Scale Mbone Seminars", ACM Multimedia 97, Seattle, USA, November 1997

[6] Orvalho, J., Andrade, T., Boavida, F., "A Conference Control Service Based on the Corba Event Service", 2^{nd} Conf. on Telecomm., Sesimbra, Portugal, 15-16 April, 1999

[7] Gdm-Focus, "Multimedia Internet Terminal (MInT)", http://www.fokus.gmd.de/step/mint/content.html, 1998-06-04 13:35:40 MET DST

[8] ConfMan, "http://www.rvs.uni-hannover.de/products/confman/v2.0/", Lehrgebiet Rechnernetze und Verteilte Systeme, Universität Hannover, 7-Jun-1999, Germany

[9] Wu/Koskelainen/Schulzrinne/Chen, "Use of SIP and SOAP for Conference Floor Control", IETF Draft, November 4, 2002, work in progress

[10] http://www.vrvs.org/

[11] http://www-fp.mcs.anl.gov/fl/accessgrid/

[12] http://www-mice.cs.ucl.ac.uk/multimedia/software/

[13] http://www-nrg.ee.lbl.gov/vat/

[14] Agarwal, D., Perry, M., "ConfCntlr", http://www-itg.lbl.gov/mbone/confcntlr/, Lawrence Berkeley National Laboratory, October 20, 2000

[15] Ott, J., Perkins, C., Kutscher, D., "A Message Bus for Local Coordination", IETF RFC3259, April 2002

[16] Floyd, S., Jacobson, V., Liu, C.G, McCanne, S., Zhang, L., "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing", IEEE/ACM Transactions on Networking, November 1996

[17] http://www.cnit.it