An actual implementation of a distributed multimedia conferencing framework compliant with the IP Multimedia Core Network Subsystem specification.

An extension to a standard centralized conferencing framework under definition inside the IETF XCON (Centralized Conferencing) working group.

DCON is built as an overlay network acting as glue among a number of centralized conferencing "islands"
DCON [1] is an IMS-compliant architecture offering a distributed conferencing service with enhanced functionality, such as conference scheduling and moderation. It exploits existing achievements in the field of conferencing. We started from the IETF Centralized Conferencing (XCON) framework. We developed an open source XCON implementation which has been called CONFIANCE, standing for CONFerencing IMS-enabled Architecture for Next-generation Communication Experience [2]. Our architecture effectively supports the creation and management of a distributed conference in a scenario involving a number of IMS-compliant core networks, interconnected through a communication channel created on an ad hoc basis.

**DCON Implementation**

The figure depicts the main implementation choices of the DCON architecture. The lower box of the picture presents the logical view of the server side, integrating an Asterisk based implementation of the XCON focus (left-hand side) with a brand new module specifically conceived for the SPreAding of Conference Events (which we call SPACE). SPACE is realized as a plug-in for Wildfire, a popular open source instant messaging server. SPACE actually represents a key component of the architecture, since it enables inter-focus communication through the exchange of conference information. Inside DCON, communication between the legacy Confiace modules and the newly created distribution components occurs on the basis of an asynchronous paradigm in which a number of events are generated by Confiace modules whenever something relevant occurs in the XCON island they currently supervise. The upper box presents a view of the client side that logically can be viewed as a single, integrated entity capable of interacting with the framework by means of either SIP or instant messaging. The SIP part of the client was realized through the open source Minisip softphone, appropriately modified to support both the BFCP and the CCP protocols. For the IM client, we chose to adopt Spark, an open source cross-platform client using the XMPP protocol. We added the capability to interact with the DCON platform through an ad hoc created plug-in (SpaceClient in the figure) to Spark.
