A RELOAD Usage for
Distributed Conference Control (DisCo)

draft-knauf-p2psip-disco-00

Alexander Knauf
Gabriel Hege
Thomas Schmidt
Matthias Wählisch

alexander.knauf@haw-hamburg.de, hege@fhtw-berlin.de, \{t.schmidt,waehlisch\}@ieee.org
Outline

1. Problem statement and objectives

2. Distributing a conference focus with SIP

3. Publishing a distributed conference in RELOAD

4. Proximity-aware focus selection
Problem Statement for Conferences in P2PSIP Scenarios

• A conference in the tightly coupled model is managed by a *single* entity called *focus* in SIP:
  – Maintains signaling and media parameter negotiation
  – May perform media mixing functions

• **Problem (1):** The Conference URI
  – *Identifies* the multiparty session, and
  – *locates* the conference focus
    ▪ Single point of failure

• **Problem (2):** No dedicated server architecture in P2PSIP
  – Media mixing performed at the end-user devices
    ▪ Scaling problem within large conferences
  – Conference must be registered and globally accessible
    ▪ Demands a registrar, e.g., available through DNS
Objectives of Distributed Conference Control

• *Separate* the logical conference ID from the controlling entities:
  – Allows *multiple* focus peers to manage a single conference
  – Increases robustness against focus failures

• **RELOAD Usage** for Distributed Conference Control:
  – Conference URI is registered as a *key* for several focus peers that are responsible for conference control
Distributing a focus with SIP

- **First Step:** Transparent distribution of the conference focus
  - Participants in role of *focus peers* are responsible for a subset of conference members
  - Signaling messages sent from *several* focus peers appear as originating from *one ‘virtual’* conference focus
    - Routing decision based on an additional *Record-Route* header pointing to the responsible focus peer
      
      INVITE sip:bob@dht.example.com SIP/2.0
      Call-ID: 0815@141.22.26.55
      CSeq: 1 INVITE
      From: <sip:conference@dht.example.com>;tag=134652
      To: <sip:bob@dht.example.com>;tag=643684
      ...
      Contact: <sip:conference@dht.example.com>;isfocus
      Record-Route: <sip:alice@dht.example.com>
      ...

- Alice receives message through the *Record-Route* and – as responsible focus peer - intercepts message from Bob

Here: Alice is the focus responsible for Bob
Operations in a Distributed Conference

• **Second Step:** Definition of protocol schemes for
  - *State synchronization:* Achieved by *conference event package* [RFC4575] extended by elements describing a focus peer’s local state
    ▪ Focus peers get consistent and global view of conference state
  - *Call delegation:* Transfer calls using SIP REFER requests carrying session identifier (for semantic recognition of calls)
    ▪ Used in cases of overloading, leaves or failures of focus peers
  - *Focus Discovery:* Allocating new focus peers that support the conference
    ▪ Enables load distribution
Definition of a Distributed Conferencing (DisCo) Kind

• **DisCo-Registration** stores a dictionary of:
  
  – **Address-of-Records or Node-IDs** of focus peers
  
  – A *coordinates vector* describing the focus’ relative network position

• DisCo-Registration is a *shared* resource of all focus peers

```c
enum {
    sip_focus_uri (1),
    sip_focus_node_id (2), (255)
} DisCoRegistrationType;

struct {
    opaque coordinate<0..2^16-1>

    select (DisCoRegistrationType.type) {
        case sip_focus_uri:
            opaque uri<0..2^16-1>

        case sip_focus_node_id:
            Destination destination_list<0..2^16-1>

    /* This type can be extended */

} DisCoRegistrationData;

struct {
    DisCoRegistrationType type;
    uint16 length;
    DisCoRegistrationData data;
} DisCoRegistration;
```
Creating a Conference

1) Probe on existence of Conference URI
   - StatReq is sent to storing peer for duplicate addresses detection

2) Request a new certificate that is used for the DisCo-Registration
   - Certificate for the “virtual” conference user

3) Store mapping Conf-ID to <creating peer, coordinates value> at storing peer

<table>
<thead>
<tr>
<th>Enroll.Serv</th>
<th>Alice</th>
<th>Peer1</th>
<th>Overlay</th>
<th>PeerN</th>
<th>StoringPeer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>StatReq Res:Conf-URI</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>----------------------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>StatAns</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&lt;-&lt;-------------------</td>
<td>&lt;------</td>
<td>-------------</td>
</tr>
<tr>
<td>&lt;=Cert====</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>===Cert==&gt;</td>
<td></td>
<td>StoreReq Res:Conf-URI Kinds:DisCo[,SIP]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>StoreAns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Joining a Conference and publishing Focus-ability

1) Resolve Conf-ID by RELOAD *fetch* request
   - Answer contains available focus peers
2) Select closest focus
3) Establish transport connection by *AppAttach* request to Alice
4) ICE-Checks
5) Create SIP dialog using the existing transport
6) Alice passes writing permission for the DisCo-Registration to Bob
7) Bob may stores its mapping and becomes a potential focus peer
Thanks for your attention!

Questions?

Alexander Knauf, Gabriel Hege, Thomas Schmidt, Matthias Wählisch
http://inet.cpt.haw-hamburg.de/