A Virtual and Distributed Control Layer with Proximity Awareness for Group Conferencing in P2PSIP

Alexander Knauf[1]
alexander.knauf@haw-hamburg.de

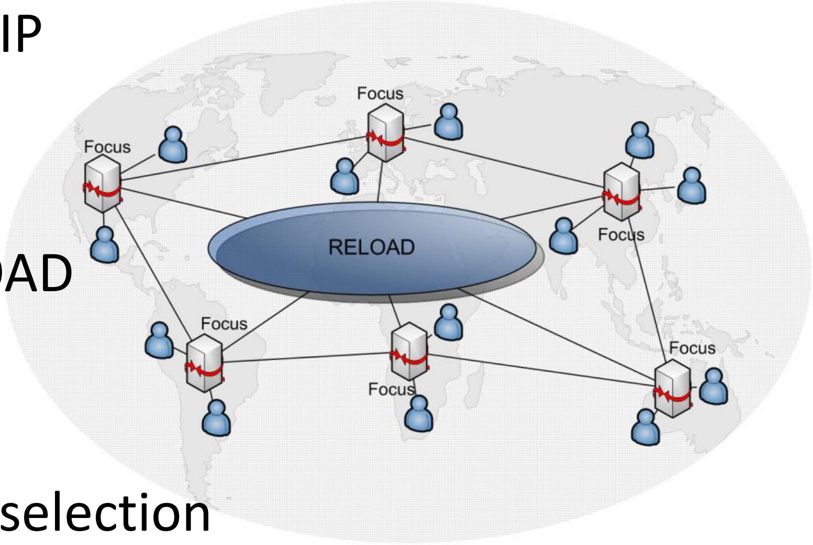
Gabriel Hege[1], Thomas Schmidt[1], Matthias Wählisch[2]

[1]AG INET
Dept. Informatik
HAW Hamburg
Germany

[2]AG CST
Inst. für Informatik
FU Berlin
Germany

Outline

1. Problem statement and objectives
2. Distributing a conference focus with SIP
3. Evaluation (1): Signaling costs
4. Virtualising the Conference ID in RELOAD
5. Proximity-aware focus selection
6. Evaluation (2): Proximity-aware focus selection
7. Conclusion & Outlook

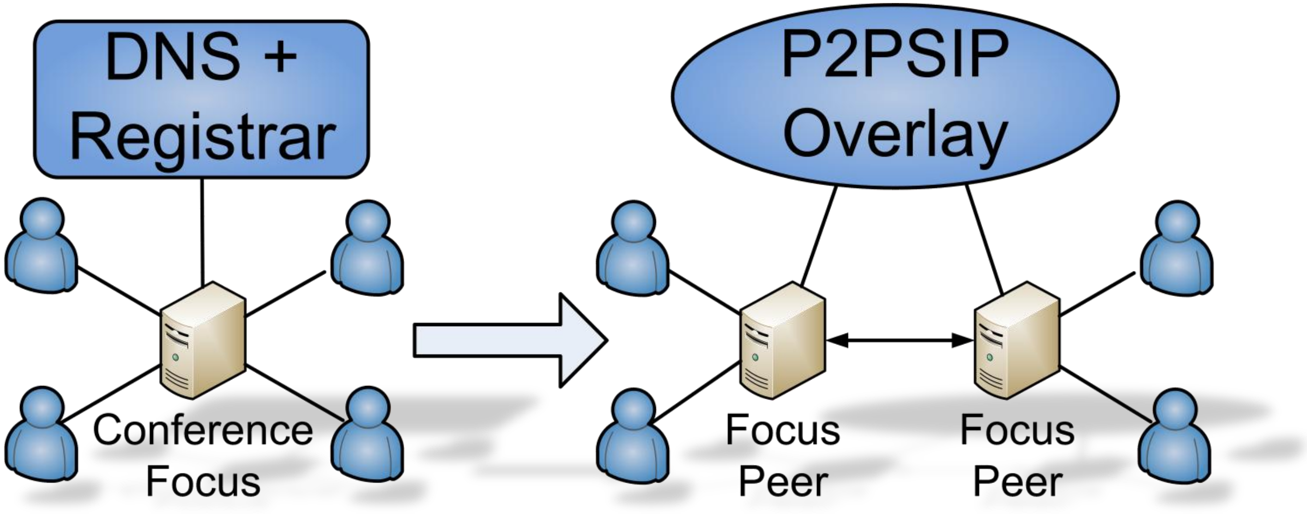


Problem Statement for Conferences in P2PSIP Scenarios

- 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
- *Replace* Registrars and DNS by a P2PSIP Overlay
  - Requires a RELOAD *Usage* for Distributed 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>
...
```

Here: Alice is Bob's responsible focus

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

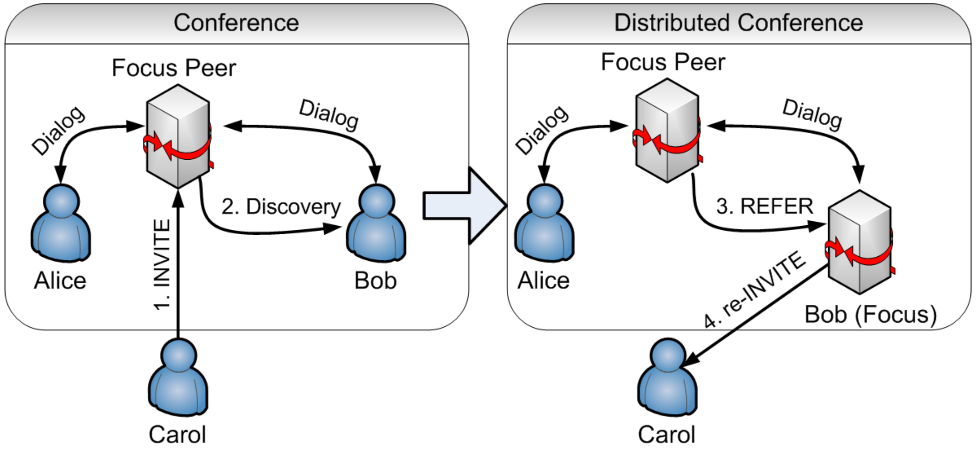
Operations in a Distributed Conference

- *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

Splitting a Focus: Common Scenarios

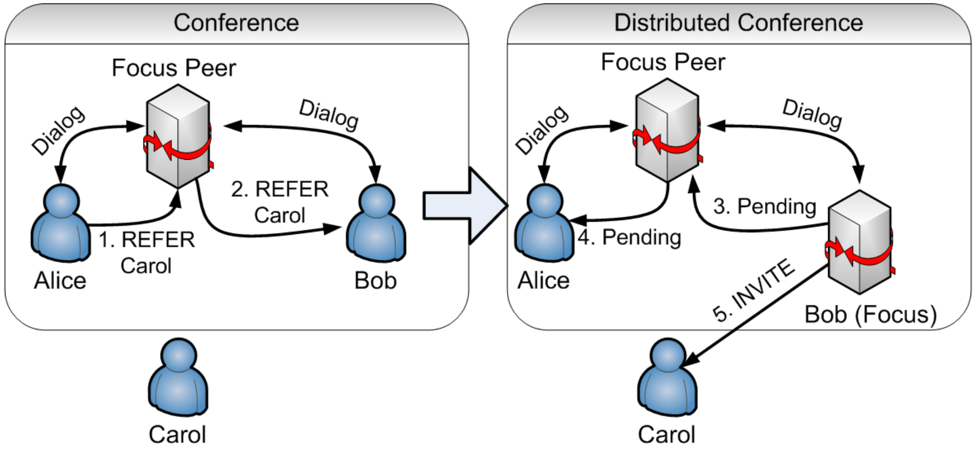
1st party invitation:

- **Situation:** Participation request sent to a single overloaded focus peer
- **Reaction:** Call delegated to other focus peer



3rd party invitation:

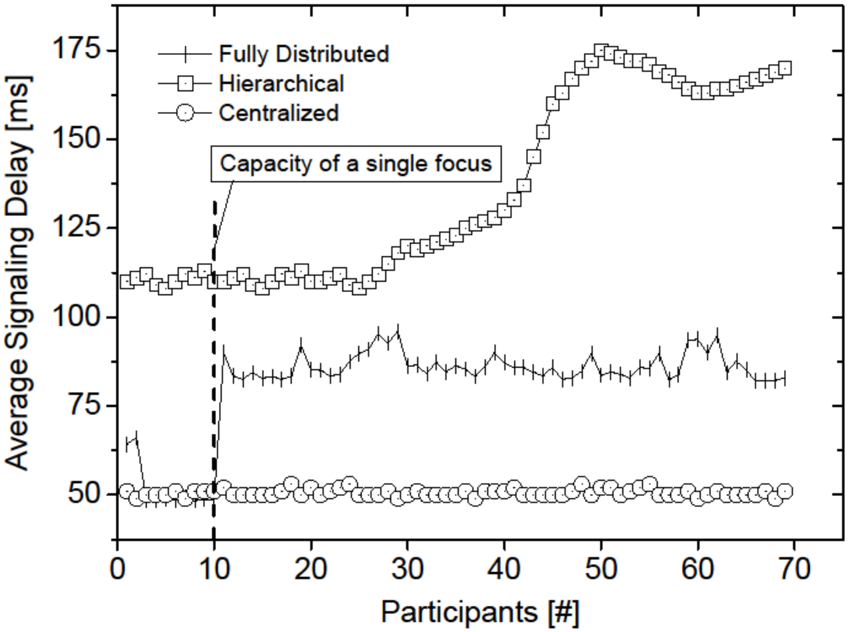
- **Situation:** Participant requests overloaded focus to invite peer
- **Reaction:** *REFER* to underloaded focus peer

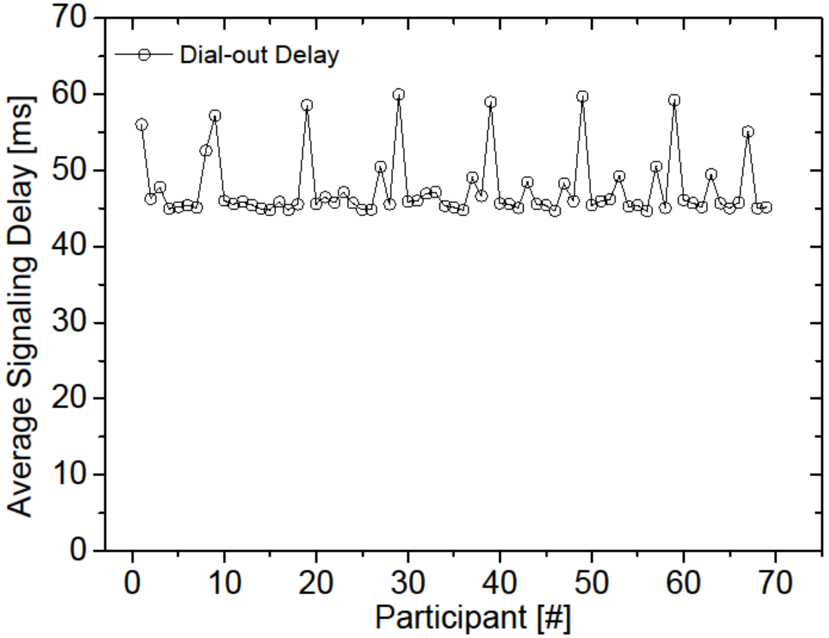


Signaling Costs - Evaluation

- Comparing three signaling schemes:
  - Centralized, hierarchical *[Cho et al, SAC-IEEE '05]* distributed conferencing

- 3rd Party invite signaling delay:
  - Amplitudes caused by call delegations

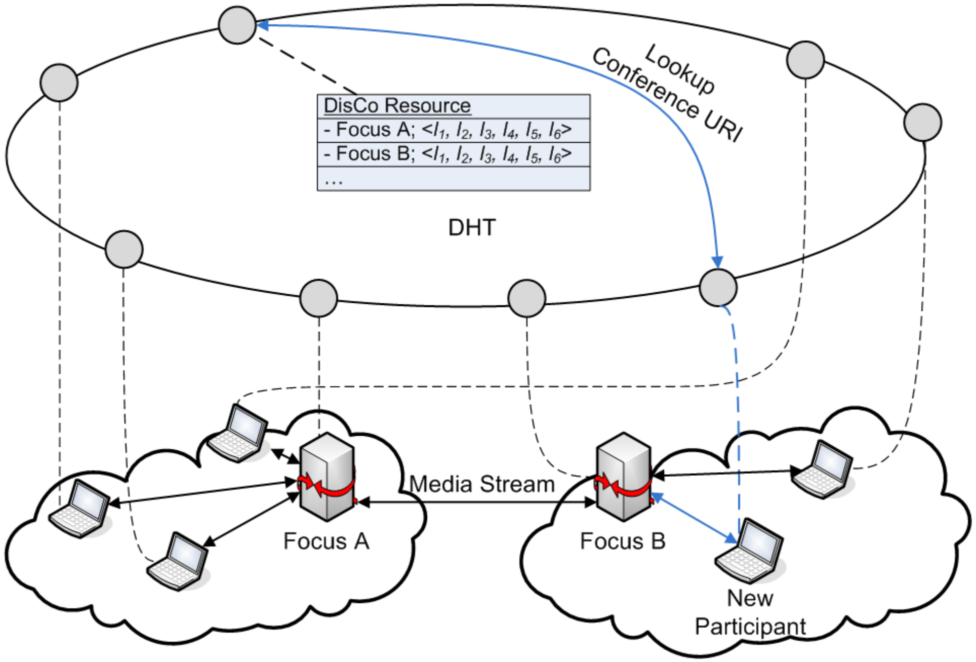




▶ Signaling delay remains *constant* with increasing conference size

Conference ID Virtualization

- **Problem:** How to distribute the conference entry point?
- **Idea:** Conference URI is registered in a *P2PSIP overlay* as a *key* for several focus peers that are responsible for the conference control
    - Achieve independence of dedicated registrar servers
    - Detach the Conf-ID from any physical instance



DisCo Resource
- Focus A; <l_1, l_2, l_3, l_4, l_5, l_6>
- Focus B; <l_1, l_2, l_3, l_4, l_5, l_6>
…

Conf-ID Virtualization – A New Usage for RELOAD

- RELOAD – REsource LOcation And Discovery
  - P2PSIP signaling standard in the IETF (work in progress)
  - Designed to support a *variety of applications*
    - Stored data identified by *Resource-ID* and application specific *Kind-ID*
  - Security based on enrollment server
  - NAT & Firewall traversal through ICE
  - Pluggable overlay algorithms (e.g., Chord)
  - Secure transport connections by TLS/DTLS

Definition of a Distributed Conferencing (DisCo) Kind

- DisCo data structure 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

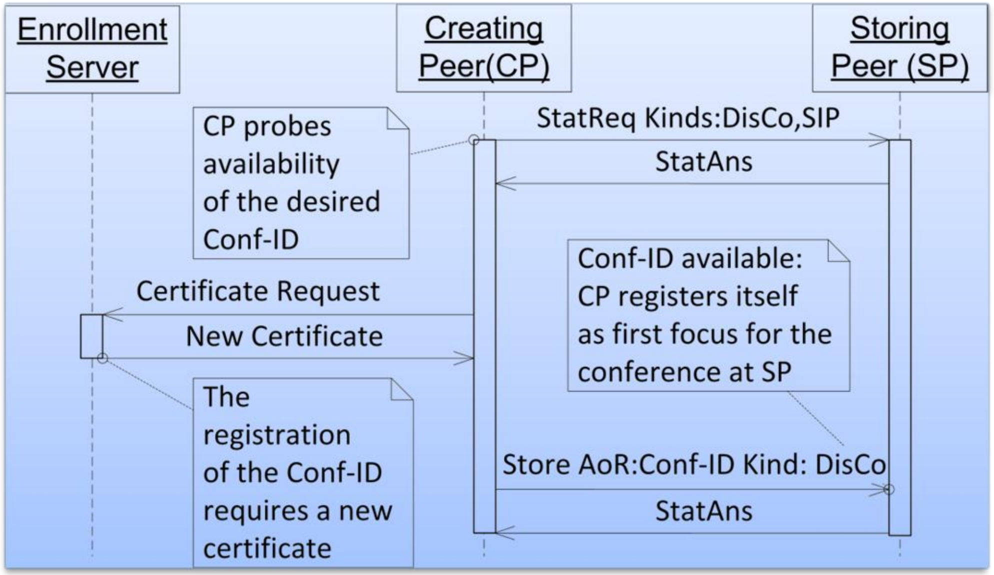
```
enum {sip_focus_uri (1), sip_focus_node_id (2)
} DisCoRegistrationtType;

struct {
  opaque coordinate<0..2^16-1>
    select (DisCoRegistrationtType.type) {
      case sip_focus_uri:
        opaque uri<0..2^16-1>
      case sip_focus_node_id:
        Destination destination_list<0..2^16-1>
    }
} DisCoRegistrationData

struct {
  DisCoRegistrationtType type;
  uint16 length;
  DisCoRegistrationData data;
} DisCoRegistration
```

Graduated Security Model for Shared DisCo-Registrations

- **Task:** Defining access control policies for *shared* DisCo-Registration
- **Solution:** Focus peers pass writing permission to participants based on chosen security model
  - Authentication by ordinary SIP mechanisms while inviting conference
    - Shared secret used to join conference
    - E.g., SIP *Authorization* header field
- Security models:
  - *Open access:*
    - No Authentication
  - *Closed access:*
    - Every Peer must be authenticate itself
  - Optionally for both: *Focus Authenticate*
    - Extra authentication for focus peers

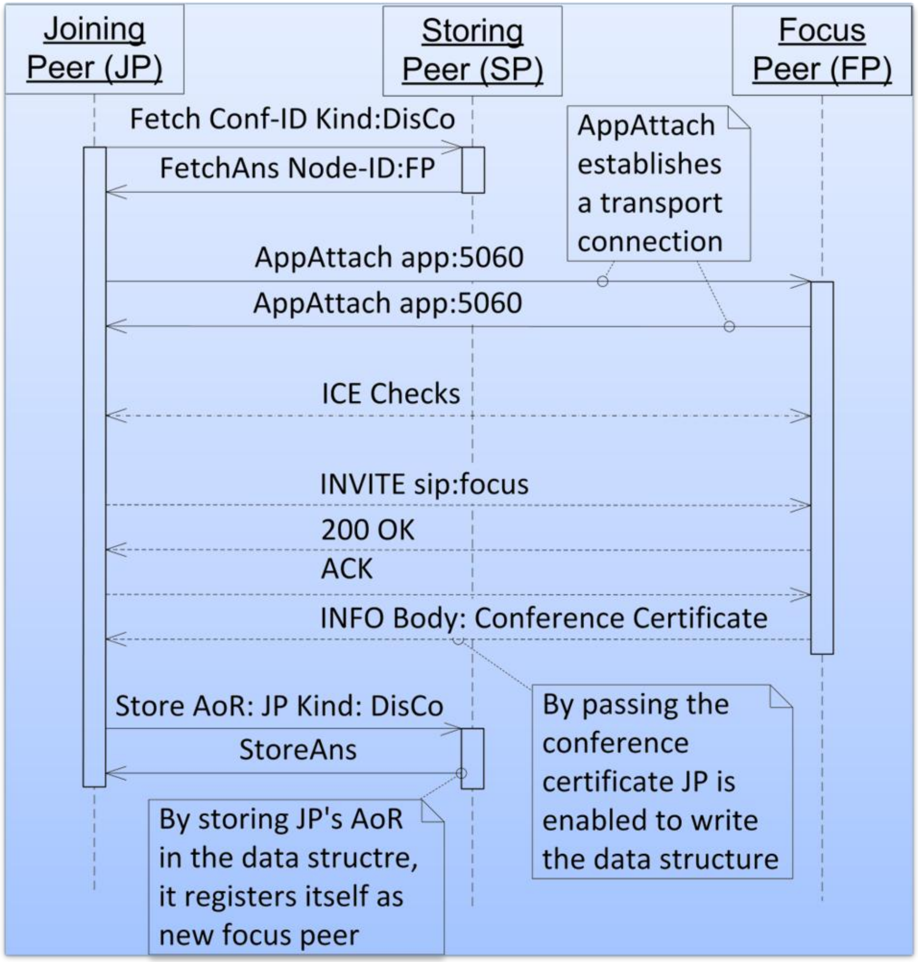
Creating a Conference

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 vector>* at storing peer


Enrollment Server
Creating Peer(CP)
Storing Peer (SP)
CP probes availability of the desired Conf-ID
StatReq Kinds:DisCo,SIP
StatAns
Conf-ID available: CP registers itself as first focus for the conference at SP
Certificate Request
New Certificate
The registration of the Conf-ID requires a new certificate
Store AoR:Conf-ID Kind: DisCo
StatAns

Joining a Conference and publish Focus-ability

1) *Resolve* Conf-ID by *fetch* request routed to storing peer
   - Answer contains available focus peers
2) Select closest focus (*next slides*)
3) Establish transport connection by *AppAttach* request routed to FP
4) ICE-Checks for NAT & Firewall traversal
5) Creating SIP dialog by using the existing transport
6) FP passes writing permission to JP
7) JP stores its mapping and becomes a *potential* focus peer



AppAttach establishes a transport connection

By passing the conference certificate JP is enabled to write the data structure

By storing JP's AoR in the data structre, it registers itself as new focus peer

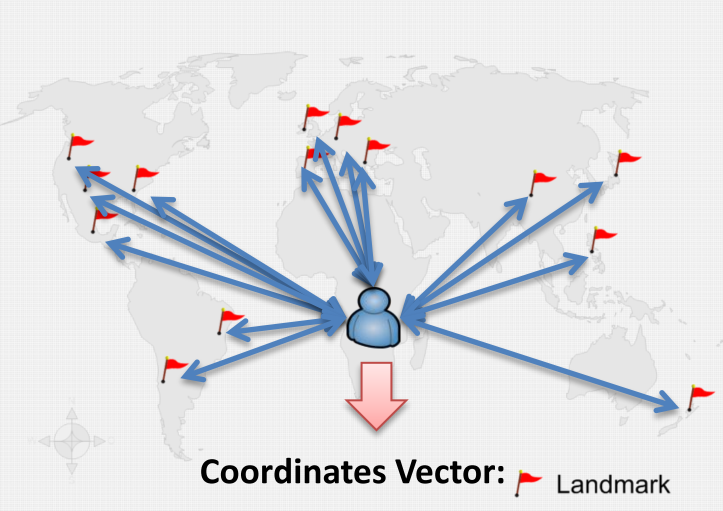
Determination of Topological Descriptors

- Each peer in a distributed conferences determines a *coordinates vector* describing a peer's position in an n-dim Cartesian space

- Distance between two peers p_1, p_2 is Euclidian distance between p_1*'s* and p_2*'s* coordinates vector:

$$d(p_1, p_2) = \sqrt{\sum_{i=1}^{n} (p_{1i} - p_{2i})^2}$$

- New participants select a focus peer whose Euclidian distance in minimal

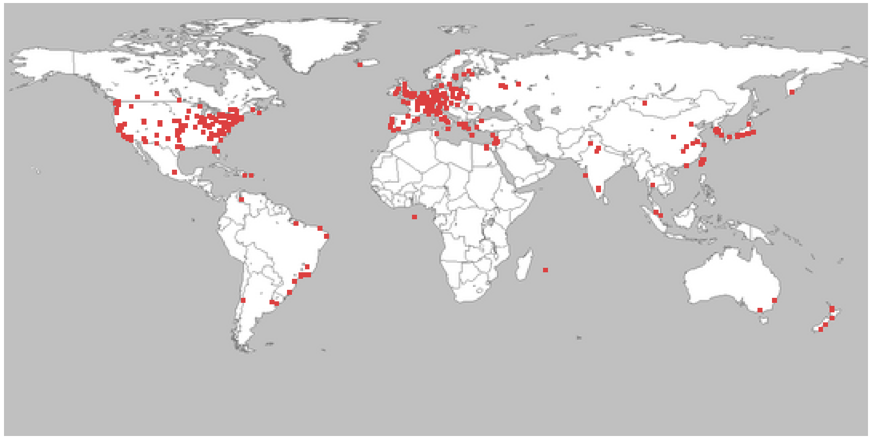


<105ms,120ms,110ms,...,320ms,175ms>

- Demonstrating a landmarking approach for proximity-aware focus selection as in *[Ratnasamy et al, INFOCOM '02]*

Proximity-aware Focus Selection - Evaluation setup

- Evaluation using *PlanetLab* platform:
  - Measurements using about 100 *PlanetLab [www.planet-lab.org, 2010]* nodes:

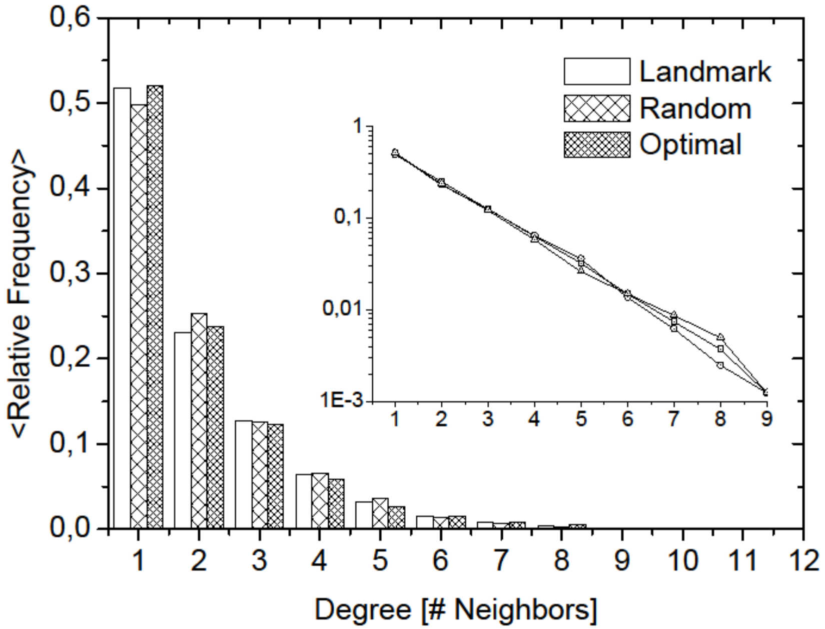


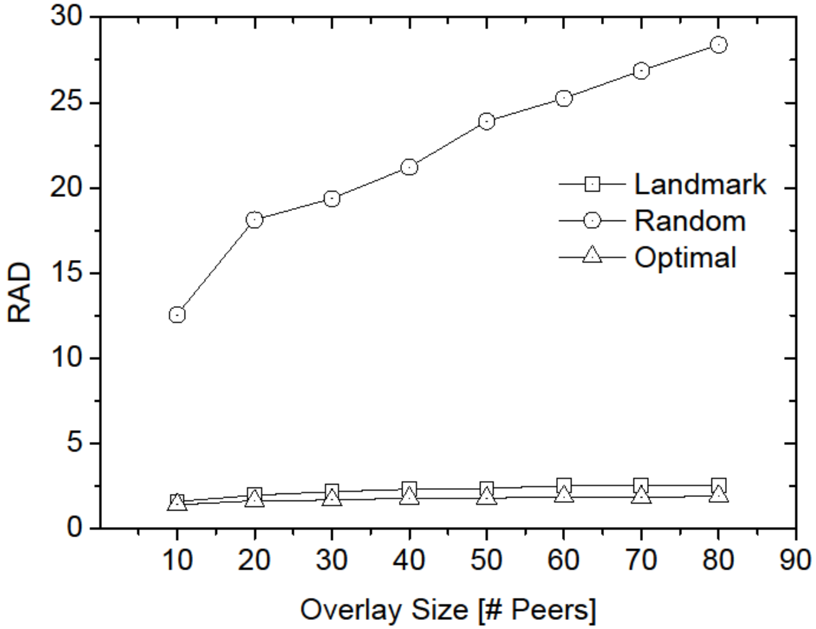
  - 15 *Caida [www.caida.org, 2010]* monitors used as landmark hosts
  - Built multiple times different peer-topologies using:
    1) Random strategy
    2) Optimized strategy
    3) Landmarking strategy

Proximity-aware Focus Selection - Evaluation

- Comparing different peer topology building schemes:

- Peer degree using landmarking vs. optimal and random strategy
- Delay stretch using landmarking vs. optimal and random strategy





- A simple landmarking strategy close to optimal solution

Conclusion & Outlook

- Virtual and Distributed Conferences:
  - Transparent ID/LOC split of the Conf-ID
  - Conference management distributed among several peers
  - State synchronization, call delegation and focus discovery
  - Virtualized conference ID within RELOAD overlay
  - Proximity-aware focus selection

- Outlook:
  - Refine optimization strategies to jointly follow constraints of proximity and load distribution
  - Progress *"A RELOAD Usage for Distributed Conference Control (DisCo)"* at the IETF P2PSIP WG

Questions?

iNET

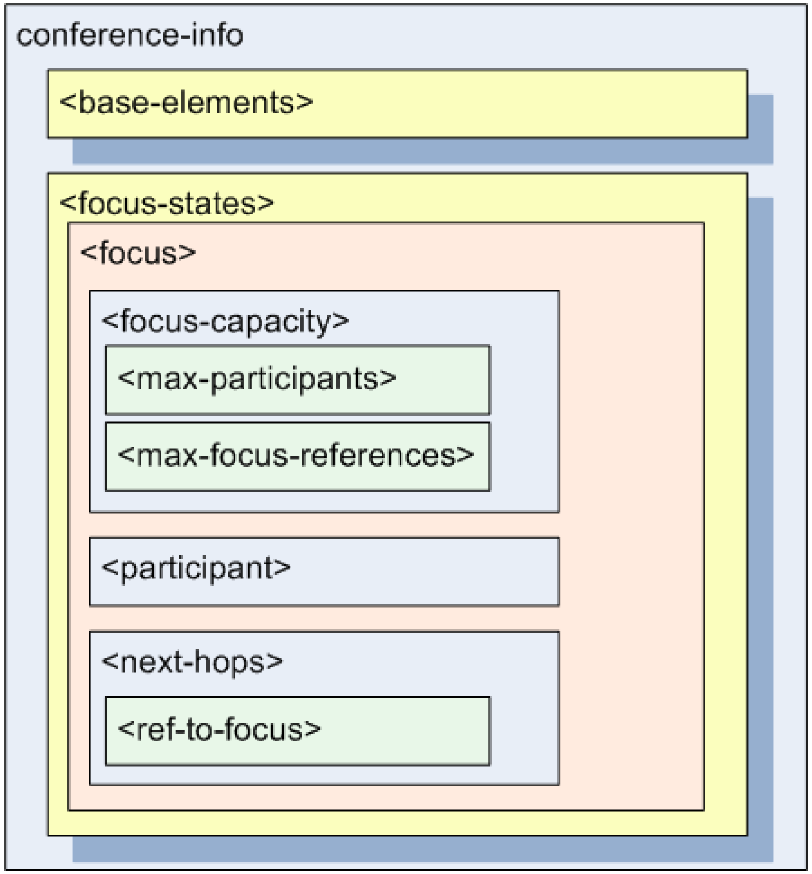
Telematics
Computer Systems

Thanks for your attention!

http://inet.cpt.haw-hamburg.de/

Additional Elements for Conference Event Package

- *Focus-states*: Container for *focus* element
- *Focus:* Describes the state at a specific focus peer
- *Focus-capacities:* Describes limits for focus peers
- *Participant:* Contains a list of all participants this focus is responsible for
- *Next-hops:* Container for all synchronization routes this focus maintains


conference-info
<base-elements>
<focus-states>
<focus>
<focus-capacity>
<max-participants>
<max-focus-references>
<participant>
<next-hops>
<ref-to-focus>