Using Geo-Spatial Session Tagging for Smart Multicast Session Discovery

Piyush Harsh & Richard Newman

Computer and Information Science and Engineering, University of Florida
Multicast! Why this low demand?

- efficient mode to transmit stored multimedia from few sources to multiple subscribers
- Still not very popular among ISPs and end users! Why?

- ISPs cite protocol complexity
- Less usable than IP unicast
- Creates low end user demand
- ISPs cite low user demand for further non deployment
- Low deployment causes further lower end-user demand. Vicious cycle!!

Vicious cycle!!
Multicast’s Low Demand! - Analysis

• ISPs cite protocol complexity
  – Shift from ASM to SSM addresses this concern!

• Low usability compared to IP unicast, why?
  – Domain names – inherently more usable than dotted-decimal IP addresses.
    • Allows popular contents to be bookmarked
  – DNS: translation between domain names to addresses
  – Search engines: allows contents to be searched
  – Existence of several applications/uses with many more coming up daily creates high user demand.
Improving multicast usability: mDNS

- Provides mechanism to assign URL style domain names to multicast sessions
  - Incorporates protocol to do URL translation between session domain names to session connection parameters!

- Allows popular sessions to be bookmarked!

- Provides multi-parameter intelligent search mechanism for end users to discover sessions
  - Real time session discovery even for extremely transient sessions!
Improving multicast usability: geo-coding

- Provides additional search parameter
- Allows better and accurate search
- Allows interesting use of multicast beyond traditional multimedia broadcast and group conferences
- Allows such services to be discovered in real time
  - Region specific disaster alert services (subscription based, push model)
  - Citizen local news multicast services
  - Content delivery from a locally hosted source for better traffic characteristics

Typical mDNS domain setup
Incorporating session geo-coding in mDNS

1° Latitude = 110.9 km
1° Longitude at latitude $\phi =$

$$\frac{\pi}{180^\circ} \times \cos \phi \times \frac{a^2 \cos^2 \phi + b^2 \sin^2 \phi}{\sqrt{(a \cos \phi)^2 + (b \sin \phi)^2}}$$

where

$a = 6,378,137$ m, $b = 6,356,752.3$ m
Geo-DB: Design and Implementation

- Level 0 grid implemented as a sparse matrix, each grid is $1^\circ \times 1^\circ$
- Depth of the tree depends on ‘k’ – the branching factor and ‘d’ the areal resolution desired (in km).
  \[ Depth = \left\lfloor \log_k \frac{111.3}{d} \right\rfloor \]
- Geo-DB is cross referenced with Global-DB and Local-DB (session databases) maintained at MSD servers at every domain.
End-User Browser search request format –

@SEARCH-REQUEST
{
    Keyword List
    Boolean Operator: AND/OR
    Location String
    Search Radius (in km)
    Session Scope: Global and/or Local
}

MSD Session Records Structure –
(depending on global, local or geo-db)

@SESSION-RECORD-DB-LOCAL
{
    Session Expiration Time
    Session Start Time
    Keyword
    URS Registered Representative Name
    Channel IP: Port
    Host Unicast IP: Port
    Stream Scope
    Geo Location Common Name
    Latitude: Longitude
}
Geo-Specific Search

Each keyword in the search request forms a separate mDNS search request.

Depending on global / local search criteria, the search request propagates to other MSD servers in the mDNS hierarchy.

Search propagation is made based on hash based keyword routing algorithm!

Hash based routing prunes requests from being unnecessarily propagated on all branches of the hierarchy.

Candidate session filtering at remote MSD servers are done by cross-referencing global-DB and geo-DB entries based on search parameters.

Candidate session details are sent to requesting end host directly from one (or more) MSD Servers

Search responses from remote MSDs are aggregated before being presented to the end user (locally at the end-user site).
mDNS – Project Status

✓ mDNS Design (complete)
✓ Hash based routing (complete)
✓ Session registration and geo-coding (using Google© Map API) (complete)
  ✓ Includes session registration request routing to correct MSD server
✓ MSD Server, URS Server implementation (complete)
✓ All session databases {global-db, local-db, geo-db} (complete)
✓ mDNS URL resolution by end-user session browser (complete)
✓ Local scope session search from end-user session browser (complete)
  ➢ Global session search request routing (ongoing)
  ➢ Candidate search result aggregation at end-user site (ongoing)
  ➢ Domain specific session search by remote user (ongoing)
✓ Network deployment and testing (to be done)
Questions & Comments

Piyush Harsh
pharsh@cise.ufl.edu
CSE Building E313A
University of Florida

Richard Newman
nemo@cise.ufl.edu
CSE Building E346
University of Florida
P.O Box 116120
Gainesville, FL 32611-6120

http://www.cise.ufl.edu/~pharsh/OSS/