

Using Collaborative Agents to Enrich Service Environments

Problem Statement:

Ad hoc peer-to-peer communication allows mobile devices to dynamically create environments (like piconets) in which they can share services and collaborate. However, these ad hoc environments that are created on the fly lack an important ingredient namely external “coordination”. Service discovery and collaboration are left completely to chance, ie. a device looking for a particular service queries its neighborhood attempting to find that service and in the best case, the device may actually find it. However, there is also an equally high probability that the desired service will not be found in that devices vicinity and hence its queries go unanswered.

Our goal is to utilize a mix of adhoc and infrastructure mode communication to maximize the probability that a device looking for a service will actually find that service in its neighborhood. We propose to use a combination of heuristics and usage profiles of services so as to pro-actively migrate services into a neighborhood in anticipation of their potential near future use. We believe that this anticipatory service migration (even before somebody actually asks for it) will improve the “hit rate” for adhoc service lookups. We are planning on building a working a prototype to demonstrate the claims

Network Model:

Our network model is similar to the notion of “infostations” that have been proposed in several previous works. Our model is comprised of a network of wireline components called “Service Portals” (similar to Mobile support stations). Each “Service portal” provides infrastructure mode communication to all mobile hosts (*MH*) that are in its cell (a well defined radius). Service Portals also carry the different services that can be offered to a user, eg. a news reader service (and associated news paper pages), a TV show viewer (and associated TV programs) etc. Service portals are geographically distributed so that their cells do not necessarily overlap. Our network is thus split into two types of zones;

- *Landing Zones* (essentially some cell where a *MH* can contact the controlling Service portal)
- *Transit Zones* (essentially the area between any two cells where the *MHs* can only talk to other *MHs* and is purely ad hoc)

High Level Architecture:

We are proposing to use these service portals to enforce some coordination in the “ad hoc” collaboration that happens within the *transit zones* in the network. Each *MH* maintains a log of the different interactions that it has observed while traveling through a *transit zone*. This includes the different types of service requests that it observed, those that were answered and those that were not. When a *MH* reaches a *Landing zone*, the Service portal controlling that zone reviews these logs (in addition to providing the *MH* any needed service). By looking at the logs, the service portal can identify services that

are needed but not available in neighboring *transit zones*. The service portal then picks *MHs* randomly and tries to equip them with these services (provided the carrier *MHs* have a capability and willingness to carry somebody else's load) so that these services can be injected into the *transit zone*. If the routes that *MHs* are taking is known to us (or learned somehow) then this information can be used to make our system more efficient (instead of randomly choosing *MHs* as carriers for in-demand services, we will choose *MHs* that are heading towards the *transit zones* that are lacking those services). Essentially our architecture does not really rely on the fact that we know the routes of all *MHs*. That is simply an optimization.

We are also planning on using the same architecture to perform the following. Consider a PDA user that desires to watch a TV show on his daily drive to work. He asks for the service at *Landing zone #1*. Now assume that we know that that user is heading towards *Landing zone #2* through *Transit Zone #1-2*. Further assume that the drive from *Landing zone #1* to *Landing Zone #2* is a 20 minute drive. Lets assume that the TV show that the user wishes to see is actually 60 minutes long. In our model;

1. The user is given only the first 20 minutes (or a little more) of data and the rest is made available to him when he shows up in *Landing zone #2*. Thus content is partitioned... There is some prior work that has been done in this area mainly from a DB perspective.... We are currently investigating exiting work in this arena.
2. Now lets assume the case wherein the user does not have memory to even hold 20 minutes of data. In our architecture, the following can be done.....
 - a. The portal controlling *Landing Zone #1* can identify some other *MH* that is heading towards *Landing Zone #2* that can carry say 10 minutes of data for that user. Essentially as long as we choose another device that we know has high probability for staying around the requesting *MHs* environment, the requesting *MHs* service requests can be satisfied by its environment
 - b. Alternately, the *Landing Zone #1* portal can give say 10 minutes to the requesting *MH*. In addition, the *Landing Zone #1* portal then informs *Landing Zone #2* portal that the remaining 10 minutes need to be sent to the *MH* before it reaches the *Landing zone #2*. To achieve this, the *Landing Zone #2* then attempts to find a *MH* that is heading towards *Landing Zone #1* and tries to piggy back the requisite data so that the users requests will be satisfied within the *Transit Zone #1-2*

Contributions:

1. Use Service portals to enforce some amount of coordination so as to improve the likely hood of service discovery and sharing when mobile devices within an area collaborate in an ad hoc fashion
2. Weak devices can now use sophisticated services mainly because their environment is willing to carry some amount of load on their behalf
3. Collaboration should improve service handling capability of the system as a whole

Assumptions:

1. *MHs* of varying capabilities ranging from laptops to cell phones
2. Each Service portal knows about all other geographically neighboring service portals
3. A common agent platform for all our agents
4. Security, privacy issues and the actual underlying transmission protocols (bluetooth, 802.11 etc) used for adhoc communication are to be determined.
5. *MHs* participating in our system are willing to carry data intended for others . Users do have the option to refuse carrying data for others.
6. *MH* velocity and other mobility characteristics to be used for our system modeling. We are currently investigating how to incorporate this in our model.

Protocol Working

(Please refer to presentation for protocol working details)