WIRELESS WORLD
RESEARCH FORUM

Using Context for Smart Group Discovery and Adaptation

M. Martin, NEC Europe Ltd.

Miquel.Martin@netlab.nec.de

***Abstract*—In this paper we highlight the relevance of human relations for the research field of Context awareness, and propose a middleware solution that enables applications and services to exploit this information. The design and implementation of the system is illustrated with a prototype application, exposing the technical challenges involved.**

***Index Terms*—Group discovery, context awareness, clustering, groupware.**

INTRODUCTION

INTERACTION between individuals is among the main factors that define human behavior. When considering Mobile Computing environments, interpretation and prediction of the user's context allows us to cope with the complexity of his environment. This has been used to provide service recommendations, content adaptation and interaction with pervasive computing devices among others.

Still, the work in the areas of context awareness, with some exceptions (e.g. [1], [2], usually fails in tapping onto this rich source of information. Consider the scenario of a user downtown at lunch time: by considering conventional context sources, one could predict that being lunch time, he might be interested in a restaurant that matches his preferences. Do these preferences make sense if the user is within a larger group? How does the situation change if group members are regular friends, colleagues or foreign visitors? Group relations in such examples, and later through the paper, prove to be of great importance to any service that intends to use Context Information.

Background and related work

In our research, we have identified two types of groups, differentiated by their motivation.

On the one hand, there are those groups defined by affinity, where members share similar characteristics which are relevant for our service. On the other hand, we have activity groups, joined by a common objective or cooperative activity, which our service will try to support. As a result, we can not assume direct interaction between members of affinity groups, but their context is still influenced by each other. Consider for instance a museum application that guides individual users to not-so-crowded highlights that fit her preferences.

The case of activity groups has been analyzed in much more detail, as in [1] which stresses relative member location for purposes like presence notification in instant messaging.

Finally, we need to consider the scalability of our system. Naturally, gratuitously searching for groups is resource consuming, so we must consider power saving strategies and a suitable trade-off between client and platform side components. In particular, we need to define a limiting factor that discriminates those users we can actually consider as candidates. While any context variable or combination thereof might be used, it will usually not make sense to consider candidates that are not within close proximity. It is common to consider only the users within the radio range (e.g. [2]) or use some sort of geo broadcasting strategy[3].

Context Aware Smart Group Discovery

Within the framework of the MobiLife [1] project, we endeavor to provide an enabler for group awareness which provides a score sorted list of groups that would suit a given user in a particular application [4].

The process is structured in three main steps: relevance assessment, group clustering and feedback gathering. The following sections illustrate how these integrate into a meaningful architecture, and elaborate the main modules.

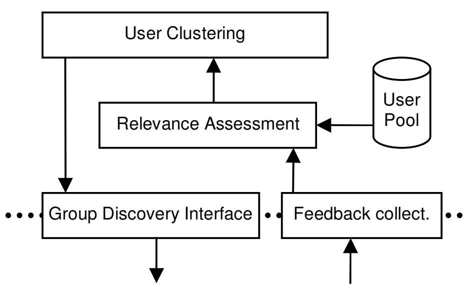

Figure 1 Architecture overview

Architecture overview

Our architecture provides an enabler-like interface that encapsulates the complexity associated to group discovery. When a group is requested, the Relevance Assessment module collects the relevant model for each user and calculates a relevance matrix as outlined in the following section.

The user clustering then infers the optimal groups given the relationships between the individuals, and returns a ranked list of groups. Finally, the feedback collection trains the models associated to each user, in order to personalize the candidate discovery process. An overview diagram is displayed in Figure 1 Architecture overview

Relevance assessment

The Profile Manager component in the MobiLife platform stores models for group allocation for each user and application.

When a user or an application requests a group, the appropriate model is retrieved from his profile and handed to a recommender system. This model contains the variables which are considered relevant for that kind of group, as customized for that user. The inference itself is done by a modular plug-in, which currently considers rule based systems and Bayesian Networks.

As an example, the MobiCar application [4] creates groups to share a car ride to a common destination. The group discovery is based on a model which, when stable, contains restrictions on the pick-up and drop-off locations of the users, their preferences (e.g. smoking tolerant or kind of music), traffic condition at their pick up point, etc…

Note that for most reasoning systems, the said restrictions need to be discretized. Furthermore, absolute values are seldom of interest, since groups often require relative comparisons between the members. For this reason, parameters are defined as Boolean operands over discrete thresholds, such as:

[*User.position – Candidate.position < 500m]*

[User.MusicPrefence == Candidate.preference]

The result of the inference is a relevancy score which indicates how suitable a given candidate is when considered in a group with the requester. By repeatedly performing this operation on the pool of available users, we obtain a relevance matrix where each position indicates the relevance between any two users. Note that such relevance is not necessarily symmetrical.

It is clear from this step that limiting the pool of candidates to consider is of capital importance, in order to avoid overloading the discovery enabler. For that reason, the pool is pre-filtered according to user defined hard policies, such as trust requirements.

User clustering

While at this step we know how each user relates to another, this is not sufficient to group the said users into optimal groups.

Consider for instance the case of users A and B, with very low relevance for each other but “joined” by C, who is very close to both of them. Without C, there would be no reason to link A and B together, unless the group type demanded a minimum number of members, in which case A-B might make as much sense as any other random link. Still, if we are able to detect the joining influence of C,

[1] IST-2004-511607 MobiLife, http://www.ist-mobilife.org

the resulting group might be much more convenient at least for C.

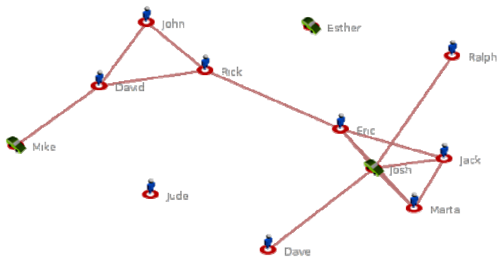

Figure 2 Relative positioning using the Force Directed Placement algorithm

Likewise, if the model for a user is not mature enough (or not evolved enough from its stereotyped start point), we might oversee relevant relations. In that case, it is likely that a transitive relation exists such that what joins B and C is a still unknown parameter which also binds A and B together.

In order to visualize these effects, we have implemented a Force Directed placement algorithm, as presented in [5].

Figure 2 shows the result of placing users in the MobiCar scenario. The nodes are initially placed randomly, and the system iterates towards a stable solution where all nodes repel each others equally, but also attract themselves according to the relevancy of the relationship.

Given this weighted graph, it is now relatively easy to identify the nuclei of closely related users, and the problem is reduced to a clustering scenario, where numerous approaches already exist (e.g. [6]).

The clustering process can be greatly simplified if restricting conditions apply. For instance, continuing on MobiCar, groups for car sharing need to include at least one car (shown as such in Figure 2) and have at most as many members as the car's capacity.

Feedback gathering

Given the nature of the relevance assessment step, we consider learning algorithms that can evolve the initial model defined by the user's stereotype. To this end, we need to collect feedback on the user's satisfaction, which will then be used for model training.

Figure 3 exemplifies this step for the case of MobiCar, but the real work is done on the platform side, where the context information during the group life has to be gathered and fed to the learning modules.

Conclusions and future work

In this paper we have highlighted the importance of context based grouping and stressed it's value in selected scenarios, as well as it's particular challenges.

We further introduce the design and architecture of the implemented middleware, and illustrate it's functionality with MobiCar, a prototype context aware application.

This approach is currently being evaluated with user studies within the framework of the MobiLife project, and we expect to fine tune both our requirements and assumptions based on their output.

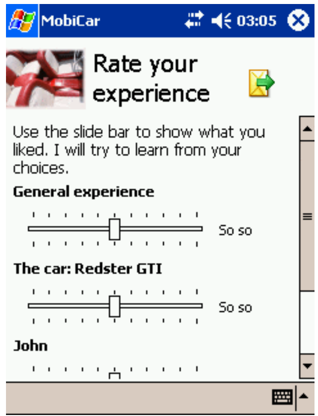

Figure 3 Feedback gathering in MobiCar

Acknowledgement

The author would like to thank Petteri Nurmi, Patrik Floréen and Jukka Suomela for their support and ideas. This work has been performed in the context of the IST project IST-2004-511607 MobiLife, partly funded by the European Union . The author would like to acknowledge the contributions of his colleagues, although the views expressed are those of the author and don't necessarily represent the project.

REFERENCES

[1] A. Ferscha, C. Holzmann, and S. Oppl., *"Context Awareness for Group Interaction Support"*. In the proceedings of the 2nd ACM International Workshop on Mobility Management and Wireless Access (MobiWac 2004), pp. 88-97.

[2] J. Parkkinen and K.Karvonen, *"Group awareness in Bluetooth"*. In proceedings of the Third International Workshop on Network Appliances (IWNA 2001).

[3] Wilfried Enkelmann, *"FleetNet Applications for Inter-Vehicle Communication"*, IEEE Intelligent Vehicles Symposium (IV 2003), pp. 162-167, Columbus, OH, USA, June 2003.

[4] Silke Holtmans, Miquel Martin, Petteri Nurmi et. al., *"D3.5, Probe 2's for Applications and Services for Mobile Family Members and Groups of Mobile Users"*, available www.ist-mobilife.org. April 2006
[5] Thomas M. J. Fruchterman and Edward M. Reingold, *"Graph drawing by force directed placement"*, Journal: Software – Practice and experience, vol. 21, num. 11, pp. 1129-1164, 1991.
[6] Romesburg, H. Clarles, "*Cluster Analysis for Researchers"*, Lulu press, ISBN 1411606175, April 2004.