Introducing Groups to an Annotation System: Design Perspective

Amjad Hawash
Dept. of Computer Science, Sapienza University
Rome, Via Salaria 113, Italy
hawash@di.uniroma1.it

ABSTRACT

Introducing groups to the MADCOW annotation system solves the privacy-collaboration problem for users. The proper matching between groups and users solved the group join problem for both groups' owners and users. We used ontological and URL-based measures to execute the match. For ontological-based measures, MADCOW *domains* were used and linked with external knowledge repositories: ontologies. URL-based measure depends on calculating the number of URLs annotated both by group members and by MADCOW users external to the group as a method for quantifying shared interests. In this work, we describe the system, the problems, and their solutions, with reference to the design choices made.

Author Keywords

Annotation; Groups; Joining Process; Ontologies.

ACM Classification Keywords

H.5.3 Information interfaces and presentation: Group and Organization Interfaces. Web-based interaction.

INTRODUCTION

Digital annotation is the process of adding information to contents of a multimedia document, enriching with additional valuable information, without altering the original content. Reasons for annotating can vary: to create mementos or clarifications of interesting content; to entertain discussions with other users; to construct or integrate documents [7, 2, 10]. On-demand training and education processes also use annotations, e.g. for learning foreign languages [6, 12, 11].

MADCOW[1] supports the annotation of (portions of) texts, images and videos with textual content, links to other resources, and user-defined tags [7], through a 3-tier architecture (see Figure 1).

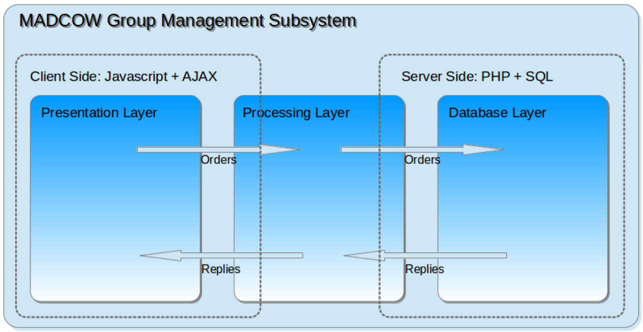


Figure 1. MADCOW 3-tier Architecture.

Annotations published in MADCOW are of three types: (1) *public*: viewable by any user, (2) *private*: viewable by their submitters only, or (3) *group-related*: viewable by any member of the group to which they are posted, and nobody else.

Users Collaboration-Privacy Conflict

Public annotations increase collaboration among users while private ones preserve users' privacy. Introducing groups to MADCOW solved this conflict through the notion of group-related annotation: group members can collaborate on annotations posted to that group, which are on the contrary hidden for external users [5]. A group owner can select one of three different policies to allow users to join it: (1) `Public`: any user can join the group, (2) `Invite`: users can join a group only if invited by some group member with *authoriser* status[2], or (3) `Apply`: users apply for joining the group, and admission is subject to approval by one of the authorisers.

Manual Group-Join Problems

The manual joining process presents two main problems:

Irrelevance Problem: Authorisers deciding to invite users to their group and users looking for relevant groups share similar, if symmetrical, problems: "**how do authorisers know who could be interested in joining their group?**" or "**what groups exist which might interest a user?**". This causes users to be faced with a wealth of irrelevant data when choosing target groups or users.

[1] Multimedia Annotation of Digital Content Over the Web.

[2] This is given to the group owner and to group moderators.

Time-effort problem: Without matching, authorisers have to list all MADCOW users (possibly looking at their public annotations) and select some of them as receivers of invitations. With knowledge provided only by the title and a textual description of the group topic, users can send join requests to the groups considered relevant to their interests. Both processes consume time and effort and become unwieldy as the number of groups or users increases. In an extensive test, average times were calculated for all groups' operations (Table1) showed a maximum amount of time for groups' join (99.25 sec.).

Table 1. Number and average duration (secs) for operations.

	Create	*Update*	*Invite*	*Join*
# of times	72	51	719	125
Average	37.3	15.9	99.25	5.6

Automatic Groups-Users Suggestions

Manual-group joining problems can be solved by proper groups-users matching so that authorisers are presented with the most relevant users and users with the most relevant groups. The association of groups with publicly available representations of knowledge relevant to the group objectives is performed by selecting well-defined terms in which to express domain knowledge [8].

To achieve this goal, a group's owner can associate it with some existing ontology, either by manually selecting the most appropriate one, or by providing a set of terms which reflect the intent of the group, to be matched with the terms of each ontology in the ontologies repository. Figure 2 represents a fragment of ontology repository scheme.

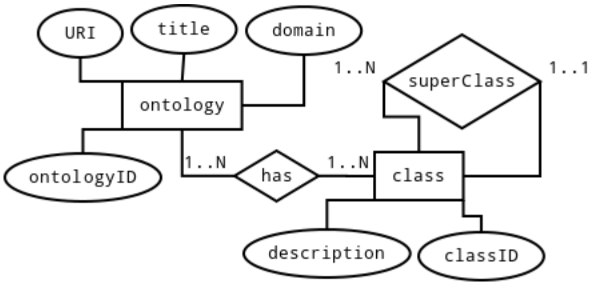


Figure 2. A fragment of the Ontology Repository Scheme.

Currently, the association between a group and an ontology is 1-1 from the group side, and 1-many from the ontology side, that is a group can be associated with one ontology, while an ontology could represent more than one group. Between ontologies and domains, the association is 1-1 from both sides. Figure 3 depicts both associations types.

On the annotator side, users can complement their annotations with tags to represent the intent of these annotations. The suggestion of users to groups is performed in the **MADCOW Ontology Browser**, which executes a matching between the terms in the ontology for the representative domain and the tags of public annotations. The same matching occurs for a user searching for proper groups, where the match will include all of the user's public and private annotations.

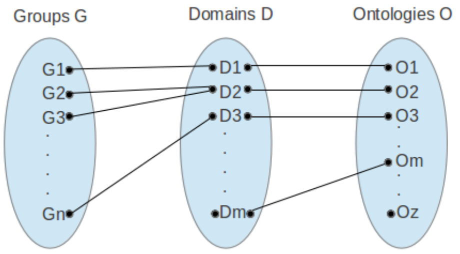
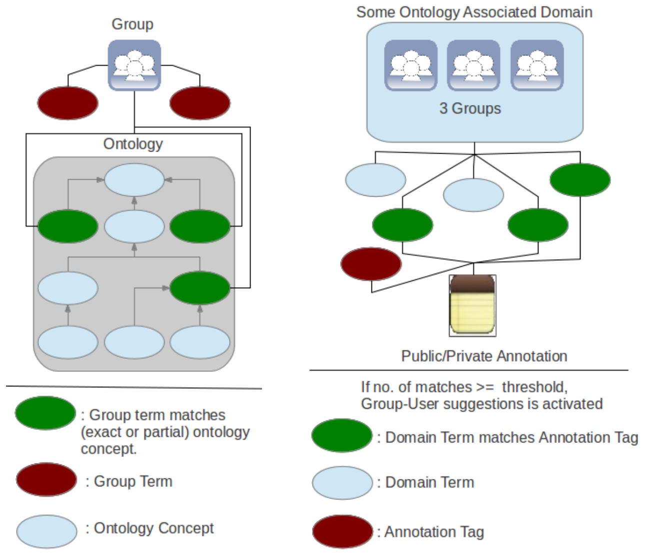


Figure 3. Associations between groups, domains, and ontologies.

The *Class Match Measure (CMM)* presented in [1] is used to measure the group-user relevance in the ontology-based matching [3, 4, 9]. Authorisers for a group may request a ranked list of likely annotators. Also users can request a ranked list of domains, for each of which the set of groups referring to it is given. Figure 4(a) depicts the matching between a group and an ontology, while Figure 4(b) depicts the matching between a domain and a user. A pilot test on the use of *CMM* has shown a decrease in the average invitation time from 99.25 to 10.6 seconds [4].


Group
Ontology
: Group term matches (exact or partial) ontology concept.
: Group Term
: Ontology Concept
Some Ontology Associated Domain
3 Groups
Public/Private Annotation
If no. of matches >= threshold, Group-User suggestions is activated
: Domain Term matches Annotation Tag
: Domain Term
: Annotation Tag

(a) Group-Ontology Matching (b) Domain-User Matching

Figure 4. A visual depiction of Group-Ontology and Domain-User matches.

Another relevance measure is based on the collection of Web sites a user targets for annotation. Users annotating specialised sites usually share the same interests. Discovering that some fellow annotators already belong in some group could encourage others to be members in that group. On the other hand, group owners can look for users to invite among those who more frequently annotate Web sites which are targets of annotations for group members.

SYSTEM ARCHITECTURE AND BEHAVIOUR

Figure 5 presents a fragment of the ER diagram defining the logical scheme of the MADCOW database.

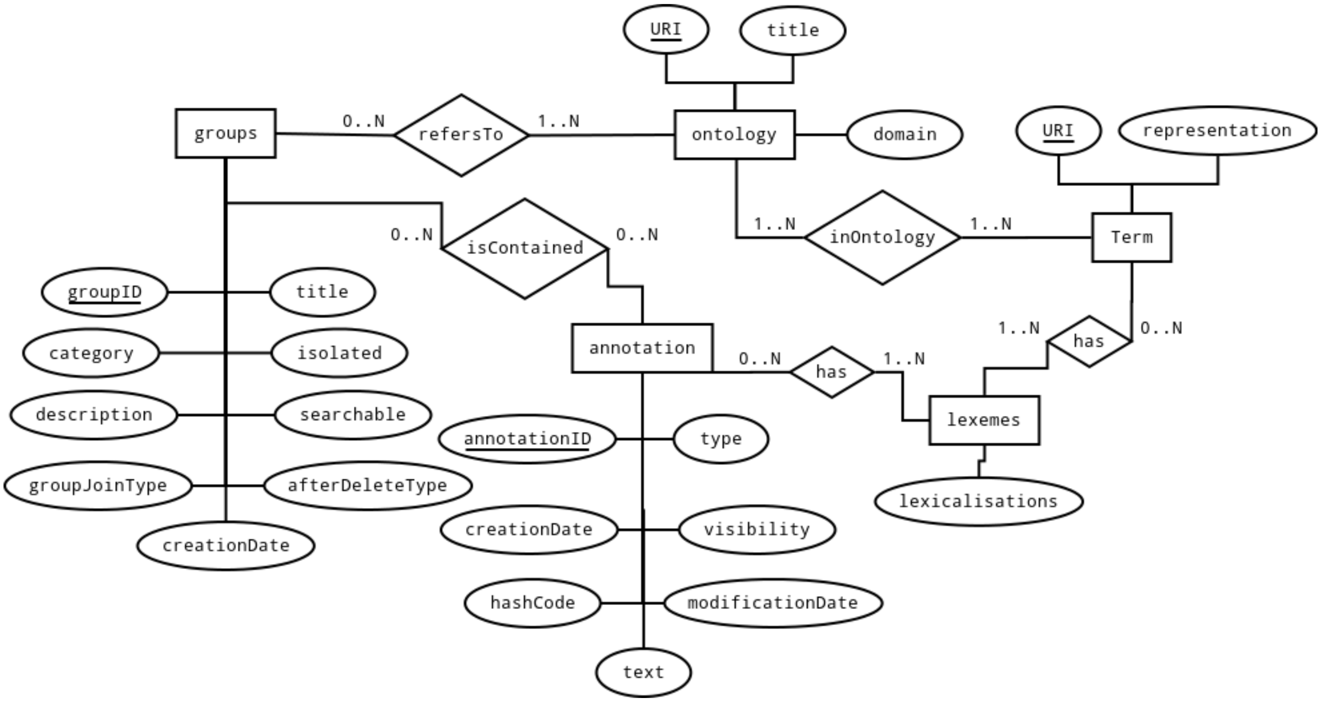


Figure 5. A fragment of the MADCOW Entity-Relation diagram, considering groups, annotations, and ontologies.

The entity *groups* is used to save related data to each group created in the system with its properties. The entity *annotation* is used to store all the data relative to an annotation. If an annotation is related to a group, a special record is saved in the proposed entity of the relation *isContained* that relates *annotation* and *groups* entities. The link between a group and a suggested or chosen ontology is modeled as a relation *refersTo* that creates a relation between the entities *groups* and *ontology*. The property *title* of the *ontology* entity maintains the name of the ontology. The entity *Term* is related to the entity *ontology* by the relation *inOntology* so that an ontology has one or more terms, while the entity *lexemes* describes its possible lexicalisations. The same table is used to maintain tags used in annotations which do not have a direct reference to terms in a domain ontology associated with a group.

Data traffic between client and server is managed via a JavaScript *XMLHttpRequest* (here and in the following we refer to Figure 6) object which acts as an intermediary to send requests from client to server and receive results in the opposite direction. In particular, *XMLHttpRequest* is used in the following processes: (1) highlighting annotated resources; (2) managing the display of possible group targets in the annotation pop-up window; (3) displaying annotations when hovering on an annotated resource; and (4) sending invitations to join a group.

The process of relating a group to a domain is triggered when a group owner selects one of his/her groups to be associated with a domain. As a consequence, the *MADCOWPortal* object contacts the *MySqlServer* to load all available domains to be displayed for the owner, who can ask to display also the concepts and lexemes related to each domain. If a suitable domain is found, the *MADCOWPortal* executes a PHP function to create the association between the group and the domain. Otherwise, the owner can ask the system to suggest suitable domains, in which case the *MADCOWPortal* object requests the *MADCOWMatcher* object to execute a match between the terms and all concepts/lexemes for all available domains. A ranking function ranks all matched domains according to their relevance with the terms and returns it to the *MADCOWPortal*. The group and one of the domains can then be associated, following a user selection.

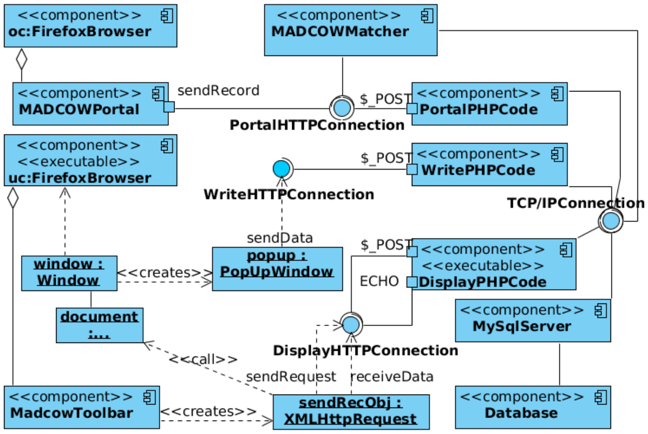


Figure 6. The components involved in annotation-related and group-related processes.

A group owner could ask the system to suggest the most appropriate users to be candidate members in his/her group. The *MADCOWPortal* contacts the *MADCOWMatcher* to execute a CMM matching between the concepts of the representative domain and all tags of all public annotations for all users in the system (loaded from the *MySqlServer*). Upon this match, the *MADCOWMatcher* returns the list of all related users, to be displayed to the group owner. If the request is one by the group owner to suggest users based on URL matches, the same objects are involved, but this time the *MADCOWMatcher* executes the match between all URLs annotated by group members and all URLs annotated publicly by all users external to the group. In both cases, when a set of users is selected for invitation, the *MADCOWPortal* contacts the *MySqlServer* to save suitable invitations for the selected users. The same objects and functions are involved when users ask the system to suggest suitable groups to join, in which case also tags of private annotations are involved in the CMM matching process, and all URLs privately annotated by a given user. Figure 7 represents the sequence diagram for actions take place in the case of CMM or URL matching.

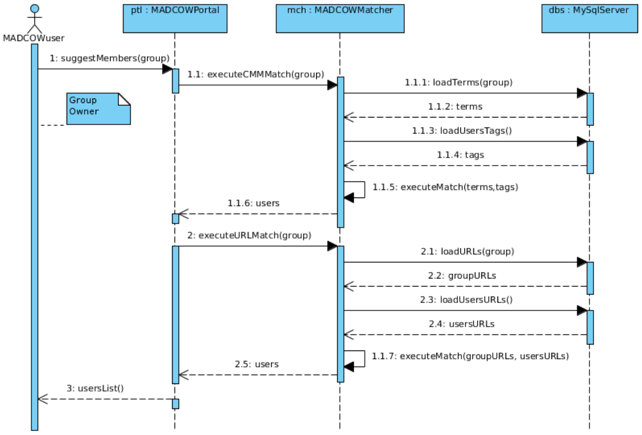


Figure 7. CMM and URl Matches Sequence Diagram.

CONCLUSIONS AND FUTURE WORK

MADCOW groups system offer a solution to the users' privacy-collaboration conflict. Problems related to manual groups-join are solved by users-groups matching based on involving ontologies as representatives for groups knowledge. A description of system mechanism for users-groups matching is illustrated. Results from experiments and participants' feedbacks are promising. As future work, we plan to allow group-domain multiple associations, tuning the ranking values by Fuzzy Logic, and to study other relevance measurements.

REFERENCES

1. Alani, H., Brewster, C., and Shadbolt, N. Ranking ontologies with aktiverank. In *Proc. ISWC'06*, Springer (2006), 5–9.
2. Avola, D., Bottoni, P., and Genzone, R. Light-weight composition of personal documents from distributed information. In *Proc. IS-EUD 2011*, vol. 6654 of *LNCS*. Springer, 2011, 221–226.
3. Avola, D., Bottoni, P., and Hawash, A. Using ontologies for users-groups matching in an annotation system. In *Proc. CSIT 2013* (2013), 38–44.
4. Avola, D., Bottoni, P., and Hawash, A. Users-groups matching in an annotation system: Ontological and url relevance measures. In *Proc. CSIT 2014* (2014), 100–109.
5. Avola, D., Bottoni, P., Laureti, M., Levialdi, S., and Panizzi, E. Managing groups and group annotations in madcow. In *Proc. DNIS 2010* (2010), 194–209.
6. Bargeron, D., Grudin, J., Gupta, A., Sanocki, E., Li, F., and Leetiernan, S. Asynchronous collaboration around multimedia applied to on-demand education. *J. Manage. Inf. Syst. 18*, 4 (Mar. 2002), 117–145.
7. Bottoni, P., Civica, R., Levialdi, S., Orso, L., Panizzi, E., and Trinchese, R. Madcow: a multimedia digital annotation system. In *Proc. AVI'04* (2004), 55–62.
8. Brewster, C., O'Hara, K., Fuller, S., Wilks, Y., Franconi, E., Musen, M. A., Ellman, J., and Shum, S. B. Knowledge representation with ontologies: The present and future. *IEEE Intelligent Systems* (January 2004), 72–81.
9. Hawash, A. Introducing groups to an annotation system. In *Proc. CHItaly2013* (2013), 43–54.
10. Heck, R., Luebke, S., and Obermark, C. A Survey of Web Annotation Systems, 2008.
11. Lai, Y.-S., Tsai, H.-H., and Yu, P.-T. Integrating annotations into a dual-slide powerpoint presentation for classroom learning. *Educational Technology & Society 14*, 2 (2011), 43–57.
12. Sakar, A., and Ercetin, G. Effectiveness of hypermedia annotations for foreign language reading. *J. of Computer Assisted Learning 21*, 1 (2005), 28–38.