Patterns for Building Communities in Collaborative Systems

Till Schümmer
Computer Science VI - Distributed Systems
FernUniversitaet in Hagen
Informatikzentrum, Universitaetsstr. 1, D-58084 Hagen, Germany
till.schuemmer@fernuni-hagen.de

Abstract

Virtual communities offer a large potential for rich human interaction. While communication technology and the internet have become ubiquitous, a culture for social interaction in virtual communities is still evolving. This paper proposes a pattern language for helping newcomers to integrate and orient within a virtual community. It focusses on the process of getting to know each others and establishing interesting connections within the community.

1 Introduction

Establishing a community is a hard job. People have to be motivated to join the community, they have to be open to recognize the community, they have to be motivated to take the first step and act within the community, and they have to be encouraged to continue their interaction in the community.

After defining the context by classifying different kinds of communities, this paper will look at one important aspect of communities: their means for integrating new members and maintaining the relationships between the community members. It presents a pattern language that aims to ease these processes.

1.1 A classification of communities

Preece (2000) defines online communities as a set of "people, who interact socially as they strive to satisfy their own needs or perform special roles, such as leading or moderating." These people share interests or needs, which make up the purpose of the community. They use group policies such as rules and assumptions to guide social behavior and computer systems to mediate the interaction.

Taking this definition, an online community is much more than just keeping in contact by means of e-mail. Places for communitiy interaction have to assist users to find common goals or purposes, make them explicit, and feel as a part of a the

community. There have to be ways of interacting with other community members and sensing their presence. And finally, there has to be an incentive to return to the community. Otherwise, there is no social binding achieved.

An increasing number of community providers has realized that virtual communities can become very binding if they reach the phase of social involvement (e.g. (Telewest 2001)). According to Chapman (2001), frequent interaction among the community members is very important to reach this stage of social involvement.

There are different classifications of communities. Most classifications distinguish at least *communities of interest, communities of purpose, and communities of practice* (cf. Marathe (1999), Carotenuto et al. (1999), or Chapman (2001)).

Members of a ***community of interest*** share the same interests in a topic (and often a common background). Examples are discussion groups on television shows or people interested in planets of the solar system. Some authors (e.g. Carotenuto et al. (1999)) also define *communities of passion*, which are very close to communities of interest. The difference is that the members are more involved in the community's topic up to the point where they become passionate advocates. Actually, a community of interest can become a community of passion. An example is the discussion group on a TV show that became a fan club of the show's host.

Communities of purpose consist of members who share a common (short term) goal. For instance customers at a virtual bookstore share the goal of finding and buying books. They all have to go through the same process (i.e. selecting the item and checking out) and they can help one another reaching the goal. Thus, the community of purpose has a functional purpose and it may dissolve after the goal is reached. In contrast to communities of interest, the members don't necessarily share the same interests. They are not likely to start activities that exceed the community's purpose (Carotenuto et al. 1999).

If the members of a community share a common profession, the community is called a ***community of practice***. Their members reflect on the way, how they perform their tasks and enhance their ways of work in a community learning process. Since the community's topic is the member's profession, members are normally highly involved in such communities. Concrete communities of practice are for instance Smalltalk programmers who meet in a Smalltalk users group to shape the process of programming. In some cases, it makes sense that the community is established by the merchants. Given the example of the Smalltalk users group, the main vendor has been heavily involved from the beginning. The well known experts are employees of the company and their involvement in the online discussions is beneficial for both parties (customers and vendor): the customers can get help from the experts and the vendor can steer the discussion regarding his business plans (carefully up to a specific degree; too much advertising would put off the customers).

Marathe (1999) adds another type of communities: a ***community of circumstances,*** which is defined by common circumstances such as current life situations, e.g. children hitting puberty. Interaction in these communities is often personally focused and third parties are not involved in the community. Therefore, these communities may not easily be influenced, but they can lead to a strong binding to a seller, if this seller is present in important life stages.

1.2 Forging links in communities

All different community types discussed in the previous section share the aspect that they need to achieve a social binding. The members should *forge links* to other members that make them return to the community and contribute to the community's goals.[1]

Fortunately, the human nature is trained to join and act in communities. While there are cultural differences in human interaction, one can observe at an abstract level that some rites are comparable across different cultures.

For instance, most cultures have developed ways to welcome each other, to establish bonds with strangers, or to maintain contacts to other community members. Most cultures have established social incentives that honor helpful behavior even though in some modern individualistic contexts these incentives vanish.

These inherited rites focus on personal interaction in the physical world. In virtual environments, a culture of interaction (or should it rather be called a non-culture) is still emerging that often neglects the basic rules of thumbs that one could observe in traditional interaction in the physical world.

Consider for instance a face to face interaction where information is exchanged within a group. An important phase of this interaction is across many cultures the process of welcoming each other. It is important to greet and to establish an atmosphere of trust. It is important to be aware of one another and see, with whom one interacts. Can a virtual community exist without these rites? I argue that these rites are extremely important to establish successful communities. But many environments for computer-mediated interaction do not support these rites. They focus on the task (e.g. the exchange of information) and suppress the social context of the task.

A basic assumption: the following patterns focus on fostering constructive behavior in collaborative environments. But unfortunately not all environments are only used by users who want to contribute to a group result. Newsgroups for instance can be flooded with spam or users can offend other users in *flame wars*. For the purpose of this paper, I ignore these users although it is important to address the problems raised by destructive behavior. First patterns addressing issues of destructive behavior and ignored privacy concerns in public interaction are Attention Screen$_{\rightarrow 3}$, Don't Disturb$_{\rightarrow 3}$, and Masquerade$_{\rightarrow 3}$, which are provided as thumbnails at the end of this paper.

[1] Even communities of purpose should make the members return after the individual purpose is reached. Returning members can assist new members to reach the purpose and a community of purpose can by this way be converted to a community of interest.

2 The Pattern Language

Within this paper, I collected a small set of patterns that show, how the social context of interaction can be reintegrated in virtual interaction of communities. The goal of this collection is twofold: first, I would like to collect good practices for establishing social bonds within a virtual community and for integrating newcomers; second, the collection should make the reader sensible for detecting the lack of social support in current virtual environments.

The structure of the pattern language is shown in figure 1. Patterns are shown as nodes in the figure. The connections between the nodes show some of the relations between the patterns. Note that the map shows only the most important relations and only some of the related patterns for a better readability.

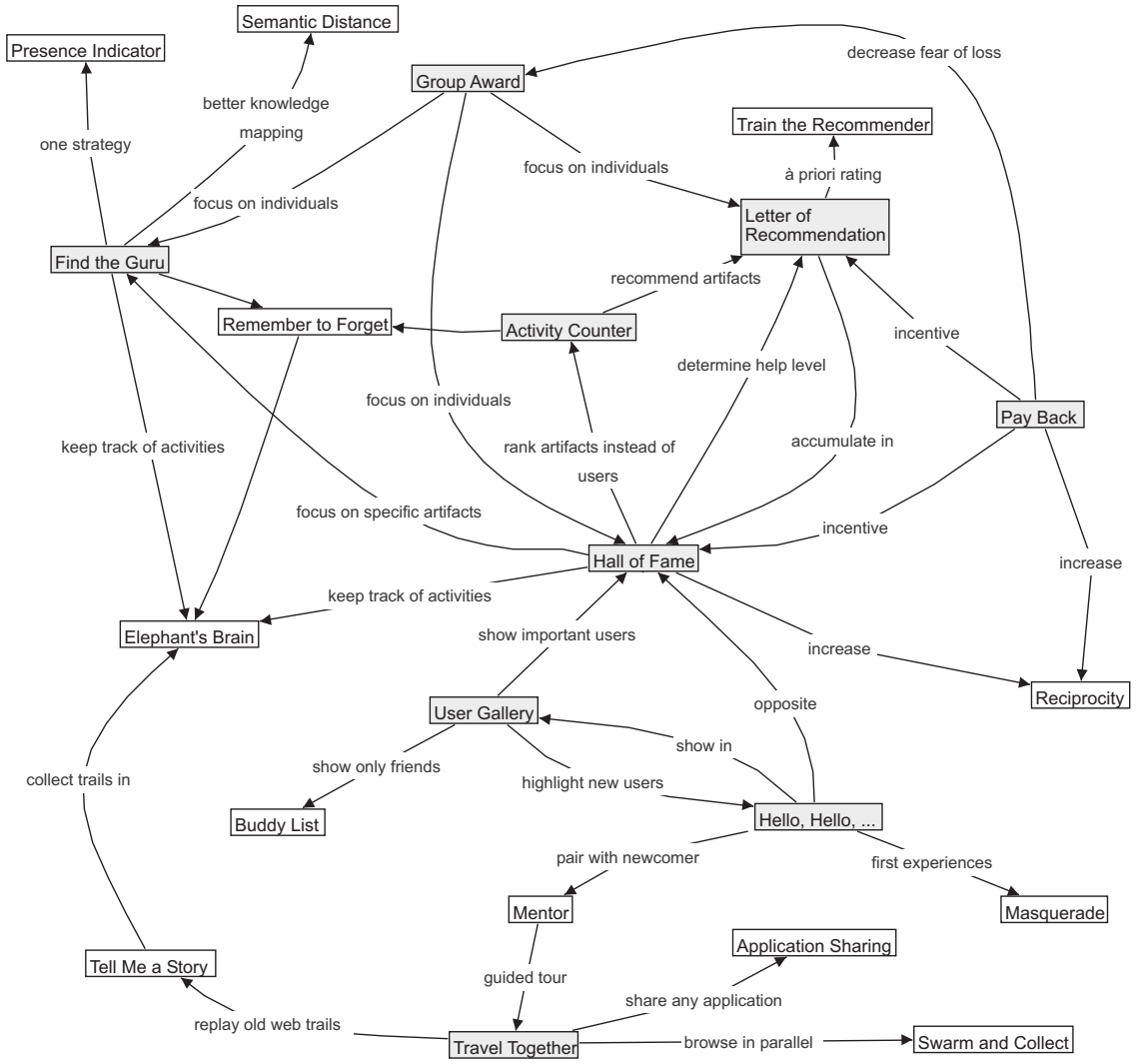


Figure 1: The structure of the pattern language.

The shadowed nodes represent patterns that are included in a long form in this chapter. The uncolored nodes represent patterns that interact with the patterns included in this paper. These patterns are currently written or have been published in other papers and will only be included as thumbnails at the end of this paper.

Note that in order to keep it simple the diagram does only show a subset of the patterns presented in the *additional thumbnails* section.

The patterns of this collection are *socio-technical patterns* which means that they describe social processes and their supportive technology. Although each pattern provides known uses that implement the pattern by means of computer technology, most of the patterns can also be found in traditional human interaction.

2.1 The Pattern Structure

The patterns in long form have the following structure:

The pattern name appears as a section title. Then follows a sensitizing image together with a scenario that should help to remember the pattern. A small version of the pattern map serves as a means for orientation. The current pattern is shown as a filled (black) node in this diagram.

The context section helps to decide whether or not the pattern may fit into the reader's current situation. It lists other patterns that the reader may have considered before and provides hints regarding the community or group size to which the pattern applies.

Then follows the core of the pattern composed of the problem and the solution statement in bold font separated by a scenario and a symptoms section. The scenario and the symptoms sections provide different access means to the pattern. The scenario tells a real world story where the problem could arise. The symptoms section describes aspects of a situation where the pattern is missing more abstract again. It lists observable forces that are unbalanced before the pattern was applied.

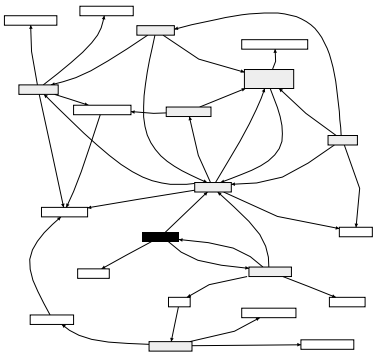
After the solution section, the solution is explained in more detail (collaborations, rationale, danger spots, known uses) and indications for further improvement after applying the pattern are provided (in the related patterns section). The collaborations section explains the main components or actors that interact in the pattern and explains how they relate to each other. The rationale section provides positive consequences of the pattern and thus explains, why the forces are resolved by the pattern. Unfortunately most patterns also have negative consequences so that new unbalanced forces arise. These negative forces are described in the section labeled Danger Spots. Thus, the symptoms section and the danger spots section together include the full set of forces changed by the pattern.

References to patterns are shown in Small Caps. If the pattern is part of this paper, the section in which the pattern is explained is provided behind the pattern name.

The patterns have two intended groups of readers: End-users should read the patterns to inform their selection of requirements. These readers will concentrate on the scenario, the problem, the solution, and the known uses section. Although the patterns present solutions on a very high level of abstraction, they may need software developers to implement the solution. Thus, software engineers are the second intended group of readers. They will mainly focus on the problem, the solution, the forces (found in the symptoms and the danger spots sections), and the collaborations section.

2.2 User Gallery

Alternative name(s): User Directory; Member Directory



Intent Show who is using a collaborative application.

Context You are building a system where users should actively participate in the community. The users are identifiable by their real name or a nickname.

Scale: The group size is increasing so that the interaction shifts from small group interaction (3-7 members) to interaction in a larger community where people do not necessarily know each other.

Problem **If more than one user interacts with shared data, it is hard to coordinate the interaction - especially with strangers. Without knowing who is using the system, it is hard to establish collaboration or to become aware of other users' activities.**

Scenario Imagine a class of students in a traditional university. The students are asked to create groups for a semester long lab exercise. Thus, all participants of the course come together in the first days of the semester and introduce themselves. They exchange information about their personal preferences and make social conversation to get to know each others.

Based on the impression that each student has from the other students, they form groups for future work.

Now consider the same goal in a virtual environment: The students should form groups that run a virtual lab exercise. But the problem is that the students don't know one another. So how should they form groups without exchanging personal information first?

Symptoms *You should consider to apply the pattern when ...*

– new users hesitate to get in contact with other users of the community.
– users find it hard to remember who is a member of the community.

Solution **Therefore: Provide a list with all users who are members of the community. Let the members provide personal information in this list that is related to the community's task. Design this list in a way that it is interesting to browse.**

Collaborations Users register to the system and provide personal information. The information about all registered users is displayed in a user gallery. This gallery includes the user's name or pseudonym and in many cases a picture of the user.

Rationale Since the users can browse through the list of community members, they can establish a sense of the community. They can compare other users' descriptions with their own preferences and find users who share their goals or preferences. This eases the process of getting in contact with other users.

By filling out their own profile, the users will construct a virtual identity that emphasizes those aspects that are important for the interaction in the specific collaborative environment. The profile describes their virtual identity with their expectations in the community. Visual clues can additionally support the process of getting an impression of the other users.

Finally, if the gallery uses pictures or provides interesting personal information, it will be fun to read for the users. This has the effect that users will return to the gallery and stay up-to-date with the current community members.

Danger Spots For large communities you should make the user gallery searchable. Otherwise it is hard to find interesting contacts in the user gallery.

When designing a User Gallery, one has to carefully balance the amount of information that a user has to provide with the benefits of anonymity. For instance, it can be advisable not to include user pictures if the discussion should not be based on visual attraction.

Known Uses **VITAL** (Pfister et al. 1998) is a groupware application that supports collaborative distance learning. While the application is targeted on synchronous interaction in small groups, it also supports learners in finding potential co-learners. Learner can browse all registered users in the user list in the world browser (shown in fig. 2).

The details for the users are shown on request in an external window, where the user may enter contact information and a

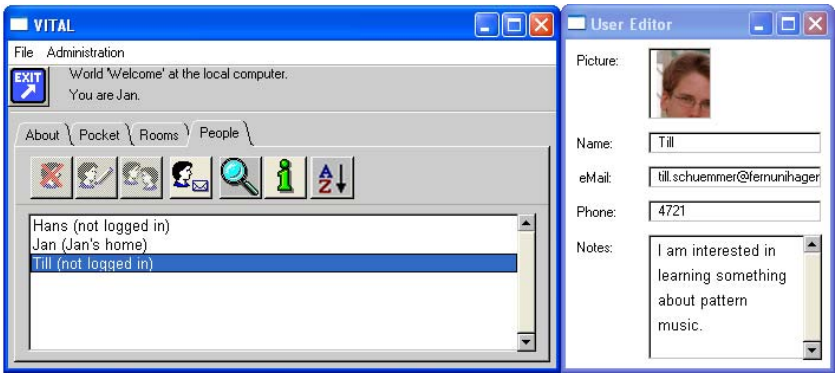


Figure 2: The VITAL world browser showing all known users.

short description.

ChiPlace (Girgensohn and Lee 2002) lists people of the CHI community in a people browser. They are arranged according to their interests provided in their User Profile.

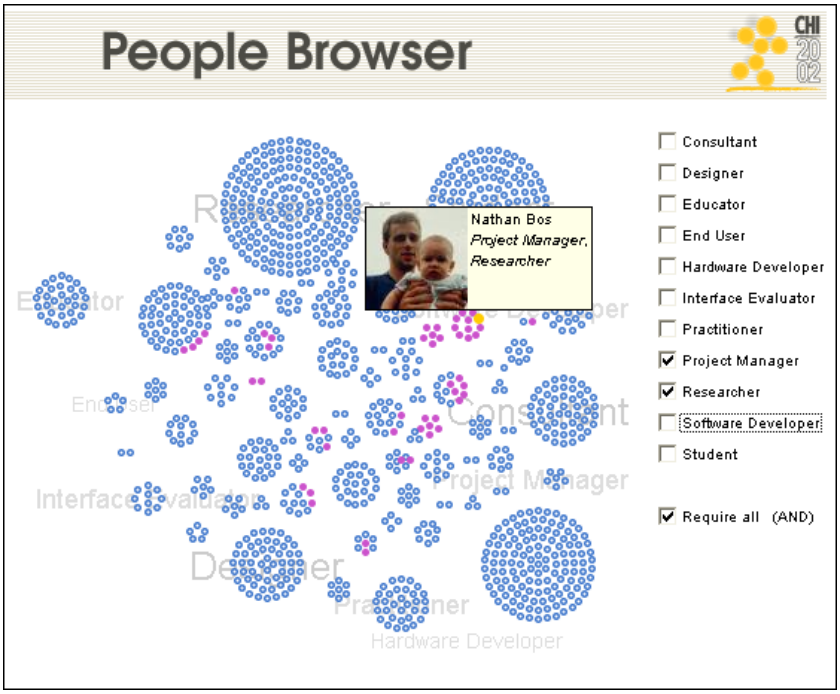


Figure 3: Clustered users in the CHI-Place People Browser (`http://chiplace.fxpal.com/people/browser.jsp`).

Figure 3 shows the user interface of the people browser. A visitor can tag different professions to narrow the set of relevant users. The relevant users will be shown with dots of a different color. When moving with the mouse over a dot, the related user information appears as a pop-up window. If users want to find other users, they have to move with the mouse over various dots. By doing so, they also see the picture and

classification of users they probably have not seen before.

Community web sites like `www.communities.com` often list the users in a User Gallery. During the registration process, the users are asked for personal information (in a structured profile). Parts of the personal information and the photo of the user are shown in the User Gallery.

Figure 4: User Gallery at www.communities.com.

Related Patterns Hello Hello$_{\rightarrow 2.3}$**:** In cases, where the community is large, it can be difficult to find new users in a user gallery. Thus provide a special area in the community, where new member are introduced. In combination with a user gallery, this could for instance mean that the new users are shown at the beginning of the user gallery.

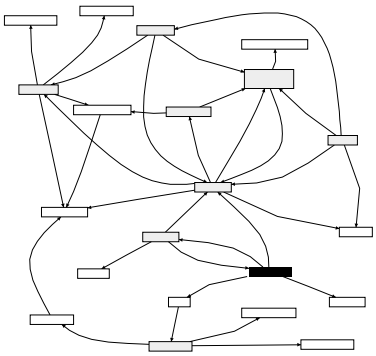
Hall of Fame$_{\rightarrow 2.9}$**:** Some users may be more important to the community than others. These users can be collected in a special user gallery, the hall of fame. In combination with the user gallery, the important users can be displayed at the beginning of the user gallery.

Buddy List$_{\rightarrow 3}$**:** Another way for keeping the user gallery manageable in a large user population is to show only those users that the local user knows in a Buddy List. Note that this pre-selection of buddies does no longer support the forging of new links between up to then unknown users.

User Model Definition (Vogiatzis et al. 2005) provides means for structuring the user's self-description. Users are asked to tag their main interests which eases to find users with specific interest profiles (especially users with comparable interests – cf. Birds of a Feather$_{\rightarrow 3}$).

2.3 Hello Hello

Alternative name(s): Welcome Area



> You say yes, I say no. You say stop and I say go go go, oh no. You say goodbye and I say hello. Hello hello I don't know why you say goodbye, I say hello.
>
> John Lennon, Paul McCartney

Intent

List new members of a group or a community at a prominent place and introduce them to other members.

Context

You have established a group and first users have created social bonds. By this way, the group has found its identity. The group members distinguish themselves from people who are not part of the group.

Scale: The group is part of a larger community. Potential group members have entered the community but have not yet found peers for interaction.

Problem

If the group wants to progress, it is often needed that they integrate new members. But since the group members are very focused on their internal interaction, they may fail to notice new potential members and ignore their possible contribution.

Scenario

Consider a scientific conference in the area of computer science. More than hundred researchers gather for several days to present their research results and exchange ideas. Besides the exchange of new ideas, the main objective of the conference is to maintain existing and extend new research contacts.

But will all participants be able to exchange ideas? More experienced visitors are eager to maintain existing contacts and get the latest news from colleagues they met in the years before. This takes time and time is limited. It is often only by accident that

returning visitors get in contact with researchers who participate at the conference for the first time. And – since the community does not know the newcomers' faces – it is very likely that the ideas of the newcomers will only have limited space for discussion and exchange.

In contrast, one can often observe that newcomers and returning participants form two distinct groups – with distinct parties in the evening. This makes the exchange of ideas and learning inside the community difficult.

Symptoms

You should consider to apply the pattern when ...

- long time community members have established tight links and are eager to communicate within the community.
- long time members share a large collective history, in which new members did not play any role.
- the community is large enough to allow the formation of subgroups so that long time community members form a closed group.
- new members find it hard to enter the community.
- new members are ignored by long time members.
- fresh ideas that bring the community forward are provided by new members but ignored by existing members.

Solution

Therefore: Provide a prominent place in the community's interaction space where new members and their ideas are introduced. In a computer mediated group, this can for instance be a special section on the group's home page. Whenever a new member joins the community, ensure that the existing group members notice the new member.

Collaborations

Welcome Area: The welcome area is a prominent region in the community's interaction space. It can be a special spatial area in a physically co-located community, a special page in a virtual web-based community, or a special topic or thread on a discussion board. While the first two examples are bound to a spatial dimension, the last example is bound to a temporal dimension. It denotes a specific time span where the community members use their interaction space as a welcome area.

Newcomer: The newcomer joins the community. He introduces himself in the welcome area and explains why he wants to be a part of the community.

Veteran: A veteran shares a long interaction history within the community. He is an accepted member and keeps many social links to other people. The veteran commits to visit the

welcome area frequently. In cases, where the welcome area is a specific point in time, the veteran commits to participate in the community at this point of time.

Rationale

Since the newcomer is asked to introduce himself, he is provided with a forum, where he can articulate his ideas and thoughts that drove him to join the community. Since all newcomers act this way, the newcomer does not have to fear that his introduction could disturb existing members.

Due to the commitment of the veterans, they will notice, when the newcomer introduces himself. Thus, the newcomer will be recognized by the existing community members.

The benefit of using a designated welcome area is that the introductions do not interfere with other group interaction. Veterans can decide consciously, when to visit the welcome area (in cases where it is a special place) or the whole group will focus on the welcome area at the same time (in cases where the welcome area is a specific time frame). This ensures that there is no overlap between the introduction of new members and the interaction within the group. Whoever is present in the welcome area concentrates on the welcoming phase. This way, veterans and newcomers construct a common experience that can serve as the basis for creating a collective history.

Danger Spots

Newcomers may not wish to attract much interest at the first time they join the group. They may need some time to look passively at the group and see how the group members interact. In this case, you should move the welcoming ceremony to the point of time, when the member actively decides to participate in the group. Until then, newcomers can act with a MASQUERADE$_{\rightarrow 3}$.

Another problem can be to convince veterans to commit themselves to pay attention to the welcome area. Veterans may just not see the need for investing efforts for newcomers. This is an indication that the community is resistant to growth. Considering participation in the welcome area as part of the metric that calculates the users' ranking in the HALL OF FAME$_{\rightarrow 2.9}$ can help to solve this issue.

Known Uses

ChiPlace (Girgensohn and Lee 2002) was used at CHI2002 to foster networking between attendants of the conference before the conference took place. Users could provide a description of themselves. When entering the people page (cf. figure 5, left), a list with new members was shown in a prominent section (at the beginning of the page). Clicking on the user's name opened a user profile (cf. figure 5 right).

www.visualbuilder.com is a community of software developers. New members are listed in a designated welcome area on the

Figure 5: New members at CHIplace.

entry page (shown in the left part of figure 6). Clicking on the user's login opens another page that shows the user's profile (right part of figure 6).

www.communities.com (cf. also the known uses section of USER GALLERY$_{\rightarrow 2.2}$) allows to query new members can be queried by clicking on the "New Members" link (in the right part of figure 7). This will show the list of users that is shown in figure 7. By clicking on the user's login, one can see the provided information.

Games at EuroPLoP are special slots in the conference schedule where the participants meet for cooperative games. The games have the purpose of getting in touch with each other. Name games help to learn the participants' names and other games help to find similarities between the participants.

Related Patterns MENTOR$_{\rightarrow 3}$**:** Newcomers who introduced themselves in the welcome area can be paired with a mentor who accompanies the newcomer in his first steps and personally introduces him to relevant veterans.

Figure 6: New users at www.visualbuilder.com and a user profile.

MASQUERADE$_{\rightarrow 3}$**:** To allow newcomers to interact within the community without exposing themselves, they can move through the community anonymously. The MASQUERADE pattern discusses this form of interaction in depth.

HALL OF FAME$_{\rightarrow 2.9}$**:** The HELLO HELLO pattern focusses on the introduction of newcomers to veterans. The opposite is done in the HALL OF FAME: here, honorable community members are presented to the community to provide newcomers with an orientation, who influenced the community most.

BIG JOLT (Manns and Rising 2004) proposes to invite a knowledgable expert to the community who will tell about his experience in the field. Inviting the expert to the welcome area can serve two means: First, the welcome area receives a higher reputation since it is not only a place to meet newcomers but also to get in touch with experts. Second, newcomers will learn more about the ideas that the community propagates. If the expert is a community member himself, he will probably be one of the top members in the HALL OF FAME$_{\rightarrow 2.9}$.

USER GALLERY$_{\rightarrow 2.2}$**:** The welcome area is often a special section of the USER GALLERY.

INTRODUCTION SESSION (Fricke and Völter 2000) proposes a welcoming phase at the beginning of a seminar. It shows how the problem of learning more about newcomers (in this case all seminar participants are considered as newcomers) can be resolved in a co-located seminar. The solution is to "take time at the beginning of the seminar to let everyone introduce him-/herself to the others."

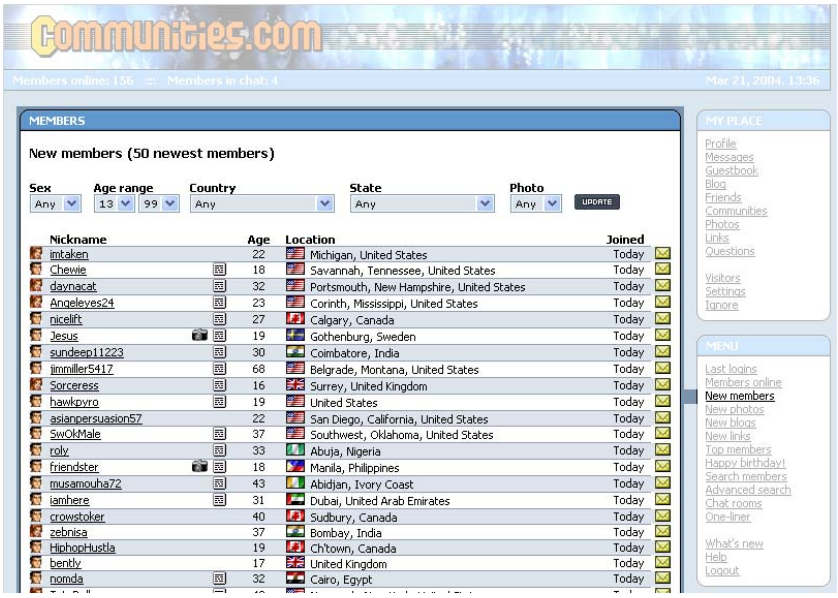
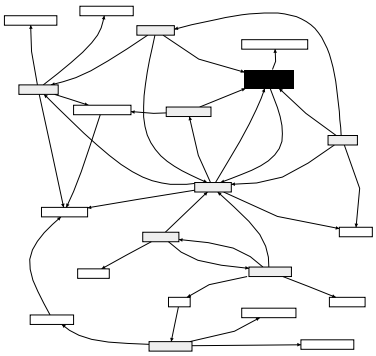


Figure 7: A welcome area at www.communities.com.

2.4 Letter of Recommendation

Alternative name(s): Rating, User Experience Feedback



Intent

Let the users rate other users regarding their reliability or expertise.

Context

Your system allows users to select their interaction partners. It assists them by letting each user provide detailed information about himself (e.g. in a User Gallery$_{\rightarrow 2.2}$). Now you are searching for a way by which objective information can be added to increase the trustiness of the information.

Scale: The pattern is intended for interaction in larger groups and communities. It requires at least a group size of three (although it does not make much sense in these small groups).

Problem

When users don't know potential interaction partners, they may fear interaction because they don't trust the partner. This may result in a high inhibition threshold and thus non-existent interaction.

Scenario

Imagine that you are travelling and look for a hotel. It is already quite late, thus you cannot visit and review all hotels in the city. Fortunately, you have bought a hotel directory, which lists all addresses of hotels in the city. This makes it easy for you to find a hotel.

But how should you select a hotel? The hotel directory only lists the information on the hotel that was provided by the managers of the hotel. Can you trust them or is it more likely that the managers have presented their hotel better than it really is? Since you cannot know this, you select a hotel that sounds good – but it remains a question of luck how you will sleep in the hotel.

Symptoms

You should consider to apply the pattern when ...

– there has been no prior interaction between the interaction

partners, which means that the partners could not collect experiences on the quality of interaction before.

- the interaction consumes resources and it takes some time before a user can judge on the quality of the interaction.
- investing resources on interaction is not desirable, if a satisfying result cannot be guaranteed.
- the interaction requires trust in the interaction partner or the content.

Solution

Therefore: Let the users rate the interaction and display an analysis of all users' ratings together with the users or artifacts the user interacted with.

Collaborations

The user interacts with an interaction partner. The interaction partner can be another user or an artifact. Interaction with an artifact represents indirect interaction. This means that one user created an artifact which is later on perceived by the local user. Interaction with users can for instance be the exchange of information or goods between the users.

After the interactive episode, the user rates the quality of interaction in the episode. He expresses, how well his requirements for the interaction were met. The rating is expressed on a scale, e. g. from *poor* to *great*. It can also include a textual description why the episode was rated this way. The ratings are collected in a central repository or as an attribute of the rated artifact or user.

Whenever an artifact or a user is shown to other users, the system also shows the average of other users ratings.

Rationale

The basic assumption of this pattern is that users will act in a comparable way if they face comparable situations. Thus, if a user had bad experiences with an interaction partner, a third user will very likely also make bad experiences with this interaction partner, if the situations are comparable. On the other hand, if other users were able to interact with the interaction partner successfully, a third user will also probably have a successful interaction.

If the rating is performed on artifact level, it means that an artifact that was helpful for other users will very likely be helpful for the local user.

The knowledge of the quality of former interaction episodes can strengthen or weaken the trust in a potential interaction partner. Increased trust lowers the inhibition threshold and thus eases the start of an interaction.

The pattern does not reduce the time that is needed until the user has made his personal judgement on the interaction quality, but it reduces the risk of investing too much efforts in failing interaction.

Danger Spots

It depends on the application domain, whether the ratings will rate users or artifacts. In cases, where the artifacts are not relevant for a large user group, it can be more important to rate the users associated with the artifacts instead. Although this transfer from an artifact experience to the user who created or offered the artifact can work in many cases, it may also lead to wrong assumptions: the user could think that other users rated a user and not an artifact. Systems should therefore carefully explain any transformations applied to the ratings.

Ratings on users can be offending for the rated user. Since a rating is always a personal opinion, it can hurt the rated user in an unjustly way. You should therefore give the rated user the possibility to comment on the rating. Additionally, you should consider to moderate the ratings if destructive ratings are a problem.

Related to this issue is the fact that users can learn from prior mistakes. If a user received a bad letter of recommendation, it is possible that the user will change his interaction style so that the reported problems are fixed. Users should thus be able to rehabilitate themselves.

It can be difficult to detect the end of an interactive episode. In systems, where the interaction follows strict workflows, the episodes may last from the first to the last appearance of the user. In systems with no strict workflow, interaction episodes may start and end at any time.

The rating can distract the user from his task. The means for providing ratings should thus be unobtrusive.

Known Uses

eBay uses recommendations to indicate the reliability of a user. Since trade interactions at eBay include the transfer of money, trusting the interaction partner is a crucial issue. After each transaction, the users are asked and reminded by email to rate their interaction partner. The rating consists of a number (-1 for negative rating, 0 for neutral rating, and 1 for a positive rating) and a textual comment.

The numbers are used to calculate an overall rating for the user, which is then displayed together with his user name at each product that the user sells (cf. figure 8-left).

The overall rating computation counts all users who rated positively and subtracts all users who rated negatively. A user's opinion is counted only once even if the user provided more than one recommendation. This ensures that the rating does not depend too much on the rating user. If a user made more positive than negative recommendations, then his recommendation will be counted as a positive recommendation. If negative recommendations prevail than a negative recommendation is counted.

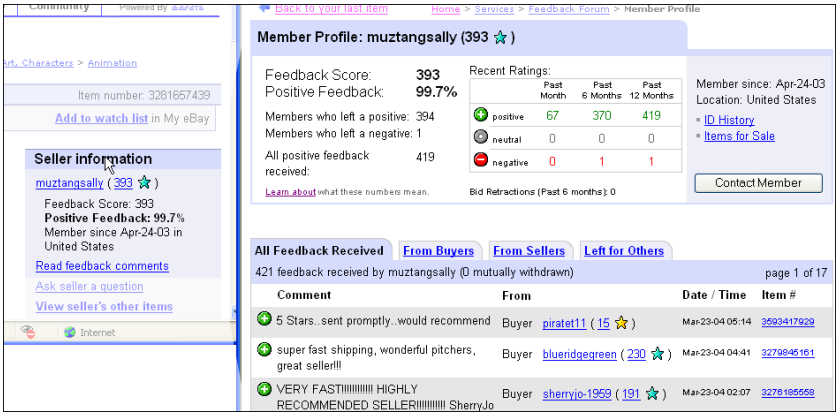


Figure 8: Rating transactions at eBay.

Seti@Home is a distributed application that seeks for extraterrestrial intelligence. Clients download raw data for later offline analysis.

While this web site started as a community of purpose (for finding extraterrestrial intelligence), one can observe a shift to a community of interest. The site designers encourage users to create a personal home page where the users can describe themselves and share their thoughts about Seti@Home (cf. figure 9).

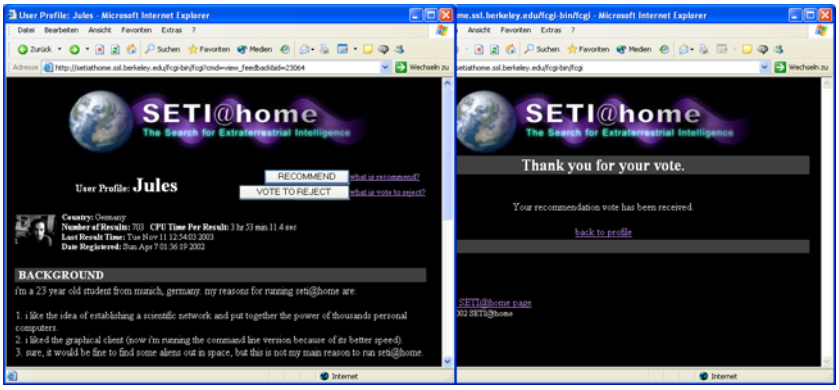


Figure 9: Recommending a profile at Seti@Home.

Other users can browse the home pages and rate the provided information: They can decide to recommend it to the community (the buttons labelled "Recommend" in fig. 9). Those users with positive recommendations are selected to become a user of the day who is shown on the Seti@Home entry page.

Amazon.com provides various means for rating the presented information. First of all, the users can rate the presented items and create a review (e.g. for a book). Each item is shown with a link labelled "Write a review". The users are encouraged to share their thoughts whenever they browse an item.

Reviews contain a textual comment on the item and a star rating (1–5). The average of all star ratings is displayed together with the item. Items can be sorted according to the number of stars they have, which eases the process of finding popular items.

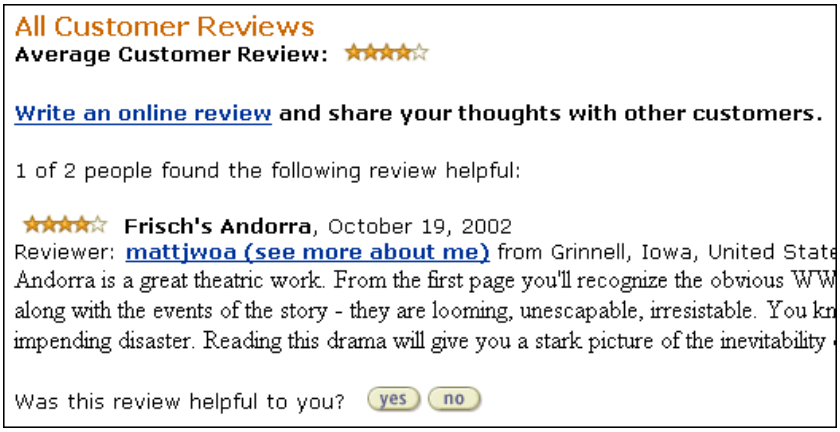


Figure 10: Rating a review at Amazon.com.

The second means for providing feedback is a rating of other users' reviews. Each review is shown with buttons asking whether or not the review was helpful for the user (shown in the lower part of figure 10). In each review's heading, the system shows, how many users found the review helpful. The reader is thus able to skip reviews that were not considered as useful by other users.

The rating of the review is also used to calculate a score for the reviewer. A user who created many helpful reviews is considered as a popular user (and receives a place in the Hall of Fame$_{\rightarrow 2.9}$). Anyhow, as mentioned in the danger spots section, this can be problematic. It is important that the users understand that famous reviewers became famous because other users liked their reviews.

A third means for providing recommendations is the rating of external merchants. As with the eBay trade relations, trust is an important issue when buying items from external merchants. Users can rate their experience (again on a star scale from 1 to five stars). The ratings are then translated to -1 (1 and 2 stars), 0 (3 stars), and 1 (4 and 5 stars) and a merchant's rating is calculated in the same way as at eBay.

Related Patterns **Hall of Fame$_{\rightarrow 2.9}$:** Recommendations for users can be accumulated in a Hall of Fame. If a user received many positive

recommendations, he will receive a prominent ranking in the Hall of Fame.

Train the Recommender$_{\rightarrow_3}$**:** Instead of rating interaction episodes *a posteriori*, users can also be asked to rate the interaction episodes *a priori*. This means that they provide an estimation how helpful an interaction episode could be based on their knowledge of the possible interaction partners. Train the Recommender supports this by letting the users rate computed recommendations. The main difference lies in the quality of the users judgements: if the users judge on quality after they experiences an interaction, they can base this judgement on empirical facts. In an a priori judgement, the user can only extrapolate experiences from other interactions to the proposed interaction.

Participant's Feedback Form (Fricke and Völter 2000) proposes to provide participants of a seminar with a feedback form at the end of the seminar. Students should be encouraged to rate the teacher so that the teacher can further on refine his teaching. The Letter of Recommendation pattern goes beyond this approach since the results of the feedback forms are presented to the participants and potential new participants again.

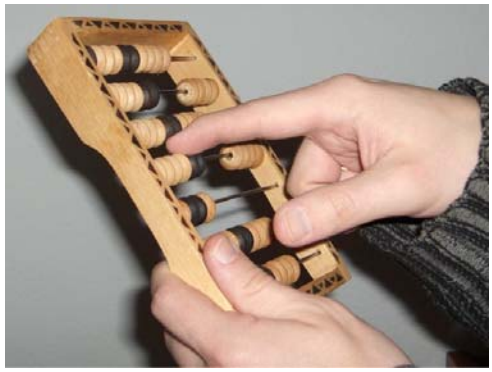
Rehabilitation$_{\rightarrow_3}$ discusses how a bad reputation can be improved if a user changed his behavior.

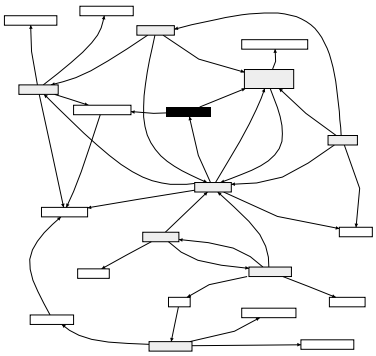
Vote$_{\rightarrow_3}$**:** The letter of recommendation can be modeled as a vote if the users are only provided with a rating scale and no means for a free text comment. The *seti@home* example follows this approach. The Vote pattern does not restrict the theme of the election to users or artifacts It is rather possible to elect on any question.

Collaborative Filtering$_{\rightarrow_3}$**:** The letters of recommendation can be used to find other users with comparable recommendations and then point to other interesting subjects.

2.5 Activity Counter

Alternative name(s): Visitor Counter





Intent

Show how many users interact with a specific unit of information.

Context

Your user community returns to important artifacts frequently respectively makes use of the artifacts frequently.

Scale: The group is large enough so that group members do not necessarily know what other group members do or have done. This may even be true for loosely coupled interaction between two users.

Problem

In a collection of shared objects, there may be more and less important objects. Especially for a newcomer, there is no easy way to distinguish important from less important objects. This may result in a situation where the newcomer gets lost.

Scenario

"Larger animals are more expensive to maintain in zoos than are smaller animals, and they breed at slower rates. Consequently, zoos could potentially contribute more to conservation efforts by concentrating on smaller-bodied species, but they could lose patronage if the zoo's public prefers to see larger species.

Therefore, we examined the relationship between the popularity of zoo exhibits and the body sizes of the animals displayed. We measured popularity as the proportion of people passing an exhibit who looked at it for more than 10 seconds and compared popularity to the body size of the animals in the exhibit. We found that exhibits of larger animals were preferred by both adults and children."

Abstract of the Publication "The Relationship between Popularity and Body Size in Zoo Animals" from Ward et al. (1998).

Symptoms

You should consider to apply the pattern when ...

- users experience it difficult to decide on the attractiveness of a region of the community space by just looking at the contained artifacts.
- attractive artifacts are used frequently by users who know the artifact.

Solution

Therefore: Add an activity counter to the visualization of the shared artifact. Artifacts that are important for the community will have a high number of activities such as visits, downloads, or updates. Unimportant artifacts may not attract as many visitors and therefore have a low activity counter value.

Collaborations

Users access artifacts in a collaborative environment. Whenever the artifact is accessed, an activity counter for this artifact is increased.

When users are looking for prominent artifacts, they submit a query to the system that returns the artifacts ordered by their activity counters' values. The query can be triggered by following a hyperlink or it can be shown automatically with the artifact. The granularity of the counter can vary from counting all accesses in the community in one counter over counting accesses to a cluster of related artifacts up to counting accesses for each individual artifact separately.

Rationale

The pattern helps users to find places where other users were interested in. Although this is no indication of the information's validity, it reflects navigation decisions made by previous users. This decision is exactly what the local user is trying to make when looking for orientation in a collaborative environment. Thus, looking at other users' previous decisions can serve as a guideline for the local user.

Danger Spots

Having a low number of visits does not always imply that the object is not important. It may be also new (and thus not yet well known in the community) or it may be hard to find.

In the case of new artifacts, you should attribute the visitor counter with a date (the creation date of the artifact) that states, when counting started.

You can also consider aging of activity counter's values. This means that all activity counters are decreased at a fixed point of time (e.g. at 12am each day or every Sunday noon).

Although this creates lower numbers of visits, it may be more accurate for sites with changing content. Remember that a large number of visits is not helpful for the user if the visits were visits to an older version of the artifact and thus are misleading for the

current content.

Visiting an artifact does not provide any information on the artifact's content but only on the way how it was advertised. Users visit the artifact because they follow a specific reference at another artifact. The more prominent or attractive this reference is, the more hits may an artifact get. One thus should be careful when interpreting activity counters.

One way of including an artifact's attractiveness in an activity counter could be that the counter does not count hits but the time that a user interacts with the artifact.

New users can be mislead by absolute numbers since they depend on the overall access frequency in the community. In very small communities with many artifacts, the difference between 2 and 5 visits may be significant whereas in large communities, significant differences mean several hundred accesses. Normalizing the activity counter with an average access counter can thus further help to distinguish popular from irrelevant artifacts.

Finally, the activity counter bears all problems that are related to metric measures. It depends on the definition of the metric to judge whether or not the metric reflects importance. The metric should be designed in a way so that it is robust regarding manipulations. Otherwise, users will try to manipulate the activity counter for specific artifacts to direct newcomers to these artifacts. Distinguishing relevant and less relevant activities while calculating the number of activities can make the metric more exact. If such a distinction is possible depends on the application domain.

Known Uses

Web Counter: Are normally graphical images that represent the number of visits to a specific page (fig. 11). When the page is loaded by the client's web browser, the browser will continue loading the image. This results in a request to the server that is counted. Based on the counter's value, the server generates an image and returns this image to the client.

Figure 11: Web Counter.

One problem with web counters is that they often do not distinguish human visitors from artificial "visitors" such as search engines that traverse the web for indexing pages. Thus, the provided number is inaccurate for web sites where the user does not need to identify himself for accessing pages.

A second problem is the fragility regarding manipulations as it was discussed in the danger spots section: some web counters simply add their value if the site is accessed. A manipulation

to the counter's value is thus very easy. The manipulating user could either manually reload the page with the counter very often or he could use a piece of software that programmatically requests the page very often.

ResearchIndex (Lawrence et al. 1999) uses activity counters to determine the popularity of a scientific publication. Together with the header of each document, the system shows the total number of citations found for this document (the framed part in figure 12).

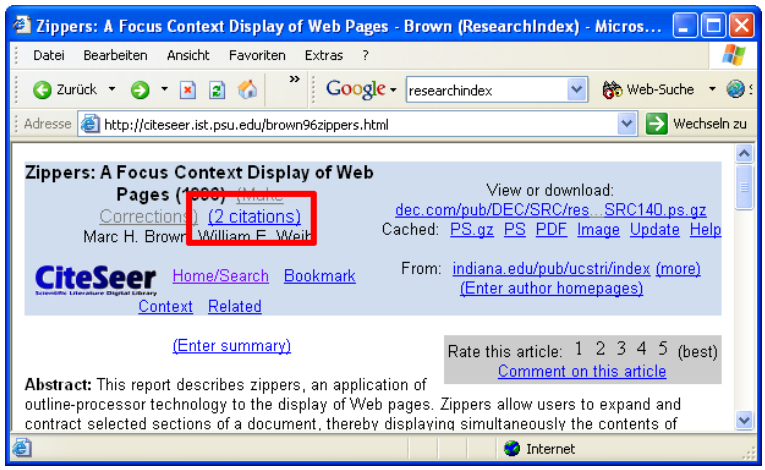


Figure 12: Citation Counter at CiteSeer.

In this example, citing related work counts as an activity while only reading the work is not rated as important.

freshmeat.net: The open source community page at freshmeat.net provides different measures of activity counters. The site counts

- the number of *record hits* for a specific project home page located at freshmeat,
- the number of *URL hits* to an external project page that leads off of freshmeat,
- the number *subscriptions* to the project (e.g. users asking to receive information about project updates or participate in project discussions), and
- the total number of *announcements* for a project.

From these values, freshmeat calculates the popularity and the vitality of a project and shows this in the project listings. The popularity is calculated as

$$popularity = \sqrt{(recordhits + URLhits) * (subscriptions + 1)}$$

which means that the number of hits is brought into relation with the number of subscriptions using an Euclidean distance.

The vitality is calculated as

$$vitality = \frac{announcements * age}{lastannouncement}$$

where *age* measures the days that a project exists and *last announcement* measures the days passed since the last announcement. This has the effect that projects with many announcements will stay vital for a longer time than projects with a low number of announcements.

The two measures show how more than one activity counter can be combined to one meaningful measure.

Related Patterns

Letter of Recommendation$_{\rightarrow 2.4}$**:** Many visits to an artifact can be interpreted as a Letter of Recommendation. The Letter of Recommendation does not only count accesses but encourages the users to reflect and comment on the accessed artifact.

Remember to Forget$_{\rightarrow 3}$ ensures that old visits will not be counted for too long.

2.6 Find the Guru

Alternative name(s): Expert Finders, Expert Recommender, or Expertise Recommender (Yiman 2000).

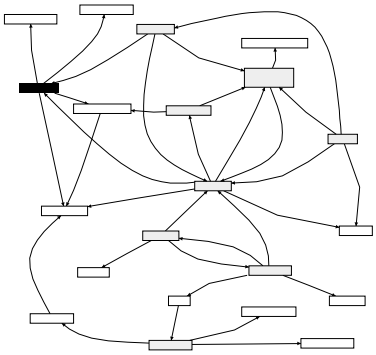


Photo: Photo Library of Congress, free at www.visipix.com

Intent

Contact the user who is most likely able to help with a specific artifact.

Context

You provided an environment where users can interact with artifacts. But users need more detailed information than that what is provided within the artifact. Thus, you are thinking about providing users with personal help regarding specific artifacts.

Scale: The group is large enough so that group members do not necessarily know what other group members knew about the artifacts in the interaction space. This may even be true for loosely coupled interaction between two users.

Problem

You know that other users have more expertise with the artifact, but you don't know who they are.

Scenario

Imagine that you are visiting an unknown city. Everything is new for you: the roads, the shops, and the monuments. Orientation is thus a challenging task for you. Once you are lost, you ask a stranger how you can reach your hotel again. But the stranger is also new to the city and thus cannot help you much. He is lacking the experience of walking through the city and can not remember the hotel you were mentioning.

Fortunately, you finally find a person who lives in the city since 20 years. He has walked the path from your current position to the hotel several dozens times in these years. Thus he can describe you the perfect route that leads to your hotel. Regarding pathfinding, he was a real guru.

Symptoms *You should consider to apply the pattern when ...*

- users are not making their expertise explicit (e.g. in yellow pages) and thus there is no directory of expertise (discussed by Yiman (2000)).
- there are no global experts (e.g. shown in a HALL OF FAME$_{\rightarrow 2.9}$ that can answer any question or the global experts are tired of answering questions since they are asked too often.
- different users make comparable mistakes since there is no way of sharing experiences.

Solution **Therefore: Find the user who shares a long history with the artifact. Use the ELEPHANT'S BRAIN$_{\rightarrow 3}$ to see who performed activities on the artifact. Sort the list of people according to the number and type of activities and/or the time that has passed since then.**

Collaborations Users perform activities on artifacts, which are tracked in an ELEPHANT'S BRAIN$_{\rightarrow 3}$. Whenever a user is working with an artifact for which he would need personal assistance, he starts a query for the user who is most likely to provide more information on the artifact. This user can be found with different strategies:

1. It can be the user who has most lately looked at the artifact. In this case, it is probable that the found user shares a common context with the requesting user (cf. the PRESENCE INDICATOR$_{\rightarrow 3}$ for more details on this strategy).
2. It can be the user who has most recently modified the artifact. This implies that the user actively interacted with the content and detected a reason to change the artifact, which requires at least some experience.
3. It can be the user who performed the largest number of activities with the artifact. This case can again be distinguished regarding reading and changing activities.
4. If the effect of the activity can be measured in the sense of an activity size, it can be the user, who performed the largest activities on the artifacts. This can be combined with the two last strategies.

If one combines all strategies in one calculus, the expertise level for a user ($expertise(u)$) can be determined as follows:

$$\begin{aligned} expertise(u) &= \#act_r(u) * f_r + \#act_w(u) * f_w \\ &+ \frac{f_{lastr}}{(now - t(lastact_r(u)))} + \frac{f_{lastw}}{(now - t(lastact_w(u)))} \end{aligned}$$

The function $lastact_r(u)$ determines the last read activity of a specific user u. The function $lastact_w(u)$ finds the last write activity of u. The functions $\#act_r(u)$ and $\#act_w(u)$ calculate the

number of read or write activities of a specific user. All the above functions only consider activities for the focussed artifact. The factors f_r, f_w, f_{lastr}, f_{lastw} define how important the different strategies discussed above are considered.

Strategy 1 can be modeled by choosing $f_{lastr} > 0$ and setting all other factors to 0. Strategy 2 can be modeled by choosing $f_{lastw} > 0$ and setting all other factors to 0. Finally, strategy 3 is modeled by setting $f_{lastr} = f_{lastw} = 0$, $f_r > 0$ and $f_w > 0$. Normally, one would select $f_w > f_r$ to reflect the fact that a user, who changed an artifact concentrated more on his activity.

For strategy 4, one can adapt the function $\#act_w(u)$ so that it weights each activity with the contribution made to the artifact. How this weight is calculated depends on the nature of the artifact. Basically, it should reflect how much the artifact changed due to a user's action and how much of this change survived in the latest version of the artifact.

Rationale

McDonald and Ackerman (1998) performed an ethnographic study how team members use artifacts to locate expertise regarding these artifacts. The authors interviewed team members of a software development team to find out how they act when they have a question:

> *"When a programmer makes a change in a program he is supposed to add his mnemonic to the line and update the date. This is how we know who last changed the program. Whoever made the the last change in the program is the default expert in that program. ... It's close enough. The logic is that the person who spent time on it has it freshest in memory and so they are the best person to ask a question"* - a user called Brad in (McDonald and Ackerman 1998).

From this quote, one can see that there are traditional means for judging on an user's expertise. The authors discovered one problem that can make the above approach fail: if users only performed small changes, they are often no experts for the artifact.

This motivates a means for finding experts that is based on the number or the impact of changes (strategies 3 and 4).

Regardless the chosen strategy, the system will provide the requesting user with another user who has experienced the artifact before. This user can be considered as an expert since he can share his experience with the requesting user.

Danger Spots

It can be very complicated to find a right measure for measuring the size of a contribution (in strategy 4). An example is the contribution in terms of text lines for textual artifacts. A simple comparison of the final result, the part contributed by the user,

and the version which the user modified leads to the number of lines that survived in the final version.

Consider three versions of a text $v_1, v_2,$ and v_3. Assuming that v_3 is the final version and v_2 is the contribution made within a specific activity, one can calculate the impact of v_2 by comparing it against its predecessor v_1 and the final version v_3. The comparison of two version each leads to two sets: the set of removed lines $r_{(i,j)}$ and the set of added lines $a_{(i,j)}$. The impact of a version i to the version j is then calculated as

$$impact(v_i, v_j) = r_{(i-1,i)} - a_{(i,j)} + a_{(i-1,i)} - r_{(i,j)}$$

which means that those additions and removals of version i are considered that survived in version j. This impact can be transformed to a scalar value by considering the size of the set $impact(v_i, v_j)$.

But counting lines does in most cases not provide the information needed to measure the essence of the contribution. The number of lines or words does not necessarily relate to the amount of knowledge that is expressed with the words. And the knowledge is the important factor for finding experts. Armour (2004) discussed this problem for the context of counting lines of code in software development projects:

> *We actually want to count how much knowledge there is (will be) in the system and there is no way to empirically measure knowledge.* (Armour 2004)

Using a SEMANTIC DISTANCE$_{\rightarrow 3}$ instead of simple text comparison can provide more exact calculations for the degree to which the artifact changed. In this case, the comparison is based on changed or extended concepts in the artifact. If a user adds a new idea to the artifact, this can result in a new concept. Thus, the new and the old version of the artifact differ in the new concept.

Again consider three versions $v_{1...3}$ of a text. Instead of producing a vector of lines for each version, the system would now produce a vector of concepts c_i for each version i:

$$\begin{array}{lllll} c_1 & = \{human, & interaction, & machine\} & \\ c_2 & = \{human, & interaction, & computer, & mouse\} \\ c_3 & = \{human, & interaction, & computer\} & \end{array}$$

Then the difference sets can be calculated based on the concepts:

$$\begin{aligned} r_{(1,2)} &= \{machine\} \\ a_{(1,2)} &= \{computer, mouse\} \\ r_{(2,3)} &= \{mouse\} \\ a_{(2,3)} &= \{\} \end{aligned}$$

The impact of a version i to the version j is then calculated accordingly and leads to $\{machine, computer\}$.

Known Uses

Expertise Browser (Mockus and Herbsleb 2002)[2] is a tool to support software development teams.

Expertise is determined from the change history of a software artifact. When a user performs a change, this change is recorded in a version management system. Each change is weighted according to the number of lines that were changed by it.

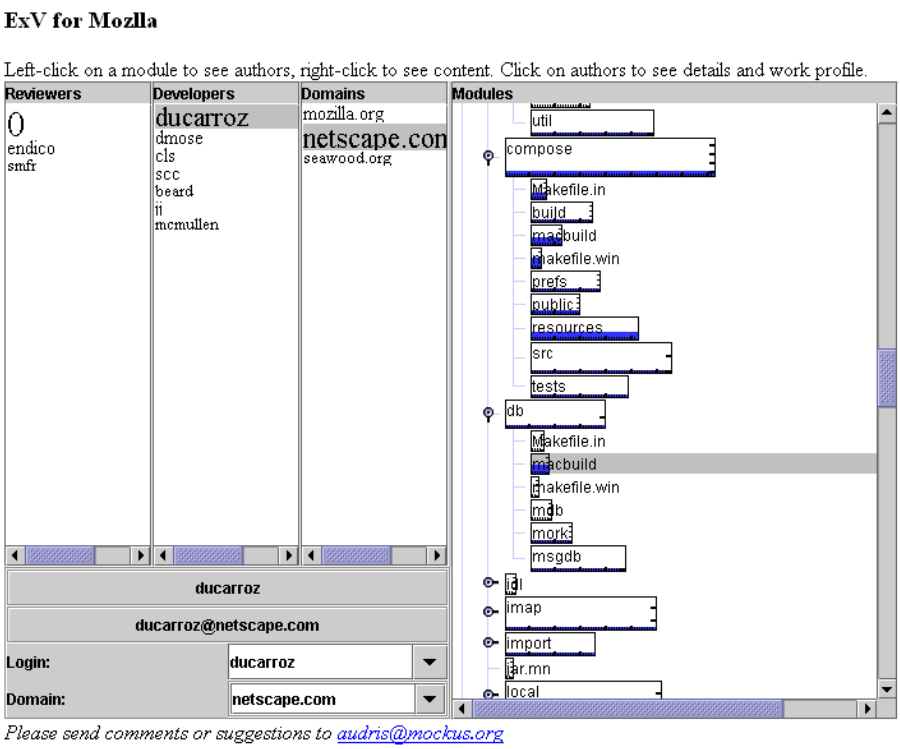


Figure 13: The Expertise Browser

Figure 13 shows the user interface for finding an expert. The user can select an artifact of the project (in the right part of the figure). The system then looks up all changes for the artifact and calculates expertise levels for all participants considering all weighted changes (strategy 3 considering only write activities and weighting the activities according to strategy 4). The result of this query is shown in the left part of fig. 13. The center column shows the contributing users sorted by their expertise. The font size represents the level of expertise. A user can contact the expert by selecting him in the middle column and clicking on his email address in the bottom of the window.

MEMOIR (Pikrakis et al. 1998) collects trails of users' web browsing behavior. When users need to find an expert for a specific topic, the system collects all trails that contain pages

[2] A demo version is available at http://www.research.avayalabs.com/user/audris/ExV/ExV2.html.

on a specific topic. Experts are listed with respect to the quantity of keyword matches in the user's trails.

Related Patterns

ELEPHANT'S BRAIN$_{\rightarrow 3}$ stores all activities performed by users.

SEMANTIC DISTANCE$_{\rightarrow 3}$ can be used to calculate differences between different versions of an artifact.

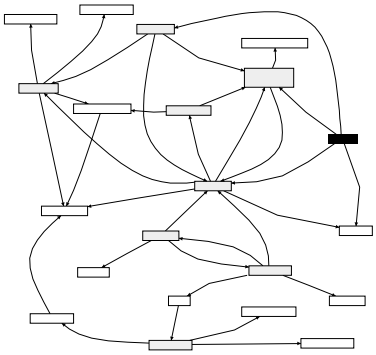
REMEMBER TO FORGET$_{\rightarrow 3}$ removes old activities. This resolves the problem of contacting users who no longer remember their activities and are thus no experts for the artifact.

PRESENCE INDICATOR$_{\rightarrow 3}$ considers only active activities of other users to show who else is currently interacting with an artifact. Although these users may have less expertise than the users recommended by the FIND THE GURU pattern, they have the advantage that they are available and share the same focus as the local user.

HALL OF FAME$_{\rightarrow 2.9}$**:** The FIND THE GURU pattern can be interpreted as a variant f the hall of fame where the focus of the all of fame is restricted to one specific artifact.

2.7 Pay Back

Alternative name(s): compensation, fringe benefit remuneration



Intent

Reimburse the users' services in the community by means of a virtual currency.

Context

Your environment supports users to actively contribute to the community. You started to establish Reciprocity$_{\rightarrow 3}$ to equally distribute benefits and efforts. Now you are thinking about how to better motivate contributors so that they spend more efforts.

Scale: Large communities.

Problem

To keep the users spending efforts for the community, they have to be motivated. But often the efforts are not reciprocal so that users don't see benefits related to their efforts. Thus, participation declines.

Scenario

Imagine a class of computer science students. The students exchange questions regarding a specific course in a newsgroup.

George has never liked computer science and has large problems solving the assignments. Thus, he asks many questions in the newsgroup. Unfortunately, his questions are on a very basic level and therefore not very interesting to answer.

Janet is a different kind of student: she has no problems at all in solving the assignments. Thus, she does not even enter the newsgroup although she could provide answers to George's questions with very low efforts.

Besides the computer science course, the class also hast to work on a course on social impacts of modern technology. This course again has a newsgroup in which the assignments (now sociological questions) can be discussed. Now it is Janet who would need some help since she did not understand the rationale behind ethnographic studies. George in contrast likes these assignments and

would have no problems to help Janet. But as Janet ignored the computer science newsgroups, George now ignores the beginner's questions in the social impacts course.

Although Janet and George could form a great complementary team, they do not see the benefits of helping one another since the topics where they could benefit are too different from the topics where they would have to spend efforts.

Symptoms

You should consider to apply the pattern when ...

- users have diverse capabilities but do not need assistance in areas where they perform well.
- the fact that user A can help user B does not imply that user B can provide any benefit for user A.
- the community needs some users who perform unpopular tasks.

Solution

Therefore: Provide each user with virtual money that can be used to purchase services within the community. Let users earn a specific amount of money, when they positively contribute to the community.

Collaborations

Every user has an account that stores his current balance. Users can act as *service providers* or as *service consumers*. When a service consumer wants to use a service from a service provider, he hast to transfer a fixed or a negotiated amount of money to the service provider.

The community as a whole can act as a service consumer for services that are of common interest. In this case, the service provider is payed by the community, which means that his balance is increased without decreasing any other user's balance.

Rationale

The basic problem resolved by this pattern is that users have different expertise and that users with compatible expertise are hard to find. For that reason, users no longer base their exchange of services on direct exchange but use an intermediary currency.

This principle is as old as the invention of money. In the times when money was invented, it was used to exchange goods with more than one partner, especially people from different tribes. Before the use of money, two people could only exchange goods if both had goods that were of value for the other. Money substituted direct utility of the other person's goods with potential utility.

By using the virtual money, users are eager to provide services to other users since they can accumulate money that can be used when they need a service of other users.

There is an advantage of using a virtual currency instead of real money: The exchange of services is restricted to members of the community. This encourages users to continue interaction within

the community since they cannot spend the money at any other place.

Danger Spots

Ensure that every user can earn money by defining a large range of positive contributions. Otherwise, new members can be impeded to participate.

If it is acceptable that some users do not contribute at all, consider unlimited credit for the users' accounts.

Note that the Pay Back pattern is based on the philosophy of a *homo economicus*, as it was defined by Spranger (1966). According to Spranger, humans can be classified in different types that express their ethics:

theoretic: tries to act objectively.

economic: acts according to the perceived utility of each action.

aesthetic: acts to improve form and harmony.

social: is helpful and emphasizes values like love and loyalty.

political: emphasizes the importance of power.

religious: acts according to the existence of a blessing God.

As one can see in Spranger's classification, there are many other ethics and human values than the perceived utility of each action. A careful analysis of the social structure of the intended user group should therefore check, if economic users are important in the community. If this is the case, the application of Pay Back does make sense.

Users can get addicted to collect virtual money with the effect that all interaction is only performed for receiving the highest possible benefit. The benefits that were initially thought as motivational factors become the reason for interaction. The initial intention of the community (the problem that should be addressed by the community) becomes a secondary issue. Oscar Wilde coined the saying that there was a "price of everything" and a "value of nothing". The price reached for an action moves into the focus while the real value of this action is no longer important.

Monetary incentives should thus always be complemented by other incentives.

Known Uses

Experts-Exchange is an online help forum that rewards users, who provide an answer to a question. Whenever a user states a new question, he has to define how many expert points he will transfer to the user who provides a satisfying answer. The expert points will be taken from the asking user's account immediately and stored for the user who provides an satisfying answer. The price of the question is shown together with the question in the list of open questions and other users can select questions that they want to answer.

Figure 14: Earning expert points at expert-exchange.com.

If an user provides an answer that satisfies the asking user, the answer is formally accepted by the asking user. The asking user can then grade the answer. The better the grade is, the more points will be added to the answering user's account. If the asking user is very satisfied, he can for instance grade the answer with an *A* and the answering user will receive four times the question points.

Figure 14 shows the profile of a user (left) who has provided the accepted answer for a question (right). In the right part, one can see the original question points of the question (250) and the grade that the asking user gave the answering user (B). That means that the answering user received $3 \star 250 = 750$ points for his answer.

eMule is a P2P file sharing system that introduces a credit system for transactions between peers. Whenever a peer acts as a server for another peer, it receives credit points. These credit points can be used when the peer is acting as a client: The more credit points a client has, the better its ranking in the download queue of the requested server will be.

Mojo Nation (no longer available but described by (Leuf) (2002, p. 215)) was another P2P file sharing environment that honors users with a virtual currency called Mojo. New users enter the community with a credit of one million Mojo. Every time a user consumes content in the network, he has to pay some Mojo. The price controls how fast a user can access specific content. It depends on supply and demand of the desired content. Every time a user donates server capacities such as disk space, computing power, or network bandwidth, he earns some Mojos.

dooyoo is a product review community that reimburses reviewers with WebMiles. Reviewers who write a review earn 30 Web-

miles (worth 3 pence) whenever someone reads their review. In addition, members who make it to the HALL OF FAME$_{\rightarrow 2.9}$ receive additional 1000 WebMiles. This encourages users to write more reviews since they have to become well known to be part of the HALL OF FAME (users have to propose other users for the HALL OF FAME).

slashdot.org (Lampe and Resnick 2004) lets users collect *Karma* points for each contribution to the community. Possible contributions are for instance the provision of interesting stories or comments or good moderation. Users who received a high Karma will receive moderation rights. This empowers them to judge on other user's contributions and make them more or less prominent.

A high Karma also helps users to receive higher attention: Users with a high Karma can post their comments with a higher initial rating. This has the effect that their stories will be shown at more prominent places.

Related Patterns

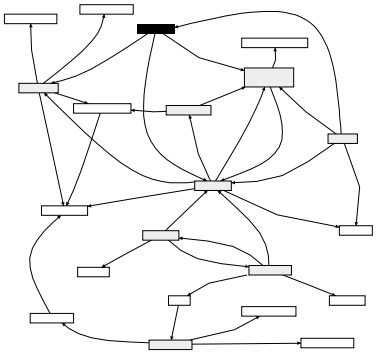
LETTER OF RECOMMENDATION$_{\rightarrow 2.4}$**:** One way for paying back helpful behavior can be to write a LETTER OF RECOMMENDATION.

HALL OF FAME$_{\rightarrow 2.9}$**:** Being listed in the HALL OF FAME can be another incentive for providing help to others.

GROUP AWARD$_{\rightarrow 2.8}$ can be used to avoid fear of losing advantages through cooperation.

HONOR QUESTIONS (Fricke and Völter 2000): In educational contexts, it can be counter-productive if only the answers to questions get rewarded. Instead, you should consider to honor students who raised questions in the community since the question is an important step to understand new concepts.

2.8 Group Award



Intent

Honor all group members for a result achieved by the group.

Context

You have created an environment in which groups can collaborate. Now you are thinking about rewarding the users.

Scale: Users collaborate in small groups. The groups act in the context of a larger community.

Problem

Often, when a result of a collaborative process is valuable, only some of the group members get rewarded for their participation. This results in a lack of motivation for those members who were ignored in the reward.

Scenario

Consider a large factory in which employees should be encouraged to improve the production process. For that reason, the upper management installed an incentive scheme: each inventor of a new idea will receive a fixed percentage of the savings caused by the idea.

To generate more and better ideas, some employees one day had the idea to meet after work and brainstorm on possible improvements together. They invented great ideas and the whole group was proud of the group outcome. But at one day only one group member appeared in press releases presenting the idea and this user was also the only user who was rewarded for the invention. The rest of the group was very angry and learned never to collaborate on preliminary ideas again.

Symptoms

You should consider to apply the pattern when ...

- users are unmotivated because other users harvest the fruits of their work.
- some users are publicly announced while others remain unrecognized.

Solution

Therefore: Ensure that the reward for the result is shared

among all group members. Find a distribution key that is considered as fair for all group members.

Collaborations — Users perform a task together. When it comes to the point where the result of the task is presented to non group members, it is presented as a group result. Rewards and social recognition for the group result is shared equally between all group members.

Rationale — Stenmark (2002) compared different systems for idea generation and found out that systems that offer rewards decrease the willingness of sharing ideas. Users instead fear that their ideas are elaborated by other users and that these other users then get rewarded. Stenmark thus proposed that users should be rewarded collectively.

This approach makes it easy for the participants to be aware of the rewards that they will receive after the task. Ideally, all group members will build up responsibilities for the group result and start collaboration rather than competition.

Danger Spots — As Stenmark (2000) argues, group awards can strengthen group coherence (which is in many cases not a bad thing). Anyhow, in cases where inter-group exchange is desired, strong group cohesion may block exactly this exchange.

The fact that the group agreed on sharing the group result equally, some users may stop feeling responsible for the group result if they see that other users perform in a satisfying way. The main problem in this case is that different group members have different expectations for the group result. Group members with low expectations may start to act as free riders.

One possibility to cope with this problem is that the group receives a group award and individual group members receive additional rewards for outstanding contributions (according to the Pay Back pattern).

Not everything should be rewarded. Instead it is important to create a good balance between activities performed for the group and activities that are reimbursed by an award.

Ensure that the award has an intrinsic value that is desired by the group members. The value does not have to be a material value. Instead it often is a social approval of the goals reached by the group.

Known Uses — **Collaborative Exercises** at the FernUniversität in Hagen (Haake et al. 2003): In these exercises, students met for 1 to 2 hours and worked on an exercise. They discussed their different contributions and arranged them in one group result. Each participant received the same number of points for participating in the exercise.

SourceForge.net is a community site for open source projects.

Every month, one project receives a special attention and is presented as the project of the month (figure 15).

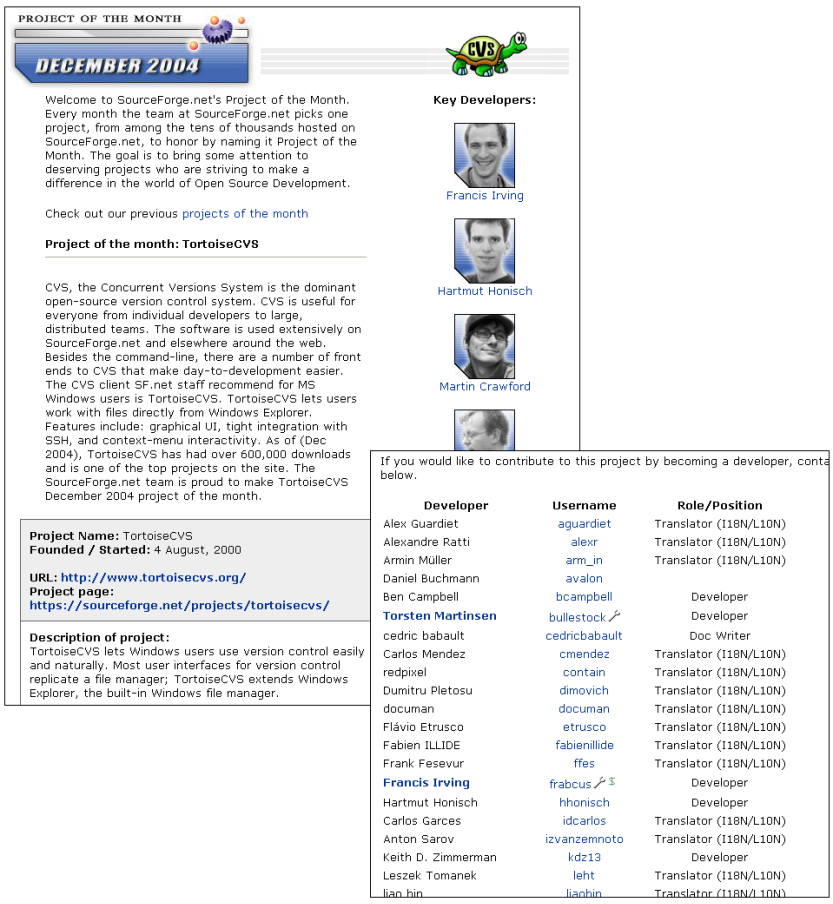


Figure 15: The project of the month at www.sourceforge.net.

Instead of only introducing the project led, sourceforge lists all key developers on the *project of the month* page. These are the developers who made the largest contributions in the group. Other members are listed on the project's detail page. The project itself is always shown as a group result. Individual contributions are not especially marked (except comments in the source code that help to find the developers of a specific component).

Mindpool (Stenmark 2001) is a system that combines asynchronous brainstorming with an idea proposal system. Although it does not implement a GROUP AWARD pattern, the author observed the problem of the pattern during a system evaluation, which was already mentioned in the *rationale* section. He then proposed to add a collective rewarding scheme.

Related Patterns Combinations of the following patterns with GROUP AWARD are

possible to distinguish between free riders and active users.

One Grade For All (Eckstein et al. 2002) also proposes to have only one grade for all members of a group in an educational context. All group members receive a grade based on a presentation given at the end of the project. Group Award transfers this pedagogical pattern to any group process.

Fair Team Grading (Eckstein et al. 2002) discusses the relation between individual rewards and group rewards in the context of education. It suggests to base parts of the grade on the team product and other parts on individual contributions.

The Group Award pattern is in contrast to patterns that focus on honoring individual members of the community. These are for instance:

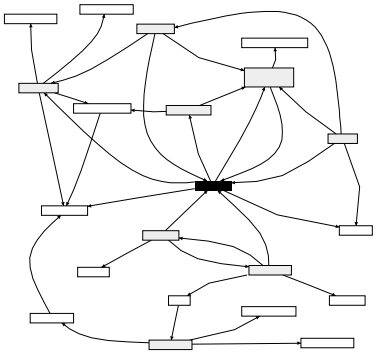
Hall of Fame$_{\rightarrow 2.9}$ presents famous individuals that contributed to the community.

Find the Guru$_{\rightarrow 2.6}$ provides means for identifying extremely knowledgable users.

Letter of Recommendation$_{\rightarrow 2.4}$ for certifying positive behavior to individual group members.

2.9 Hall of Fame

Photo: Keller, Hannes, Niederglatt,
www.visipix.com



Intent

Honor the most helpful participants in the system by showing them in a Hall of Fame.

Context

Your community system allows users to contribute with their capabilities to the community. Now, you want to encourage active users to be even more active.

Scale: Groups of all sizes.

Problem

The motivation for participation in a community is related to the feedback that participants receive from the community. But often very active participants are not enough recognized by the community members. This decreases the motivation of the active participants to continue their efforts.

Scenario

This is the story of a 4th grade school class that went on a school trip: The whole class stayed at a youth hostel with rooms for approx. 6 students. And – as it is at least common for German students – all rooms were messed up within the first hour of the stay. But instead of telling the children off, the teacher decided to honor students who helped to make the hostel a friendly and tidy place during the stay.

Students received points for well cleaned wardrobes or for cleaning the dishes. Those students with the highest points were publicly announced and received an award at the end of the trip.

During the whole stay, the students started to see the task of tidying up their hostel as a competition. The public recognition of most eager students played an important role in this process.

Symptoms

You should consider to apply the pattern when ...

– the number of passive participants is much higher than the

number of active participants and the ratio between benefits and efforts is better for passive participants than for active participants. This means that RECIPROCITY$_{\rightarrow 3}$ is violated.

- participation is not equally distributed, which results in frustration for very active participants.
- participation decreases over time although the subject is still interesting for the participants.

Solution

Therefore: Provide a list of those participants who participated most. Calculate the participants's participation level with respect to the degree that the participants helped others. Let each participant compare himself to those participants shown in the Hall of Fame.

Collaborations

A participant contributes to the community. A participant can either be a member of the community or a group of users (cf. GROUP AWARD$_{\rightarrow 2.8}$). The participant's contributions are rated by other users or by means of a contribution metric. A contribution metric can for instance be the number of posts in a discussion board.

Participants who received the highest ratings are listed in a *Hall of Fame*. All users can browse the Hall of Fame and perceive information about the participants with high ratings. They can also see how much more the other participants contributed by comparing their personal rating or position in the HALL OF FAME with other participants' ratings.

Ensure that the Hall of Fame is positioned at a prominent place in the community.

Rationale

The Hall of Fame adds a competitive element to the community. Only participants who contribute to the community will reach a place in the Hall of Fame. If participants stop contributing, they will be outrun by other participants (assuming that at least a small set of participants contributes).

How well the Hall of Fame works depends greatly on the community's culture. If the community members are eager to know who is the best participant in the community, it will also encourage participants to outrun other participants in the hall of fame. This helps to convert relatively passive participants into active participants.

Danger Spots

Depending on the differences of fame between a newcomer and a participant shown in the hall of fame, it can be demotivating for new users to work for being included in the Hall of Fame. One solution for that problem can be to ensure that fame vanishes over time. Otherwise, participants may have become famous (because they participated a lot in the past) but then stopped participating

in the community. For the same reason, newcomers will have to invest very much before they will be recognized in the hall of fame if fame does not vanish.

A vanishing fame can for instance be achieved by dividing the perceived value of the contribution by the elapsed time that has passed since the contribution.

Other alternatives for motivating both newcomers and old stagers are provided in the related patterns section.

As discussed in the ACTIVITY COUNTER$_{\rightarrow 2.5}$, participants can try to trick the metric to reach a better position in the HALL OF FAME. Consider, for example, a HALL OF FAME for software development projects, as it is modelled at *freshmeat.net* presented in the known uses section. Since the vitality of a project is measured by counting contributions to the project's discussion board, project members can pretend a high vitality by generating noise (meaningless messages) in the discussion board.

Known Uses

software-engineer.org is a community web site for software developers. To encourage participation, the designers grouped activities regarding their helpfulness to others. Participants can earn 50 points by submitting job offers or links, 100 points by submitting news or downloads and 150 points by submitting an article to the site.

Figure 16 shows the top contributors to the site. Besides the detailed list of contributors (in the middle of the figure) it shows the top-five contributors on the introduction page (and in the right part of fig. 16).

For registering in the community, the users receive 10 initial points. Most of the 8723 participants (95.5%) never actively provided content from then on. 2.7% earned a maximum of 100 points and 1.5% earned between 100 and 1000 points. Only 0.2% of the users earned more than 1000 points. Thus, we can see a negative logarithmic correlation between the number of users and the earned points as shown in figure 17. One can see that the community is mainly passive.

As pointed out in the safety rules, one problem with the model of collecting points can be that users will never lose the points. Thus, the longer a user participates, the more points he will earn and keep the points regardless the fact that he might no longer be active in the community.

freshmeat.net hosts an open-source community and lists the most popular and vital projects. Popularity and vitality is calculated as explained in the pattern ACTIVITY COUNTER$_{\rightarrow 2.5}$. Besides the total popularity, it also shows those projects that improved their popularity within the last month (shown as *projects on the horizon* in figure 18). This solves the problem seen in the previous example that newcomers would have to

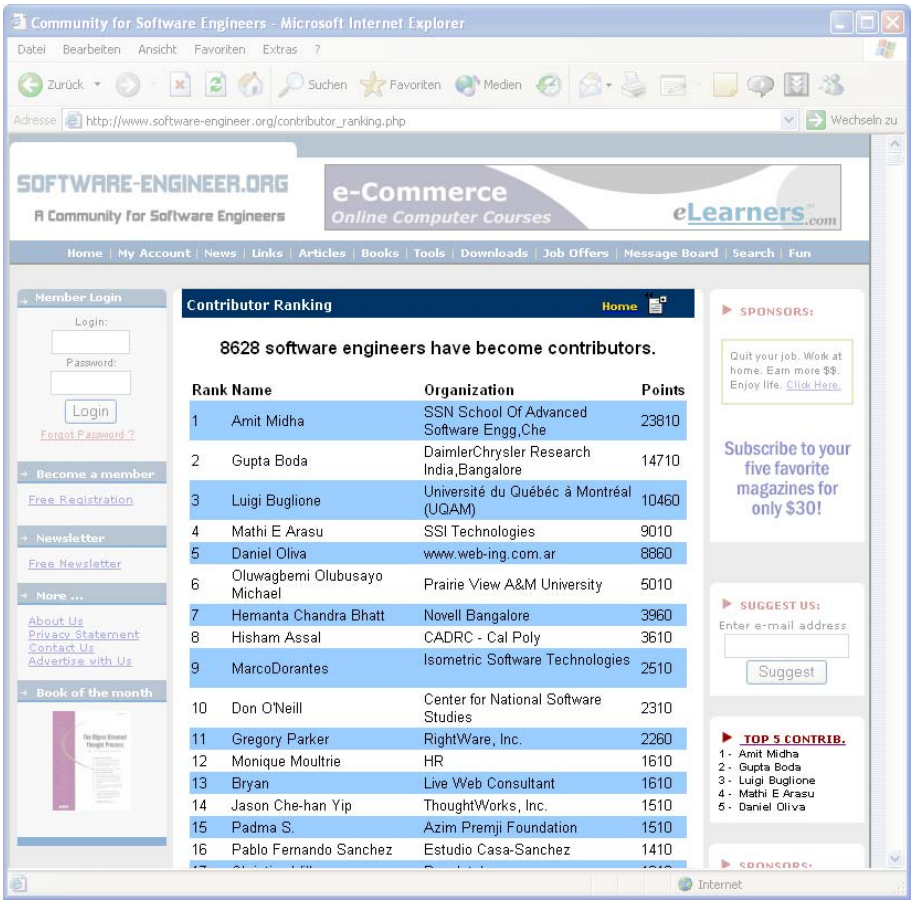


Figure 16: Hall of Fame of the software-engineer.org community.

invest very much before they are recognized by the community. By showing the tendency of the participation, active newcomers will be honored in this section of the hall of fame.

amazon.com lists all *Top Reviewers* as shown in figure 19.[3] Unlike the the previous examples, the reviewers are not ranked according to the number of reviews they wrote but they are ranked using the number of positive feedbacks of other users regarding their review. Thus, the contribution metric relates closer to the LETTER OF RECOMMENDATION$_{\rightarrow 2.4}$ than to the ACTIVITY COUNTER$_{\rightarrow 2.5}$.

CiteSeer (Lawrence et al. 1999) maintains a list of most cited authors in the area of computer science (http://citeseer.ist.psu.edu/cs). In this case, the value of an author's contribution is measured by the number of other authors who mention this author in a scientific paper. Each citation of a paper adds a point to the cited author's rating. The rating thus measures, how many other works were influenced by the author's contribution. Those authors with the

[3] http://www.amazon.com/exec/obidos/tg/cm/top-reviewers-list/-/1/

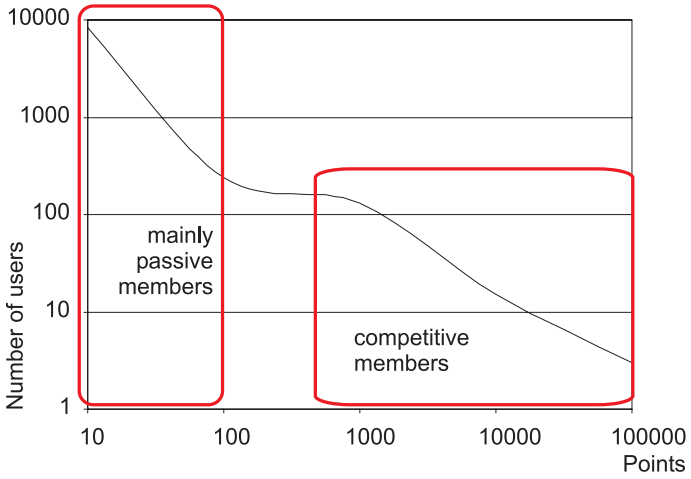


Figure 17: Participation of users in the software-engineer.org community.

highest total ratings are shown at the top of the Hall of Fame.

A problem with the CiteSeer metric is that only those works found by CiteSeer are considered for computing the impact of a contribution. The community in this case is not bound to a specific media or interaction platform (as CiteSeer) and thus interacts by various means. This means that CiteSeer only captures a subset of the interaction with the effect that the Hall of Fame does not reflect objective fame in the community.

Related Patterns

Find the Guru$_{\rightarrow 2.6}$ reduces the focus of the calculation of prominent users to the interaction with one specific artifact. The guru for an artifact is the first in the artifact's Hall of Fame.

Group Award$_{\rightarrow 2.8}$ can be used to emphasize important groups instead of individuals. One example was the *freshmeat.net* community in the known uses section, where projects were listed instead of single users (developers).

Letter of Recommendation$_{\rightarrow 2.4}$ can be used to determine how much a user helped other users. A user with many letters of recommendation will receive a better position in the Hall of Fame.

Activity Counter$_{\rightarrow 2.5}$ provides means for ranking artifacts regarding their popularity. Instead of listing users with good recommendations in a Hall of Fame, one can also list artifacts regarding the number of activities that took place with them. This leads to a list of most important artifacts, which can ease the newcomer's orientation.

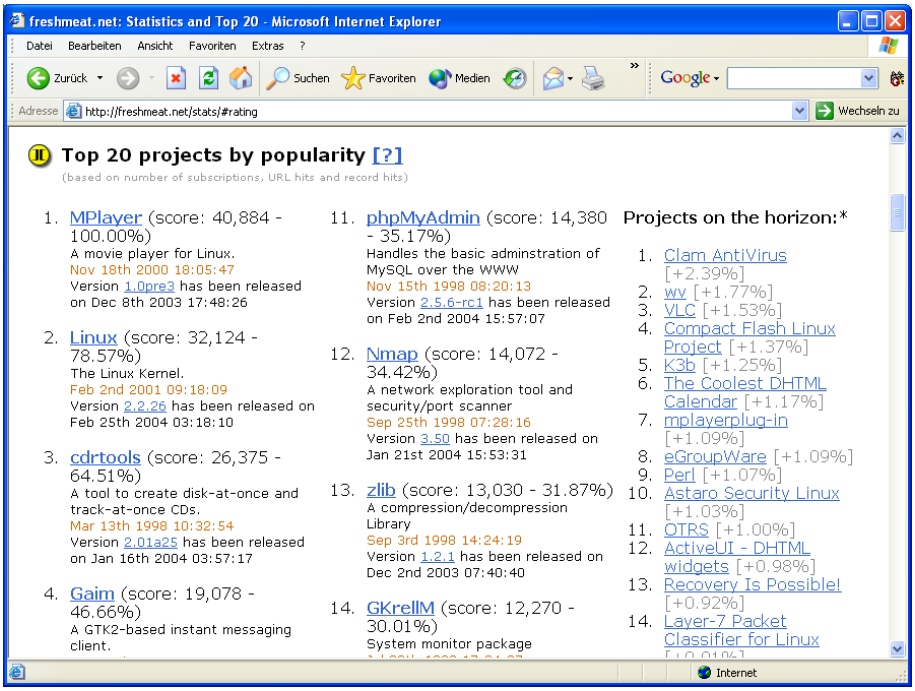


Figure 18: Hall of Fame in freshmeat.net.

ELEPHANT'S BRAIN$_{\rightarrow 3}$ can be used to keep track of users' activities. But in most cases it is not needed to keep track of all activities for calculating an expert rating. Instead, users can collect the points with each activity and thus always know their current expert rating.

Rank	Reviewer
1	**Harriet Klausner** **Total reviews written: 6297** **#1** *Reviewer* I was an acquisitions librarian in Pennsylvania and wrote a monthy review column of recommended reads. I found I liked reviewing and went on to freelance after my son was born. I have 2 dogs, a cairn and a pom, and four cats. Oh, I have a 21 year old... more
2	**Lawrance M. Bernabo** **Total reviews written: 7151** **TOP 10** *Reviewer* To review, or not to review: that is the question: Whether 'tis better to post reviews and cover The pros and cons of action figures, Or to write reviews about best sellers, And by reviewing diss them? To critique: to review; ... more 
3	**Don Mitchell** **Total reviews written: 2137** **TOP 10** *Reviewer* Thank you to the readers who have voted for my reviews and made me Amazon's #1 most helpful reviewer of nonfiction books! Many thanks also to the 123 reviewers who have praised The Ultimate Competitive 

Figure 19: Top reviewers at amazon.com.

Most cited authors in Computer Science -

Generated from documents in the CiteSeer database. This list does not
the citing and cited articles match, or citations where the relevant autho
multiple authors (e.g. J. Smith). This list is automatically generated and
from search results because this list is generated in batch mode wherea
659481 authors were found. Homepages listed may not be for the mos
corresponds to multiple authors. Click on HPSearch to see and update

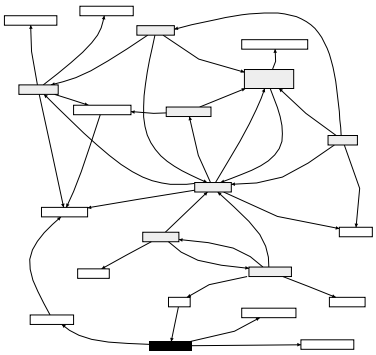
Next 250

1. D. Johnson [2] [3] [4] [5] ... (HPSearch): 12119
2. J. Ullman (HPSearch): 11041
3. A. Gupta [2] [3] [4] [5] ... (HPSearch): 8407
4. R. Milner (HPSearch): 7900
5. R. Rivest [2] (HPSearch): 6930
6. M. Garey (HPSearch): 6732
7. R. Tarjan (HPSearch): 6525
8. J. Dongarra (HPSearch): 6522
9. V. Jacobson (HPSearch): 6494
10. L. Lamport [2] (HPSearch): 6410
11. J. Smith [2] [3] [4] [5] ... (HPSearch): 6052

Figure 20: Most cited authors at CiteSeer.

2.10 Travel Together

Alternative name(s): Shared or Collaborative Browsing



Intent

Explore an information space together with a team mate.

Context

Users of your application have a different knowledge about the data (or the virtual environment) that is presented in the application. Now you are thinking about ways to ease the orientation in the environment.

Scale: The group interacts in a large interaction space (of shared artifacts). This space as well as the group is large enough so that group members do not necessarily know the content of the space and activities of other users within the space. In large spaces, this may even be true for loosely coupled interaction between two users.

Problem

When finding their way through an unknown environment, users can often get lost.

Scenario

In many history books, you can read that the first man who reached the north pole was Robert Edwin Peary. On April 9th, 1909, he wrote in his diary: "The Pole at last!!! The dream prize of 3 centuries, my dream and ambition for 23 yeas. Mine at last."

But was he alone? No, he was accompanied by Matthew Henson and four Inuit men named Ootah, Seegloo, Egingway, and Ooqueah. These six people started together on an exhibition to a space that was never reached before.

No one has seen the environment before. It is a real exploration. And as it is the case for many explorations, it has the potential danger that individual members could get lost. Travelling together (and staying together) reduces this risk since the orientation task is performed by more than one group member.

Symptoms

You should consider to apply the pattern when ...

– users need a long time to find the information that they are looking for.
– different users have different previous knowledge of the information environment. This leads to a situation where the users have different orientation skills in specific parts of the collaborative environment and users with bad skills get lost.
– the goal is to find the information as a group but several group members spend duplicate efforts to reach this goal.
– navigation demands creative decisions of selecting the right trails but single users do not have enough creative ideas, where to search for the desired information.

Solution

Therefore: Browse through the information space together. Provide means for communication and collaborative browsers that show the same information at each client's site.

Collaborations

Two or more users open a collaborative browser on the same information artifact. A user can move on to another artifact. This has the effect that the collaborative browsers of the other users are informed with the address of the new artifact. The other browsers may then also move to the new artifact.

It is important to decide how the navigation takes place. This includes that the floor control mechanism is adapted to the users' needs. Examples for different floor control strategies in collaborative browsing are:

Master Slave Browsing where one user is driving and the other users follow. This method of collaborative browsing is suitable in situations where newcomers should be guided through the information space by an expert.

Anarchistic Browsing that does not propose any roles. Whenever one user moves to a new location, all other users follow. This is suitable when a group is seeking for information together and all group members have about the same knowledge of the information space.

Democratic Browsing: In this method of collaborative browsing, the group has to form a collaborative opinion first before the group members move on to the next artifact.

Rationale

In a study of traditional libraries, Twidale et al. (1997) showed that browsing should be a collaborative action. Although many search for information is carried out alone in these places, the authors could show that interaction between users takes place.

Since all browsers are always showing the same artifacts, the users will be able to communicate about the shown content. This

helps them to better understand the artifacts. When a user navigates to another artifact, all other users follow, which ensures that the group remains focussed on the same artifact.

By discussing the route, the team will choose the most appropriate path. And since many users travelled together, there is a better history of the steps taken.

Danger Spots

Since all browsers are coupled, the group will always travel at the same pace. This can be a problem if the comprehension speed for the browsed artifacts differs significantly. In this case, fast users can feel obstructed by the slower users.

Known Uses

TUKAN (Schümmer 2001) is a collaborative software development environment. It informs the programmers of the presence of other programmers to support dynamic group formation. After users have met, they can navigate through the software system using a tightly coupled browser (cf. figure 21). The selection of the class, the protocol, and the method are shared by all users, thus they always see the same method. Within the method, the users can read independently (the scroll position of the text pane is not coupled).

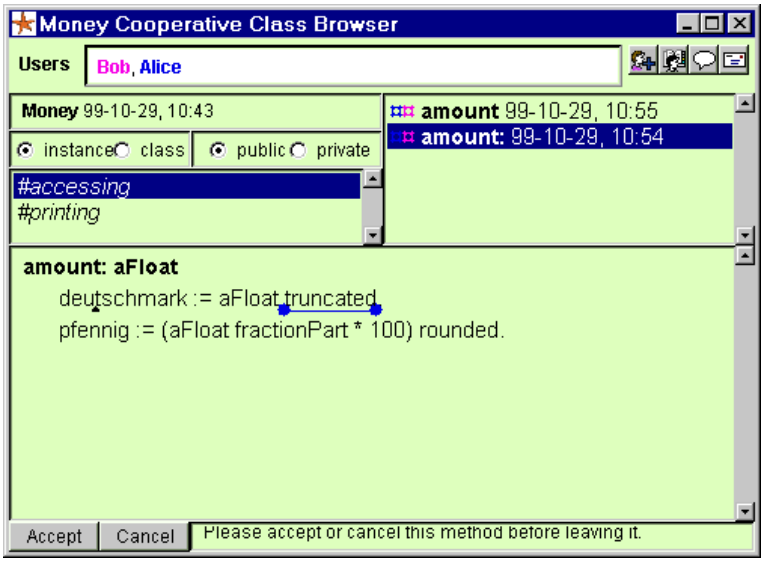


Figure 21: The Collaborative Class Browser in TUKAN.

During the collaborative exploration of the source code, the developers can communicate about the shown code fragments using an integrated chat tool.

CobWeb (Stotts et al. 1998) allows a group of users to browse web pages together. It uses two frames in a standard web browser: the content is shown in one frame of the browser while the other frame is used for controlling the browsing.

This frame also includes means for requesting the floor (requesting to driv).

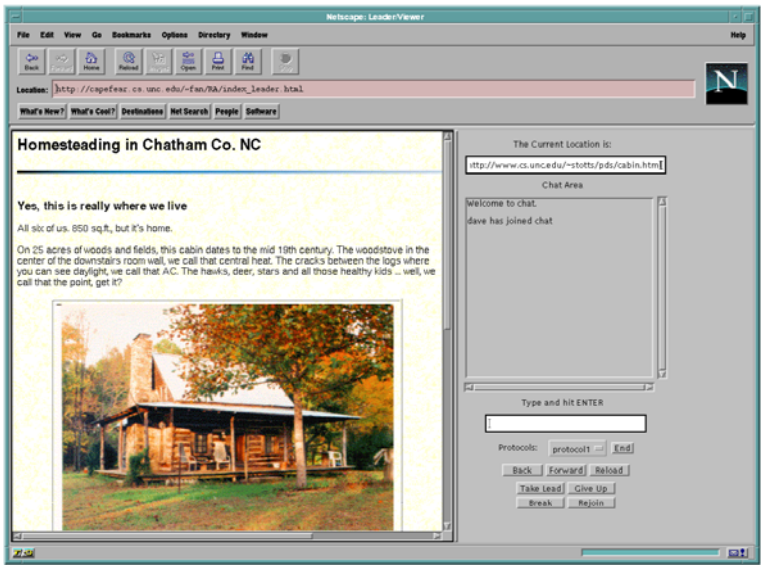


Figure 22: Collaborative Web Browsing in CobWeb (Stotts et al. 1998).

One special feature of CobWeb is that it allows the system developer to model interaction models for collaborative browsing by means of a petri net. This approach provides flexible means for designing all interaction processes discussed above.

Comparable to the CobWeb system are GroupScape (Graham 1997) and CoWeb (Jacobs et al. 1996). Both systems also support collaboratibe synchronous browsing of the web.

Related Patterns

Application Sharing$_{\rightarrow 3}$ is the general concept of using an application together with another user and seeing exactly the same application state at each point in time.

Swarm and Collect$_{\rightarrow 3}$ is comparable to collaborative browsing with respect to the fact that a group wants to explore a collaborative information space together. The difference is that each group member explores parts of the space individually and then shares the relevant results in the group.

Tell Me a Story$_{\rightarrow 3}$ uses records of other users' browsing activities to simulate a collaborative browsing activity. The main differences are that only one user defines the path and that this user is not co-present.

3 Additional Thumbnails

All patterns that were not described in a prosaic way will be presented as thumbnails in this section. The patterns are listed in alphabetic order.

Active Neighbors

Problem: The Local Awareness$_{\rightarrow 3}$ pattern only signals confocal users on the same artifact. If users work on related artifacts, they are not aware of each other, which implies that no collaboration will be established. On the other hand, especially collaboration on relate topics can support creative processes and mutual learning.

Solution: Provide awareness on peripheral activities that take place on related artifacts. Use a Semantic Distance$_{\rightarrow 3}$ to show, how relevant those activity are. Rate activities on artifacts with a short semantic distance more important than activities with a long semantic distance. Ensure that activities on related artifacts do not distract the user's attention too much from the focused artifact.

Application Sharing

Problem: You want to collaborate using a problem-specific application but this application does not support synchronous use of many users.

Solution: Use an application sharing system that replicates the view and the controller of the application to the users' machines.

Attention Screen

Problem: Every request for attention needs to be processed by the user. Thus, it already takes some of his attention. But in situations, where the user needs to focus his attention on other things, this is disturbing.

Solution: Enable the user to filter the information which reaches him. Use meta-information (e.g. sender details) or content information (e.g. important keywords) to distinguish important information from not so relevant information. Collect the less important information at a place where the user can process it on demand and forward relevant information directly to the user.

Birds of a Feather

Problem: If people don't know one another, it is hard to decide who could be a good partner for a collaborative activity. For co-located situations, humans have developed intuitive strategies (based e.g. on visual clues) that help them to select, whom they should contact if group formation is needed. In distributed work environments, the presence of other users is often reduced to their user name. This makes it hard to find another user for a collaborative activity.

Solution: Compare user profiles or interaction histories to identify two users who share big parts of their history. Propose these users as collaboration partners.

Buddy List

Problem: When many users are able to interact in the interaction space, it is

hard to maintain an overview of relevant interaction partners since the number of users exceeds the number of relevant contacts for a specific user. User lists grow very large and it is hard to find people who the local user knows. On the other hand, the local user is often only interested in those people who he knows.

Solution: Provide buddy lists, where a user can enter other users who are of interest to him. Whenever the local user browses other users, first show only users from the buddy list.

Collaborative Filtering

Problem: To recommend activities, the system has to predict a user's behavior. In cases where the user has recommended subjects (e.g. by a Letter of Recommendation$_{\rightarrow 2.4}$), the system can assume that the user will act in a comparable way with related subjects. But this calculation restricts the scope of recommendations to objects within a Semantic Distance$_{\rightarrow 3}$ of the user's experience. Recommendations will thus not introduce the user to interesting subjects of semantically unrelated topics. But the user may like these topics.

Solution: Apply collaborative filtering: Find users who made comparable recommendations and scan their set of recommendations for subjects that are not in the local user's current focus.

Color-Coded Distances

Problem: You want to distinguish important information that is related to the user's current focus from information that is less relevant.

Solution: Use color-coding to distinguish near events from distant events. Use warning colors (e.g. red) for events that should catch the user's attention and comfortable colors (like green, black, or blue) for events that occurred further away.

Don't Disturb

Problem: To allow spontaneous collaboration, users have to be open for contact requests. But each contact request disturbs the user in his current task.

Solution: Include a virtual don't disturb sign in the application that signals that the user should not be disturbed.

Elephant's Brain

Problem: Merging two user's (past or current) work is a difficult task. It requires that the activities are transferred to the same context and that the goals are aligned. But many applications don't provide access to the artifact's history, its use, and its evolution. Thus, merging is vulnerable to errors and often collaboration does not take place since the merging efforts exceeds the estimated gains of a collaboration.

Solution: Remember all activities that users perform on shared artifacts – not only modifying accesses, but also read accesses. Provide access to the activities, so that a user can understand (and merge) other users' activities with his own activities.

In-Place Awareness View

Problem: You want to provide awareness information that relates to specific

artifacts. But explaining this relation explicitly puts an extra burden on the user.

Solution: Place the awareness information next to the artifact to which it refers.

Local Awareness

Problem: Although most systems that work on shared data provide support for coordinating shared access, they often don't tell the user, who is working on a specific artifact. Such information is needed to establish ad-hoc teams that share a common focus. Without such information, users assume to work alone – and do not see the possibility or urge for collaboration.

Solution: Provide awareness in context. This means that the system tells the local user, who else is currently interested in the local user's focussed artifact and what they do with this artifact. Show this information whenever the artifact is shown on the screen. The information should contain details about the user drawn from his user profile, the artifact, and details on the activity, which the user is performing. Ensure that the information is always valid.

Masquerade

Problem: User monitoring is required for providing awareness information to remote users or associating work results with a specific user. On the other hand users often do not act as confident if they know that they are monitored as they would act in an anonymous environment. Especially openness and the courage for taking risks may be much lower.

Solution: Let the user control how much interaction information he provides to the system. This means that the user should be able to filter the information, which is revealed from his personal information. Remember to consider Reciprocity$_{\rightarrow 3}$.

Mentor

Problem: Newcomers have problems orienting in an unfamiliar community. Especially, they don't know how community members normally act in specific situations.

Solution: Pair the newcomer with an experiences group member. Initially let the newcomer observe the veteran and gradually shift control to the newcomer.

Presence Indicator

Problem: The In-Place Awareness View$_{\rightarrow 3}$ makes it easy to connect other users' activities with focussed artifacts. But the surrounding of the artifact only provides limited space for information. Awareness information thus competes with application data.

Solution: Limit the size of the awareness information's representation so that it uses only a small part of the available information channels. For a GUI system, this means that you should represent the confocal or peripheral users as a single icon instead of a long textual form. Focus on telling that there *are* other users, rather than providing much information on the other users' identity or task. Ensure that the indicator differs from the other artifacts representing application data.

Reciprocity

Problem: It is easy to agree on participation, if the goal is beneficious for everyone. But in many work situations, some people benefit more than others from a reached goal. In the extreme case, the beneficiaries of the reached goal do not have to participate in the group efforts at all. This leads to a situation, where the people who have to spend efforts on the group result no longer see the need to participate since the results are not valuable for them.

Solution: Establish reciprocity. Ensure that all group members' activities result in an improved group result that is beneficial for all group members again. Prohibit people to benefit from group results if they are not willing to help the group in return.

Rehabilitation

Problem: User ratings always reflect the impression of a user at one specific point in time (and in a specific context). When reading the ratings, users expect that the rated user will act in a comparable way in comparable interaction contexts. But users may change.

Solution: Provide a voting workflow (Vote$_{\rightarrow_3}$) to allow a user to adapt his user's rating (or letter of recommendation) to his current behavior. Ensure that many users, especially users with negative views and users with positive views, are involved in this workflow.

Remember to Forget

Problem: You are using the Elephant's Brain$_{\rightarrow_3}$ and Time Compressor$_{\rightarrow_3}$ to provide asynchronous Local Awareness$_{\rightarrow_3}$. This ensures that users leave their traces. The traces stay persistent to allow future reference to a user's activity. At some point of time, you detect that some activities are no longer remembered by the user, but still displayed as asynchronous context awareness. The users will no longer be able to understand, why they relate to a specific artifact, since they forgot that the have ever seen this artifact. Thus, references to these activities do no longer encourage communication or collaboration, but confuse the parties.

Solution: Remove remembered activities after a point in time, when an average user will no longer remember this activity and the activity is no longer important for the application. Consider the type and length of interaction as factors that influence, how long an activity is remembered.

Semantic Distance

Problem: Your Semantic Net$_{\rightarrow_3}$ is very dense in a sense that artifacts have a semantic relation to many other artifacts. But not all artifacts have the same importance for the user. If the user sees only the semantic net, he might get lost in the diversity of relations.

Solution: Use weighted edges to describe the strength of the semantic relation. Interpret these edges as distances. If two artifacts are semantically strong related, ensure that the connecting edge in the Semantic Net represents a short distance.

Semantic Net

Problem: Detecting short semantic distances between artifacts based on a similarity measure often leads to ineffective and inexact results. It is time consuming, when there are many artifacts with large distances because this would involve much unnecessary computation. In addition it fails, if two artifacts are related by means of an intermediate artifact.

Solution: Produce a semantic net that contains artifacts and relations between artifacts. Relate two artifacts, if they have much in common (as in the Semantic Distance$_{\rightarrow_3}$ pattern). Define the distance between two artifacts as the length of the shortest path between these artifacts.

Spatial Domain Model

Problem: Users of a system using a Semantic Net$_{\rightarrow_3}$ have difficulties to understand what it means to work with a specific artifact. Especially, they have difficulties to understand that one object has a strong semantic connection to another object. On the other hand it is essential that the user understands these relations to cope with Active Neighbors$_{\rightarrow_3}$.

Solution: Use the metaphor of a virtual space, which can be inhabited by the application's users. Provide easy means to inspect two related objects in a sequence. Describe the change of focus from one object to another with terms that are commonly used to describe movements in the real world.

Swarm and Collect

Problem: Your group has to find information in a large information space. For each individual group member, this would take more time than he has. Thus, the whole information space cannot be scanned individually.

Solution: Let the users swarm out in the information space and collect relevant information in a shared workspace. At the end of the session let them discuss and consolidate the found information.

Tell Me a Story

Problem: If users have not participated in the evolution of an artifact in a collaborative information space, they may have difficulties in understanding the artifact's current state.

Solution: Replay the other users' activities in a way that the user sees how the artifact evolved. Show old versions of the artifacts together with a description of what the users did. Include both navigational and modifying activities.

Time Compressor

Problem: You are using Local Awareness$_{\rightarrow_3}$ to inform the local user of activities of his colleagues. The underlying Spatial Domain Model$_{\rightarrow_3}$ is large and the number of users is relatively small. Thus, tow users will not likely be at the same place at the same time. Thus, the spatial domain model will look as if it was empty.

Solution: Collect and combine activities that took place at the same artifact

at different points in time. Sow the distance in time between two usages of an artifact (e.g. by using Color-Coded Distances$_{\rightarrow 3}$). Let the user scale and limit temporal distances.

Train the Recommender

Problem: Recommendations may be wrong because the recommender worked with incomplete or incorrect user profiles.

Solution: Whenever a recommendation is made let the user provide feedback whether or not the recommendation made sense for the user. If not adapt the data that led to the recommendation.

Vote

Problem: Users need to be aware of other user's attitudes if they want to interact. But User Profiles cannot provide an answer to all attitudes that a user has. And even if the attitudes are part of the User Profile, it is still hard to figure out the distribution of opinions in the community.

Solution: Present the community members with or let them organize polls on controversial questions in the context of the community's topic. Show a virtual ballot on a prominent place in the community. After the user has voted provide him with the result.

4 Conclusions and Acknowledgements

This paper provided a set of patterns for establishing contacts and encouraging participation in community systems. Since one often finds a divide between the newcomer's knowledge of the community and veterans' experiences, it is crucial to develop mechanisms to bring these two fractions together. In cases where this works, the potential of newcomers with fresh ideas can be used, which in many cases leads to prospering communities.

One example for such a community is the EuroPLoP community. Newcomers are brought together with experts, the shepherds, who invest large efforts to share their experience with the newcomers. In this context, I'd like to thank my shepherd, Joe Bergin, who significantly helped me to improve the patterns of this paper.

References

Armour, P. G. (2004). Beware of counting loc. *Commun. ACM 47*(3), 21–24.

Carotenuto, L., W. Etienne, M. Fontaine, J. Friedman, M. Muller, H. Newberg, M. Simpson, J. Slusher, and K. Stevenson (1999). Communityspace: Toward flexible support for voluntary knowledge communities. In *online proceedings of "CHANGING PLACES - the workshop on workspace models for collaboration"*, London, pp. available online at http://www.dcs.qmw.ac.uk/research/distrib/Mushroom/workshop/final–papers/lotus.pdf.

Chapman, R. (2001). Community continuum - building online communities on the web.
http://www.informationhighways.net/mag/mprevious/00apr01.html

Eckstein, J., J. Bergin, and H. Sharp (2002). Feedback patterns. In A. O'Callaghan, J. Eckstein, and C. Schwanninger (Eds.), *Proceedings of the Seventh European Conference on Pattern Languages of Programs (EuroPLoP'02)*, Konstanz, Germany, pp. 343–373. UVK.

Fricke, A. and M. Völter (2000). Seminars – a pedagogical pattern language about teaching seminars effectively. In M. Devos and A. Rüpping (Eds.), *Proceedings of the Fifth European Conference on Pattern Languages of Programs (EuroPLoP'2000)*, Konstanz, Germany, pp. 87–128. UVK.

Girgensohn, A. and A. Lee (2002). Making web sites be places for social interaction. In *Proceedings of the 2002 ACM conference on Computer supported cooperative work*, pp. 136–145. ACM Press.

Graham, T. C. N. (1997, July). Groupscape: Integrating synchronous groupware and the world wide web. In S. Howard, J. Hammond, and G. Lindgaard (Eds.), *Human-Computer Interaction, INTERACT '97, IFIP TC13 Interantional Conference on Human-Computer Interaction*, Volume 96 of *IFIP Conference Proceedings*, Sydney, Australia, pp. 547–554. Chapman & Hall.

Haake, J. M., T. Schümmer, and A. Haake (2003). Supporting collaborative exercises for distance education. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track1*, Big Island, Hawaii, pp. 31.2. IEEE Computer Society.

Jacobs, S., M. Gebhardt, S. Kethers, and W. Rzasa (1996). Filling html forms simultaneously: Coweb – architecture and functionality. In *5th International WWW Conference*, Volume 28, pp. 1385–1395. Elsevier.

Lampe, C. and P. Resnick (2004). Slash(dot) and burn: distributed moderation in a large online conversation space. pp. 543–550.

Lawrence, S., C. L. Giles, and K. Bollacker (1999). Digital libraries and autonomous citation indexing. *IEEE Computer 32*(6), 67–71.

Leuf, B. (2002). *Peer to Peer: Collaboration and Sharing over the Internet*. Addison-Wesley Longman Publishing Co., Inc.

Manns, M. L. and L. Rising (2004). *Fearless Change: Patterns for Introducing New Ideas*. Addison-Wesley.

Marathe, J. (1999). Creating community online.
http://www.durlacher.com/research/resrepdetail21.asp

McDonald, D. W. and M. S. Ackerman (1998). Just talk to me: A field study of expertise location. In *Proceedings of Computer Supported Cooperative Work*, pp. 315–324.

Mockus, A. and J. D. Herbsleb (2002). Expertise browser: A quantitative approach to identifying expertise. In *Proceedings of the 2002 International Conference on Software Engineering*, pp. ALR–2002–001.

Pfister, H.-R., C. Schuckmann, J. Beck-Wilson, and M. Wessner (1998). The metaphor of virtual rooms in the cooperative learning environment clear. In N. Streitz, S. Konomi, and H. Burkhardt (Eds.), *Cooperative Buildings - Integrating Information, Organization and Architecture. Proceedings of CoBuild'98*, Volume 1370 of *LNCS*, Heidelberg, pp. 107–113. Springer.

Pikrakis, A., T. Bitsikas, S. Sfakianakis, D. D. Roure, W. Hall, S. Reich, and G. Hill (1998). MEMOIR: Software agents for finding similar users by trails. In H. S. Nwana and D. T. Ndumu (Eds.), *Proceedings of the 3rd International Conference on the Practical Applications of Agents and Multi-Agent Systems (PAAM-98)*, London, UK, pp. 453–466.

Preece, J. (2000). *Online Communities*. Chichester, UK: Wiley.

Schümmer, T. (2001). Lost and found in software space. In *Proceedings of the 34th Hawaii International Conference on System Sciences (HICSS-34), Collaboration Systems and Technology*, Maui, HI. IEEE-Press.

Spranger, E. (1966). *Lebensformen – geisteswissenschaftliche Psychologie und Ethik der Persönlichkeit* (9 ed.). Halle, Germany: Niemeyer.

Stenmark, D. (2000). The role of intrinsic motivation when managing creative work. In *Proceedings of ICMIT*, Singapore. IEEE Press.

Stenmark, D. (2001). The mindpool hybrid: A new angle on ebs and suggestion systems. In *Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS-34)-Volume 1*, pp. 1037. IEEE Computer Society.

Stenmark, D. (2002). Group cohesiveness and extrinsic motivation in virtual groups: Lessons from an action case study of electronic brainstorming. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)-Volume 1*, pp. 16.1. IEEE Computer Society.

Stotts, D., J. Prins, L. Nyland, and T. Fan (1998). Cobweb: Tailorable, analyzable rules for collaborative web use. Technical report, Dept. of Computer Science, University of North Carolina, Chapel Hill.

Telewest (2001). Online communities can keep your custommers coming back for more.
`http://www.telewest.co.uk/yourbusiness/newsandinformationtechnology10.html`

Twidale, M. M., D. M. Nichols, and C. D. Paice (1997). Browsing is a collaborative process. *Information Processing and Management 33*(6), 761–783.

Vogiatzis, D., A. Tzanavari, S. Retalis, P. Avgeriou, and A. Papasalouros (2005, July). The learner's mirror. In *EuroPLoP'04 – Proceedings of the ninth European Conference on Pattern Languages of Programs*, Isree, Germany. UVK.

Ward, P. I., N. Mosberger, C. Kistler, and O. Fischer (1998, December). The relationship between popularity and body size in zoo animals. *Conservation Biology 12*(6), 1408–1411.

Yiman, D. (2000). Expert finding systems for organizations: Domain analysis and the demoir approach. In *Beyond Knowledge Management: Sharing Expertise*, Boston, MA. MIT Press.