Technische Universität Wien

M A G I S T E R A R B E I T

MOTIVATE ONLINE COMMUNITY CONTRIBUTIONS USING SOCIAL REWARDING TECHNIQUES

A FOCUS ON WIKI SYSTEMS

Ausgeführt am Institut für

SOFTWARETECHNIK UND INTERAKTIVE SYSTEME

der Technischen Universität Wien

unter der Anleitung von

AO.UNIV.-PROF. MAG. DR. SILVIA MIKSCH
DIPL.-ING. DR. WOLFGANG AIGNER

durch

BERNHARD HOISL

NEUSTIFTGASSE 66/2/2
1070 WIEN

2. September 2007

Kurzfassung

Online-Communities haben eines gemeinsam: Ihr Erfolg steht und fällt mit der Teilnahmefrequenz ihrer Benutzer. Für viele Online-Communities ist daher der Benutzermangel ein fundamentales Problem.
In dieser Arbeit wird auf bestimmte Methoden der sozialen Belohnung eingegangen, welche darauf abzielen, Benutzer zu motivieren, in einem Wiki aktiv teilzunehmen. In einer Online-Community basiert soziales Belohnen in den meisten Fällen auf das Hervorheben besonders aktiver Mitglieder. Geld kann dabei nicht als Motivator eingesetzt werden, sondern andere Faktoren wie Status, Macht, Ehre und Ruhm sind entscheidend. Es werden verschiedene Methoden der sozialen Belohnung vorgestellt, die die Zufriedenstellung dieser Bedürfnisse als Ziel haben und somit Vorteile für alle aktiven Benutzer eines Wikis bringen sollen.
Weiters sind vier Methoden sozialer Belohnung in die Wiki Software MediaWiki implementiert worden. Diese vier Techniken beziehen sich auf benutzte Referenzen eines Artikels, Artikelbewertungen, Artikelbesuche und Benutzerempfehlungen. Unter anderem sind dabei die Kriterien sozialen Belohnens durch die Darstellung einer Rangliste von besonders aktiven Autoren erfüllt. Es wird dabei auf den der Berechnung zugrunde liegenden Algorithmus, auf den Implementierungsprozess und auf die Informationsvisualisierung der Ergebnisse der Entwicklung eingegangen.

Abstract

Online communities have something in common: their success rise and fall with the participation rate of active users.
This thesis focuses on social rewarding mechanisms that generate benefits for users in order to achieve a higher contribution rate in a wiki system. In an online community, social rewarding is in the majority of cases based on accentuation of the most active members. As money cannot be used as a motivating factor others, such as status, power, acceptance, and glory have to be employed. Different social rewarding mechanisms are explained which aim to meet these needs of users.
Furthermore, four methods were implemented within the MediaWiki system, where social rewarding criteria are satisfied by generating a ranking of the most active members. These techniques refer to used references in an article, user votes, article visits, and user recommendations. In addition, this paper also focuses on the calculation algorithm the ranking of authors is based on, the implementation process of the development, and the use of adequate information visualization techniques to present results.

Download the SocialRewarding MediaWiki extension at:

http://www.mediawiki.org/wiki/Extension:SocialRewarding

To Christina for her patience.

Contents

1 Introduction

1.1 Document Conventions

Throughout this thesis the following conventions are used:

- Important terms are *italicized* on first use or, for any reason, if it is wanted that they are emphasized.
- All code fragments and file names are formatted in `typewriter style`.
- Names of functions are marked with trailing parentheses, for example: `function()`.
- References are made by section numbers rather than names, for example: Chapter 1.4 on page 11.
- Figures, tables, footnotes, equations, and files are numbered throughout the whole paper – not per chapter.
- References to chapters, appendixes, figures, tables, and equations are made with page numbers or with the adjunction *on the following/preceding page*, except the chapter, appendix, figure, table, or equation in question is on the same page as the reference.

This document has been typeset using LaTeX2e in the MiKTeX 2.4 distribution with e-TeX 3.141592-2.2 engine.

1.2 Definition of Terms

This chapter defines and explains the most important terms used in this paper:

- *Social reward* – "Social reward refers to something that causes a behavior to increase in intensity. There is nothing that is intrinsically rewarding. A thing is called rewarding if, when it is applied, it results in the intensification of behavior" [Abo07]. Social rewards have a non-monetary basis and are, for example, friendship, peer reputation, prestige, and external feedback.
- *Social rewarding mechanism* – Technique to reward people contributing to an online community[1]. Within an online community social rewarding is in the majority of cases based on accentuation of the most active members. For example, ranking of users' activities ("top contributors").

[1] In this paper social rewarding mechanism, social rewarding technique, and social rewarding method are used as synonyms.

- *Online community* – An online community (or virtual community or web community) organizes and brings together individuals, groups, and businesses on the Internet – rather than face to face – around common interests or purposes [Mal02, 734].

- *Wiki system* – A wiki is a web-based software that allows all viewers of a web-page to change the content by editing the page online by means of a web-browser. This makes the wiki a simple and easy-to-use platform for cooperative work on texts and hypertexts. The first wiki, with the name "WikiWikiWeb", was developed in 1995 by Ward Cunningham.

  "Wikiwiki" is a Hawaiian word that means "quick" or "hurry". The name stands for the programming characteristic of wiki software in which content can be made available in a quick and uncomplicated manner [Ebe05, 10f].

- *MediaWiki* – MediaWiki[2] is the particular wiki engine developed for and used by many projects like Wikipedia[3]. It is freely available and there are many copies in use by all sorts of projects around the world [Wik07a]. The MediaWiki software is written in PHP and uses MySQL as database management system[4].

- *Information visualization* – "Information visualization aims to produce graphical representations of abstract information structure for human users. [...] The goal of information visualization is to reveal patterns, trends, and other new insight into phenomenon" [Ger06, 19f].

1.3 Preface

Wikipedia – the most famous free encyclopedia – has grown to the biggest wiki community site where hundreds of thousands of users all around the world post and edit articles in many different languages. The tremendous contribution rate on Wikipedia has led to many problems, such as wrong information, copyright violations, or user misbehavior, for example, spammers or trolls [San07]. Other online communities beside Wikipedia have massive troubles motivating users to participate actively. This paper presents techniques in which the fundamental problem of online communities – reaching a critical mass of active users – are addressed.

[2] `http://www.mediawiki.org`, retrieved on 1 September 2007.

[3] Wikipedia is a multilingual web-based free-content encyclopedia. It is written collaboratively by volunteers, allowing articles to be added or changed by anyone with an Internet connection [Wik07l]. The Wikipedia front page can be reached at `http://www.wikipedia.org` (retrieved on 1 September 2007).

[4] For more information on PHP and MySQL it is referred to `http://www.php.net` (retrieved on 1 September 2007) and `http://www.mysql.com` (retrieved on 1 September 2007).

On the one hand, Wikipedia has the problem that published information is not checked for its accuracy and legality by a formal process of reviewing. There has to be a large and heavily involved community which is cross-checking and proofing information for its correctness voluntarily. However, the operators of Wikipedia have not only a social but also a legal responsibility to publish only correct and faultless information to assure their creditability. On the other hand, many online communities have troubles motivating enough users to form an active community. Participation of members is the key factor for a successful online community, and that is why appropriate motivating factors are essential.

As information provided over the Internet is treated like public goods, problems such as free riding[5] or social loafing[6] arise. In Wikipedia users are not charged in proportion to their use, that is why it appears rational for people to view articles without contributing anything on their own. If an economic point of view is assumed, it can be said that a user has costs by publishing an article on Wikipedia (e.g., information acquisition and presentation costs or Internet connection costs) and therefore she/he wants something in return. Extending the benefit for a user so that it exceeds her/his costs is a good starting point to increase participation. This paper focuses on methods to motivate users to participate actively in an online community by making use of a number of different social rewarding techniques.

1.4 Research Goals

Goals of this master thesis are the analysis and evaluation of different social rewarding mechanisms and their implementation in the software system MediaWiki. First, an analysis of online communities in general and the behavior of their users is done as well as the developed social network structures and public good problems. It is tried to give an answer to the question why users want to contribute to an online community (especially to a wiki system). A basic question is the motivation of users: “How can users be motivated to participate actively in an online community by applying social rewarding techniques?”. Performed studies are based on related books, articles, and on self-experience.

[5] In this case, free riding means that a user shoulders less than a fair share of the costs of the whole information production of a wiki [Cor96]. If everybody contributes the same value of information to a wiki, nobody rides free. One of the biggest problems is that the value an individual has from an information resource is very subjective and hard to determine.

[6] Social loafing is the phenomenon that persons make less effort to achieve a goal when they work in a group than when they work alone [Jac85]. As the least articles in Wikipedia (like in nearly every other wiki) are written by only one user but in a team, the problem of social loafing is likely to occur. The answer to social loafing are motivational factors which are partly solved in the MediaWiki software by displaying which sections of an article belongs to which author. Thus, a contribution is linked to an author’s name and can therefore be evaluated.

Following the presentation of different social rewarding methods, the most suitable (which are the best to solve the given problems) are chosen. They are implemented into a MediaWiki test environment with detailed instructions and documentation described in this paper.

A special focus is set on presenting the retrieved data in a suitable graphical way (visualization of information). That is why there is an extra chapter dealing with information visualization, regarding the former described social rewarding mechanisms.

Above all, the following questions arise:

- Why do users want to contribute to an online community, what is their benefit and what are related problems? How can users be motivated to participate in an online community?
- Which social rewarding methods can be useful for an online community in general and in particular to a wiki system?
- How can the chosen social rewarding mechanisms be optimized by using adequate information visualization techniques? Can a surplus be generated and for whom?
- How is it possible to implement the chosen social rewarding mechanisms into a MediaWiki system (along with the use of appropriate information visualization techniques)?

To aim at the previously defined objectives, the following research question is answered in detail throughout this paper: *Which psychological and technical design should social rewarding methods have to be successfully implemented into a wiki system (along with the use of appropriate information visualization techniques) with the aim of increasing community's contribution rate?*

1.5 Related Work

This thesis is part of the following well-known fields of research: social computing, online communities, and human computer interaction. That is why, there are numerous books and articles about the wiki phenomenon (e.g., [Ebe05], [Leu01], [McF06], [Tap06]). However, most of these focus on technical details, such as installing and running a wiki or the revisioning system and its vantages for collaborative information development. Unfortunately, insufficient attention is paid to investigating users' behavior in online communities. Some research is done to explain the problem of free riding ([Ada00] and [Fel04]) which is likely to occur in times of the Internet and shared information platforms. There are also studies about

communication activities of users in virtual communities [Sch03], but their focus is not on motivational factors for users of online communities.

The question which aspects are motivating for a human being, was already discussed by Abraham Maslow in his hierarchy of human needs theory [Mas87]. In an article about social psychology which is used to motivate contributions to online communities [Lin04] an experiment took place where the problems of under-contribution and social loafing were addressed. In the article, as predicted by theory, individuals contributed when they were reminded of their uniqueness and when they were given specific and challenging objectives. As other predictions were disconfirmed, the results of the experiment had to be interpreted carefully. In another article [Lud04] the same co-authors tried to manipulate two factors to increase participation in online communities: on the one hand, the factor *similarity* shows how similar contributions of group members were, on the other hand, *uniqueness* described how unique member contributions were within the group. As a result, both factors influenced participation positively.

Described methods to increase the participation of users in a wiki are based on accentuation and reputation [Res00]. By motivating many users, it is wanted to increase the community so that cross-checking takes place and wrong information is automatically sorted out. The proof that such an approach of member-maintained communities increases the quantity and quality of contributions was affirmed [Cos05] and empirically tested on Wikipedia [Che06].

Parts of this work were published at the following two conferences: 2nd International Conference on Online Communities and Social Computing (held as part of 12th International Conference on Human-Computer Interaction, HCI International 2007), Beijing, P.R. China, 22-27 July 2007 [Hoi07a] and 2nd Austrian Wikiposium 2006, Vienna, Austria, 25 November 2006 [Hoi07b].

1.6 Structure of this Thesis

Chapter 2 on page 15 deals with online communities in general and with the social behavior of users and the interaction between them. The question “What is the basic motivation of users contributing to an online community?” is answered. Furthermore, related topics, such as the problem of free riding or the public good problem are mentioned.

The next Chapter 3 on page 47 deals with different social rewarding mechanisms, especially covering useful methods for the MediaWiki software. The design of social rewarding techniques is explained as well as their useful application. The out-of-the-box social rewarding mechanisms in the MediaWiki software and applied social rewarding methods in Wikipedia are presented.

Chapter 4 on page 76 discusses necessary elements regarding the field of information visualization in connection with social rewarding mechanisms concerning MediaWiki. This chapter is a short introduction to examples of information visualization methods used in this work.

The implementation of social rewarding techniques in the MediaWiki system is described in Chapter 5 on page 80. First, the technical structure of MediaWiki is going to be outlined. After that, the implementation process of the previously chosen social rewarding mechanisms is described. The implemented information visualization techniques are illustrated before at last the overall result is presented.

The last Chapter 6 on page 107 draws a conclusion and summarizes the initial question that is answered throughout the paper. Furthermore, research results are discussed and an outlook on future work is given.

Moreover, Appendix A on page 111 gives an overview of social rewarding techniques, along with their potentials, problems, and implementation complexities.

In Appendix B on page 122 file `SocialRewardingREADME` is displayed containing important information about installation requirements and the installation procedure of the software developed throughout this thesis.

A detailed sample configuration file (`SocialRewardingConfigDetail.php`) to set up the social rewarding techniques implemented in the MediaWiki software is shown in Appendix C on page 124.

At last, an explanation of all files and directories the social rewarding software consists of is given in Appendix D on page 141.

2 Online Communities and User Motivation

2.1 Introduction

Since at least 1979, when the first Usenet news sharing programs were created, online communities have co-evolved with the growth in computer networking. Today, 28 years later, people share news, information, jokes, music, discussion, pictures, videos, and social support in hundreds of thousands of online communities. People benefit from the presence and activity of others in online communities – from the information and other resources that are provided and the conversations they participate in.

Despite the vibrancy of online communities, large numbers of them fail. In many online groups, participation drops to zero. [But99] found that 50% of social, hobby, and work mailing lists had no traffic over a period of four months. On the popular peer-to-peer music sharing service, Gnutella, 10% of users provide 87% of all the music [Ada00]. In open-source development communities 4% of members account for 50% of answers on a user-to-user help site [Lak03] and 4% of developers contribute 88% of new code and 66% of code fixes [Moc02]. Although not every member needs to contribute for a group to be successful [Non00], groups with a large proportion of non-contributors may have difficulties providing needed services to members.

Motivating contributions, especially contributions to the communal good, is a topic that has received substantial attention in many branches of social sciences. Economists and political scientists have observed that across a wide range of settings, people contribute less than the optimal amount of public goods and consume more than their fair share of common pool resources. Nevertheless, the antisocial behavior is considerably less than theories based on pure short-term self-interest would predict [Led95]. Social psychologists have identified an analogous phenomenon called social loafing ([Kar01] developed the collective effort model to explain social loafing). According to [Kar93] people exert less effort on a collective task than they do on a comparable individual task [Lin04].

While for online communities it is vital to have frequent contributing users, for a wiki it is even more important because there are no (or only a few) editorial entries. That means that the amount of content produced in a wiki depends strongly on the participation rate of its members. It can be said that an online community's reputation increases with the quality and quantity of information provided. Especially for a wiki this can only be guaranteed with a lot of heavily involved users. That is why it is so important to motivate users to frequently contribute to an online community such as a wiki. Therefore, the following chapters deal with different types of online communities and with user motivation.

2.2 From Individuals to Online Communities

Talking about collaboration (as it is the case in an online community) it can be differentiated between three layers of participants:

- *Individual actor* – Works alone and shares her/his ideas with nobody else. No collaboration in any form takes place.
- *Small group* – Key characteristics are a shared group awareness between members of a group and the existence of a group structure (hierarchies, role assignments). Furthermore, the group is structurally determined with repetitive forms of interaction. There is a set of shared norms and values in the group.

  Issues pertaining to small groups are a mutual dependency between the individual and the group, the structure of interactions between members of a group, and the decision finding process in a group.
- *Community* – Communities are considered large-scale "groups" with a shared context, interacting over a shared communication infrastructure, and sharing the intentional principle of voluntary membership.

In this context a wiki can be categorized as a "community" – more precisely an "online community". This thesis focuses on online communities in contrast to real-world communities, pointing out the following differences:

- limited communication capabilities and anonymity (pre-dominance of textual exchange),
- group awareness, and
- spatial and temporal independence [Men05].

In online communities many people "live" together, therefore their actions depend on each other. On the one hand, every individual has a certain influence on the community she/he is living in. On the other hand, the group also influences the individual, which – according to [Men05] – can be summarized as follows:

- *Group size* – An extremely large group has a different influence on individuals than a small group where everyone knows each other.
- *Group roles and hierarchies* – If a group has some sort of roles, an individual is likely to stick to her/his initial role. Roles can be formal or informal. Formal roles are assigned explicitly, informal roles are developed over time without a defined structure.
- *Group setting* – The homogeneity and heterogeneity of groups with respect to value systems, gender, ethnography etc. have a high influence on the behavior of an individual.

- *Spatial distribution of group members* – On the Internet groups with members living around the world are formed. Spatial distributed groups act differently than, for example, a group of members living in the same country.

These criteria can directly be transferred to a wiki community. An author of a wiki behaves differently if a bigger group is against her/him, for example, if she/he is the only one who thinks that an article contains wrong information. The group heterogeneity also plays an important role. In online communities members are likely to be spatially distributed, such as over different countries. So, generally they do not know each other. As a wiki has a plain hierarchy with almost no formal roles, a community member has to develop her/his own role over time. This can be positive because a member can change her/his role whenever she/he wants. But there might also be situations in which no one feels responsible due to a lack of formal role assignment. To sum up it can be said that as a member of a community there are two elementary deterministic influences of individuals on groups: their own behavior and the behavior of fellow members [Men05].

Every community has different types of members. As they all act autonomously, nobody has to stick to the norms of the community. In this context, conformity is defined as the tendency to change perceptions, opinions, or behaviors in ways that are consistent with group norms. The driving forces of conformity and non-conformity are [Men05]:

- *Concept of informal communication* – More intense and frequent communication helps re-aligning non-conforming members.
- *Theory of social impact* – Increasing the number of conforming members reduces the influence of the non-conforming minority (which is an issue of free-riding and social loafing).
- *Concept of effective minorities* – Normative influence occurs when group members conform to the norms of the group (the majority opinion) in order to prevent conflicts and ensure acceptance in the group. The majority influences the minority using normative influence. The majority defines the norms of the group and the minority wants to be accepted, so the minority conforms to the majority's view to ensure group harmony and acceptance.

  In contrast, informational influence happens when the aim is a high-quality decision and the decision is based on arguments, or at the least majority group members are still open to evidence and arguments. A minority cannot use normative influence to change the majority's view as it does not define the norms of the group. The minority can, however, use informational influence to change the majority's opinion [Nem74].

Regarding a wiki it can be said that a good communication between its members is essential for building effective working groups. Therefore, communication resources must be accessible for group members. This includes a sufficient technical

implementation of communication tools. As a wiki controls itself, it is assumed that non-conforming members are a minority which can be handled by the others. Otherwise, the online community happens to fail.

2.3 Types of Online Communities

By looking at the very generic definition of online communities described in Chapter 1.2 on page 10, there are many different types of online communities. Some of them no longer play a major part on the Internet, but there are a few types which are accepted from a majority of users.

The purpose of the community and the needs of the involved group (the users) determine the most suitable kind of community. Internet access, access costs, computer and browser types, geographic and time zone issues all influence the type of community. In addition, one needs to decide if a private or a public community suits one's needs best. If sensitive or private issues are discussed, as it might be the case within an illness-support group, or a business workgroup, it can be wanted to develop a private community. If attraction of new and diverse members and ideas is wanted, a public community is the best opinion.

People come together on the Internet for all kinds of reasons, as, for example, collaborative workgroups, family groups, social rooms, role playing groups, illness support groups, rooms for primarily face-to-face groups, ethnic groups, professional groups, geographically related groups, software support, intellectual discussion groups, special interest groups, creative groups sharing techniques and work [Boe07]. It can be said that for every group mentioned a wiki system can offer an ideal technical infrastructure and has a very high potential to suit its members' needs best.

Consistent with [Wen98] it can be distinguished between two types of online communities:

- *Property-centric concept of communities* – Social networks resulting from fundamental similarities: spatial proximity, shared interests, and group feeling. Online communities based on a wiki system (Wikipedia for example) are of this type.
- *Activity-centric concept of communities* – Informal groups or networks sharing a set of interests or problems, exchanging and co-operating over a longer period of time in order to augment their knowledge and to learn from each other.

According to another classification three typologies of communities can be defined:

- *Community of interest* – "Communities of interest bring together stakeholders [...] to solve a particular problem of common concern. They are characterized by their shared interest in the framing and resolution of a problem. Communities of interest often are [...] temporary: they come together in the context of a specific project and dissolve after the project has ended" [Fis01, 4].

- *Community of purpose* – "are communities of people who are going through the same process or are trying to achieve a similar objective. Such communities serve a functional purpose, smoothing the path of the member for a limited period surrounding a given activity. For example, researching a topic on wikipedia.org, buying a car on autobytel.com, antique collectors on icollector.com or individual investors on fool.com. Members of the community assist each other by sharing experiences, suggesting strategies, and exchanging information on the process in hand" [Wik07c].

- *Community of practice* – "A group of professionals, informally bound to one another through exposure to a common class of problems, common pursuit of solutions, and thereby themselves embodying a store of knowledge" [Man96].

Another categorization of online communities can be done by comparing time and space parameters:

	co-located	**remote**
synchronous	face-to-face interaction	synchronous distributed interaction
asynchronous	asynchronous interaction	asynchronous distributed interaction

Table 1: Time and space categorization of online communities.

Regarding Table 1 contributing to a wiki is definitely a remote and asynchronous task, therefore an asynchronous distributed interaction. Posting content to a wiki can be done from everywhere, the only requirement is an Internet access and a compatible web-browser. Many users work at different or even the same resources asynchronously, which means at different points in time.

Technically speaking there are many possibilities for building an online community and for interacting with others over some sort of network – a so-called distributed system. According to the topic of CSCW some groupware applications are:

- *Email and bulletin board* – "An electronic forum that hosts posted messages

and articles related to a common subject" [Vit06, 20]. For example, web logs ("blogs").

- *Structured Message System* – Exchange of any sort of data by using a structured and self-defined plain text format, like EDI or XML.
- *Video conferences and communication* – Interaction with others takes place by filming oneself and broadcasting the video to all other participating users.
- *Argumentation tools* – A distributed system that helps to underline arguments, which is often achieved by applying diagrams and graphics or other structural information visualization techniques.
- *Meeting rooms* – An (electronic) meeting room, arranged with special IT equipment, for organizing a meeting. Members can attend the meeting in this room or from everywhere else (where a compatible IT infrastructure is installed).
- *Shared work surface* – Two ore more users work on the same resource synchronously. For example, an electronic whiteboard where all participants draw at the same time.
- *Shared PCs and shared window systems* – All users share one single PC or one single window where interaction takes place.
- *Shared editors* – Two or more users write, edit, or delet the same resource. This can be done synchronously or asynchronously, for example, editing an article from Wikipedia.
- *Co-authoring systems* – Similar to shared editors. There are two levels of representation: the document itself and annotation/discussion. Not only is there the possibility to edit the document, but also to discuss prior changes (sometimes by applying roles like author, commentator, reader, etc.). The MediaWiki software supports both levels of representation.
- *Shared diaries* – Two ore more users share one single diary or calender. Questions of interest to be answered are: "Who can see my diary or appointments?" (question of privacy) and "Who can edit my diary/calender?" (question of access control)[Men05].

Wiki systems can be classified as shared editors. Users can work on the same article any time and from anywhere. With an additional feature that allows for the discussion of articles (like in the MediaWiki software), a shared editor can be turned into a co-authoring system.

In case of the MediaWiki system the technical architecture is a centralized one. There is a single copy of the application and the data on one (or clustered) server(s),

reachable at one specific place which is defined by a unique URL. This client-server architecture (visualized in Figure 1) is typical for Internet applications.

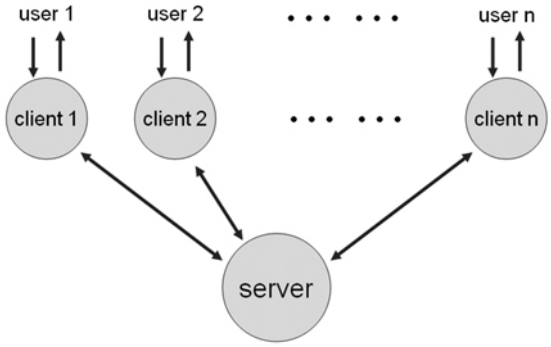

Figure 1: Client-server architecture [Dix03].

As one can see there are n clients connecting to and communicating with a single server (in this case a web-server) in order to retrieve and send information, such as viewing a web-page or editing resources on the server-side. In most cases the client uses a web-browser to view HTML-formatted web-pages, which the server delivers on request by the client. By typing the associated URL of a web-page into the web-browser, a HTTP-GET request is sent to the server and (if successful[7]) a web-page and its references (to images or other multimedia content) is sent back to the client where they are displayed. In a wiki system the client can edit content stored on the server-side (in most cases in a database). Therefore, data has to be sent from the client to the server using the HTTP-POST method.

Figure 2 on the next page displays a categorization of online communities and groupware systems by *communication*, *coordination*, and *collaboration*. A wiki system can be characterized as a hypertext group editor (or shared editor), which is a further development of some sort of bulletin boards. Obviously the main dimension is collaboration along with little influence of communication. A wiki is a system primarily designed for collaborative work and only in a second place for communication between users[8]. The meaning of coordination is almost totally ignored as wiki systems are not designed for coordination. In some cases this can be a nuisance if, for example, two or more users work on different articles with the same textual content. Then a situation can occur within which two articles are published with different names, but having the same content. To avoid such a scenario a large and active community is needed which monitors all changes.

[7]Else HTTP error messages including error message numbers are sent back.

[8]To see how communication works in the Wikipedia project, it is referred to the paragraph at the end of Chapter 2.4 on page 23.

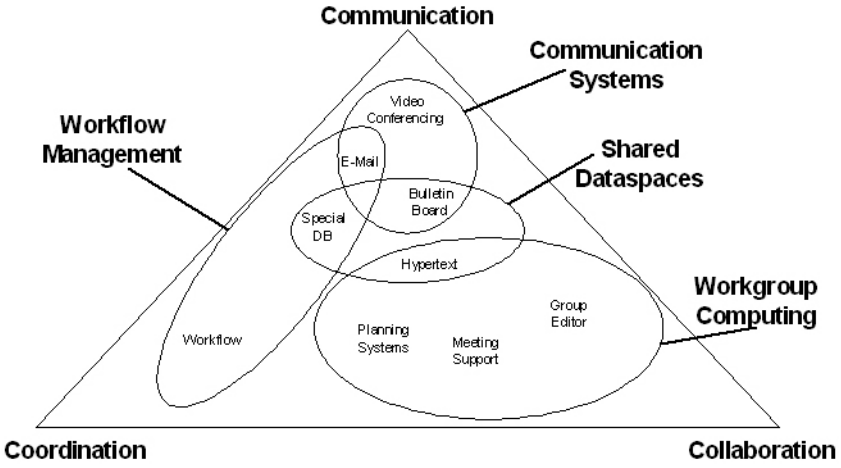

Figure 2: 3C-categorization [Men05].

2.4 Wikipedia as a Co-Authoring System

As already mentioned, the Wikipedia open-source project is based on the MediaWiki software. The project's aim is to create a free encyclopedia in as many languages and with as many articles as possible. As for now, most of the articles in Wikipedia are in English (nearly 2.000.000), German (over 630.000), French (over 550.000), Polish (over 420.000), Japanese (over 400.000), Italian (over 340.000), and Dutch (over 330.000). Overall, the Wikipedia project is available in over 100 different languages, all with a rapidly increasing amount of articles.

Wikipedia is categorized as a co-authoring system, because users can edit the same resource collaboratively (and discuss changes made). The result of this process is a document which has been written by many different users. Of course, a single user can write a document on its own and nobody may edit it just because there is nothing to add, but that behavior is not very common. Most of the time a single user forgets something or makes a mistake and another user corrects the error or adds additional information.

As users can easily edit other users' documents, the former versions should not be deleted. Therefore, the MediaWiki system stores every *revision*[9] of a document and logs every change made[10]. That is why it is very easy to switch back to any prior version of an article at any time. Older versions can be restored if, for example, someone makes a mistake or even completely deletes an article. Therefore, no in-

[9] A version of an article at a specific point in time.

[10] Much like other versioning systems, for example, such as CVS (`http://www.nongnu.org/cvs`, retrieved on 1 September 2007) or Subversion (`http://subversion.tigris.org`, retrieved on 1 September 2007).

formation gets lost and prior changes made can be tracked back until the creation date of an article. Another interesting thing in Wikipedia is that there are only a couple of reviewers and a small group of editorial staff. The correctness of information is mainly checked by other users and as it can be seen, this principle seems to work.

All articles of Wikipedia are published under the GNU Free Documentation License. "The purpose of this License is to make a manual, textbook, or other functional and useful document 'free' in the sense of freedom: To assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others. This License is a kind of 'copyleft', which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software" [Fre07].

Being classified as a co-authoring system, a wiki has to meet some requirements:

- *Fault tolerance* – Difficulties regarding network interactions have to be solved, for example, lost TCP/IP packets (handled by HTTP).
- *Concurrency control* – Every operation must be visible to every user immediately. Wikipedia has implemented an optimistic approach – a version control system[11].
- *Multi-user revocation* – For a shared editor the following requirements must be met: find out the right operation (history-list) as well as find out the right inverse function.
- *Collaboration assistance* – For example, a function for viewing who is online or for viewing who is editing an article right now (not implemented in MediaWiki). The possibility to get in contact with another person editing an article would be a surplus.

Within Wikipedia there are several ways to communicate with others, for example, by posting a public message or by writing a private email[12] to another user. Other ways to get in touch are discussion pages, mailing lists, IRC chats, instant messages, or even face-to-face meetings.

[11] Pessimistic approach: If someone edits a resource, it is locked for others. That means only one user can edit a resource at a particular time. If the changes are committed, the resource is freed again and can be edited by others.

[12] A user can decide whether her/his email address is visible to others and therefore, if other users are able to contact her/him by email.

2.5 Community Member Roles and Types

Every community and online group is different. The purposes vary, the structures and the members are different. But there are some common participation styles or patterns that have been observed. These can be helpful when it is tried to understand participation patterns in an online interaction space. Note that for each style there are attributes that can be seen as both positive and negative. In this chapter only a small amount of community member roles and types are presented. The following roles are the most important regarding a wiki system.

Core participants are usually a small group of people who quickly get used to online interaction and provide a large proportion of an online group's activity. According to [Ada00] 10% of the membership makes up 90% of the community activity. These 10% are important members as they visit the portal frequently and often post content. On the one side, they can be a source of volunteer leadership and provide new ideas for improving the community. On the flip side, one must be careful that they do not control the community and make it hard for less active folks to participate. In a wiki such core participants can have administrator rights, thus keeping the community running.

Readers or lurkers are the unseen forces that *do* affect a community. Following [Non00], community owners estimate that there are approximately 10 to 100 readers per active participant (in Wikipedia there might be much more). They represent a combination of people new to the community, who are not yet comfortable with posting, people who only read and never post content, and people who come in and then drift away without engaging. This group represents a huge pool of potential active members (a group on which it is focussed on with the aim of motivating its members to go a step further and contribute to a wiki). The readers also play another very important role – audience to the active contributors, especially in larger, open, social communities. For commercial communities, which rely on page views to drive advertising revenues, readers are indispensable.

Dominators are people who post frequently and influence the pace of an online interaction space and can, unknowingly and unintentionally, dominate that space making it harder for others to participate. Most often, dominators are not aware of their dominating role. Facilitators can gently ask via email for the member to give others a little more time to respond, while also acknowledging their important contribution. The line between core member and dominator is pretty fuzzy. Dominators can often be given productive roles (e.g., administrator rights) to take advantage of their interest and time, such as volunteer hosts or content experts.

Adjusters are users who do not write new articles in a wiki system, but correct already written ones. On the one hand, there are some users who only correct spelling, grammar, and punctuation mistakes as they cannot tolerate a misspelled word or an

incorrect set comma. On the other hand, there are users who add just a few words or sentences to an already existing article. In some cases the adjusters are at the changeover from a newbie or a lurker to a more integrated member, like a core participant or a dominator.

Linkers, weavers, and pollinators are very important in larger communities where a huge selection of topics is available (as in a wiki). These members tend to participate across a wide range of interests, and in doing so, they are in the best position to let others know of interesting events across the community. They are ideal mentors, and often have interest in acquiring new resources to the community. They keep spaces from getting dull or stale. But they can disrupt slower, deeper conversations with their "flitting" in and out.

Flamers live, as they say, to flame. Flaming is defined as sending hostile, unprovoked messages. What is actually considered a flame varies by community, but often there are people who enjoy challenging other members just for the "fun of it". Name-calling, teasing and such are the tools of flamers. The interesting dynamic of flaming is that to a certain extent, it serves as entertainment for the community and thus attracts new members. On the other end, it drives people away if it exceeds the line of community norms. Flamers can also be the source of new ideas which, when applied within community norms, can be helpful in workgroups and brainstorms. In a wiki it is possible for everyone to revert a version of an article in no time, which prevents most flaming attacks.

Needlers – it only takes one line, repeated, inserted, and insinuated, over time, to recognize a needler. They have a point to make and it appears again, and again, and again (their target are the discussion or talk pages in a wiki). Often using cynical phrases such as "I told you so", needlers know they are right and do not let one forget it. No matter whether the point may be insightful or irrelevant, the audience soon gets bored due to the repetition. This is different to a spammer whose point is often "on point" (thus makes sense), but it can loose its power and context, regardless of the quality. In some cases, this may be from a visionary who is ahead of her/his time and who needles with the interest of the group in mind. Some other times it is from a person who does not budge from their stance. Needlers can also keep community members "honest" by not having a group evade critical issues or behaviors. They can be bellwethers of new ideas.

Newbies are members new to a community. They might also be new to online interaction. When new folks join an online interaction without checking it out, observing the interaction, or learning the community norms, they can be perceived as rude and clueless. In some communities, newbies are forced to a baptism of fire by experienced hands as a way of either being accepted or rejected from the group. Newbies are also the source of new ideas, interest, and "pollination". Newbies deserve more attention and should be supported with information to help them become part of the

group. Presented social rewarding methods try to integrate newbies to become part of a wiki by applying motivational factors.

Spammers post the same thing over and over again. Often it is commercial material with little or no relevance to the community. Sometimes members start spamming as an answer to the feeling of not being "heard". Sometimes it is simply a matter of ignorance of community norms and the general disapproval of spam by experienced Internet users. Additional to the fact that in a wiki the prior versions of an article can be restored in almost no time, it is also possible to block spammers with the knowledge of their IP address and/or username.

Black and white folks are the people who present immutable positions. They appear to be initially unwilling to accept opinions except for their own. They usually are willing to take the blame for their style (ownership) but get around the responsibility of the impact of their behavior. They only engage in their own terms, but may refuse to engage others who utilize the same tactics. Interactions often escalates with the aim of winning. They also are keepers of important information that the community may need, but not particularly likes. They ask the tough questions, but may not like to be asked them back in return. As they are not really good team players, there is little space for this sort of community members in a wiki.

Shades of grey folks are sometimes characterized as "wishy-washy", with no clear convictions, and as members who shrink away from the tough issues. Often they do not fully engage or justify their positions. On the other side, they can often help to neutralize a polarized situation and offer new, combined viewpoints for a community. They tend to carry new information into a group that has polarized on issues such as a breath of fresh air. They are good for writing articles in a neutral point of view, but never act in collusion with someone. Therefore, within a discussion of an article they have no clear opinion.

Untouchable elders is an archetype tended to thrust on others – for example, the expert, the guru – and sometimes unconsciously create a different set of rules or norms for the elder. Most often, the elder does not seek this recognition. Elders may not be accountable for the same community norms or scrutiny of the other members. Elders can dominate new members by the use of solely a few words, regardless of the opinion of others. Their wisdom is important to a community, but their influence can inadvertently muzzle the rest of the group who might feel uncomfortable posting in such company [Whi07]. As collaboration of all users is very important in a wiki, the strength of the elders is weakened, although they certainly have a strong position. A helpful way can be to grant elders more rights in order to have them switch from a normal user to one who engages in managing the system.

2.6 Social Networks

2.6.1 Social Network Theory

Social network theory shows social relationships in terms of nodes and ties (Figure 3). Nodes are the individual actors within the network and ties are the relationships between the actors. There can be many kinds of ties between the nodes. The network can also be used to determine the social capital of individual actors. These concepts are often displayed in a social network diagram, where nodes are the points and ties are the lines.

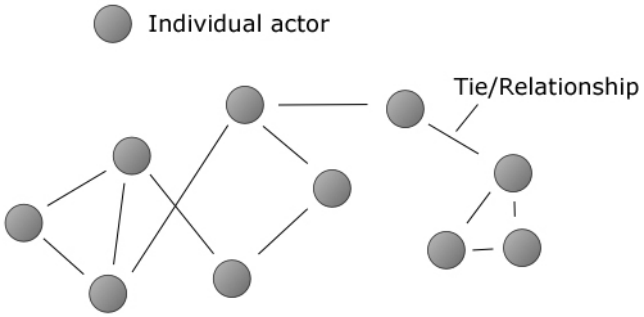

Figure 3: Social network [Wik07g].

The shape of the social network helps to determine a network's usefulness for its individuals. Smaller, tighter networks can be less useful to their members than networks with lots of loose connections (weak ties) to individuals outside the main network. More "open" networks with many weak ties and social connections are more likely to introduce new ideas and opportunities to their members than closed networks with many redundant ties. In other words, a group of friends who only does things with each other already share the same knowledge and opportunities. A group of individuals with connections to other social worlds is likely to have access to a wider range of information. It is better for one's individual success to have connections to a variety of networks rather than many connections within a single network. Similarly, individuals can exercise influence or act as brokers within their social networks by bridging two networks that are not directly linked.

The power of social network theory results from its difference to the traditional sociological studies which assume that the attributes of individual actors – whether they are friendly or unfriendly, smart or dumb, etc. – is what matters. Social network theory produces an alternative view, where the attributes of individuals are less important than their relationships and ties with other actors within the network. This approach has turned out to be useful for explaining many real-world phenomena, but leaves less room for individual agency and for the ability for individuals to influence their success.

Although there was evidence of social networking on the web in 1997 with web-sites such as SixDegrees.com, it was not until 2001 that web-sites using the *Circle of Friends* technique (explanation in the following paragraph) started appearing. This form of social networking, widely used in virtual communities, became particularly popular in 2003 and flourished with the advent of a web-site called Friendster[13]. There are many social networking sites, though Friendster is one of the most successful at using the Circle of Friends technique. The popularity of these sites rapidly grew, and major companies such as Google and Yahoo have entered the Internet social networking space.

In the online communities described above the Circle of Friends technique is used, where an initial set of founders sends out messages inviting members of their own personal networks to join the site. New members repeat the process, increasing the total number of members and links in the network. Sites also offer features such as automatic address book updates, viewable profiles, the ability to form new links through "introduction services", and other forms of social online connections [Wik07i].

These Cirecle of Friends techniques of virtual communities demonstrate the small world phenomenon. The small world phenomenon (also known as the small world effect) is the hypothesis that everyone in the world can be reached through a short chain of social acquaintances. The concept was introduced after a small world experiment by psychologist Stanley Milgram which found that two random US citizens were connected by an average of six acquaintances [Mil67]. However, after more than thirty years, its status as a description of heterogeneous social networks still remains an open question [Kle99].

Albert-László Barabási from the Physics Department at the University of Notre Dame was able to find an even simpler model for the emergence of the small world phenomenon [Bar99]. The former model was able to explain the high clustering coefficient and the short average path length of a small world, but it lacked an explanation for another property found in real-world networks such as the Internet: these networks are scale-free (Figure 4 on the following page). In short terms, this means that they contain relatively few highly interconnected super nodes or hubs: The vast majority of nodes are weakly connected, and the connectivity ratio of the nodes remains the same whatever size the network has attained. If a network is scale-free, it also is a small world.

Barabási's scale-free model is strikingly simple, elegant, and intuitive. To produce an artificial scale-free network possessing the small world properties, two basic rules have to be obeyed:

[13] `http://www.friendster.com`, retrieved on 1 September 2007.

1. *Growth* – the network is seeded with a small number of initial nodes. After a certain step in time, a new node is added. This new node is connected to m existing nodes.

2. *Preferential attachment* – the probability of a newly added node being connected to an existing node n depends on the degree of n (number of connections from n to other nodes). The more connections n has, the more likely the new nodes will connect to n.

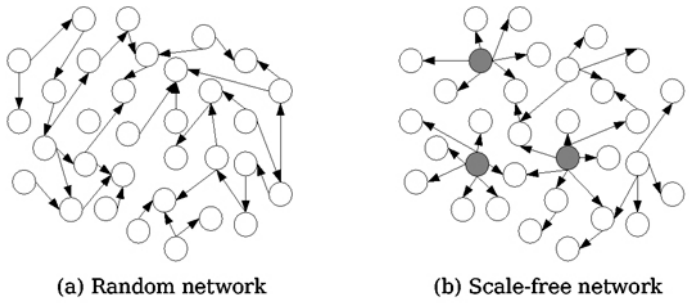

Figure 4: Random network vs. scale-free network [Wik07f].

The same mechanisms apply to the WWW and especially to a wiki. The web and its communities are in a constant state of growth. In a wiki new articles are added every second. If a user creates a new web-page, she/he most likely includes links to other well-known pages (e.g., for further information, references, related topics etc.).

2.6.2 Metcalfe's Law

Bob Metcalfe, founder of 3Com Corporation[14] and the designer of the robust ethernet protocol for computer networks, is known for pointing out that the total value of a communication network grows with the square of the number of devices or people it connects. Since a user cannot connect to itself, the actual calculation is the number of diagonals and sides in an n-gon [Odl05, 2]:

$$\frac{(n(n-1))}{2} \tag{1}$$

Metcalfe's law explains many of the network effects of communication technologies and networks such as the Internet and WWW. The law is often illustrated with the example of fax machines (Figure 5 on the next page): A single fax machine is useless, but the value of every fax machine increases with the total number of fax

[14] `http://www.3com.com`, retrieved on 1 September 2007.

machines in the network, because the total number of people with whom one may send and receive documents grows.

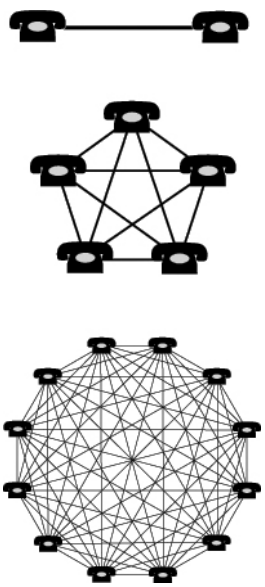

Figure 5: Network effect [Wik07e].

In March 2005, Andrew Odlyzko and Benjamin Tilly published a preliminary paper which concludes Metcalfe's law significantly overestimates the value of adding connections [Odl05]. The rule of thumb becomes: "The value of a network with n members is not n squared, but rather n times the logarithm of n" (Equation 2).

$$n \lg n \tag{2}$$

Their primary justification for this is the idea that not all potential connections in a network are equally valuable. For example, most people call their families a great deal more often than they call strangers in other countries, and so do not derive the full value n from the phone service [Wik07h].

By comparing Metcalfe's law to the Odlyzko-Tilly thesis, one can see that the value of a network grows much faster when calculated following Metcalfe's law (as it is shown in Figure 6 on the next page). This is not surprising, since the Odlyzko-Tilly thesis is a linearithmic function, which grows faster than a linear term but slower than a quadratic term (and Metcalfe's law is an understated quadratic function).

If there is one member in a network (though it can not really be spoken of a network, yet), the value of the network is 0: nobody can be called if no one has a telephone. As the number of members in a network increases, so do the value of the network. In the examples above, at a maximum of 60 members the value of the network calculated with Metcalfe's law is more than seven times greater than with the Odlyzko-Tilly thesis.

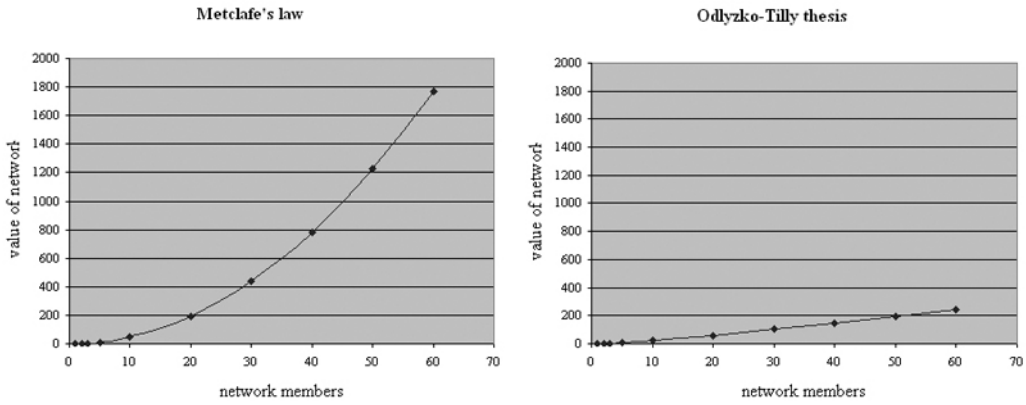

Figure 6: Metcalfe's law vs. Odlyzko-Tilly thesis.

For a wiki it does not really matter which model is the correct one. It has only to be known that the value of an online community increases with the number of active members. Although it should be said that the value for the members of a wiki community certainly tends to increase following Metcalfe's law. It is clear that when there is only one member a wiki does not make sense. As a collaborating tool for big groups a heavy involved community is required to achieve a high value for every individual. By looking at Wikipedia, where an encyclopedia is developed, it certainly can be said that the more users are involved the more it is likely to occur that articles are published about special topics. This is a good fact because Wikipedia is trying to cover nearly every topic in its encyclopedia. So, there is no limit of user participation – every user contributing is generating a surplus.

2.6.3 Network Externalities

The term *externalities* describes situations where a consumption of one person is directly influencing the benefit of another person. The expression *network externalities* is a special form of externalities, where the benefit of one person's good depends on the number of consumers of that particular good. Network externalities are an economic approach similar to Metcalfe's law described in the former chapter. In the sense of an online community this means that the benefit of one person depends on how many other persons are involved in this community.

The amount of connected users of a network (in this case an online community, which is a part of a network – the Internet) is primarily very low and increases step by step at falling costs. When a critical mass has been reached the system increases disproportionately high (Figure 7 on the following page) [Var06, 658ff].

This behavior can also be seen by analyzing users' participation in wikis. A wiki is not very useful and valuable when there is only a little amount of members. At

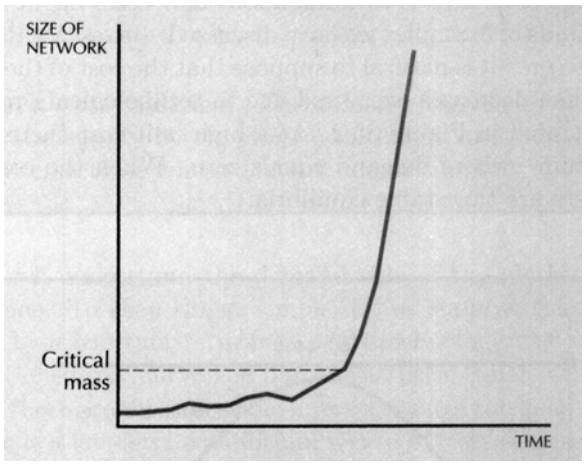

Figure 7: Critical mass phenomenon [Var06, 662].

the start of an online community the number of members increases very slowly. If a critical mass of users has been reached (as in Wikipedia), the online community provides a suitable amount of information which attracts new users heavily. The more users contribute to a wiki, the more it grows, and the more users get familiar with the system, which then for themselves are contributing and so on.

2.7 Human Motivation

Believing psychology, motivation is the driving force behind all actions of human beings, animals, and lower organisms. Motivation is often based on emotions and it is important because it is involved in the performance of all learned responses. Following Edward L. Deci and Richard M. Ryan, motivation can be viewed as either extrinsic or intrinsic [Dec00]:

- *Intrinsic motivation* – "Intrinsically motivated behavior is by definition self-determined. It is done freely for the inherent satisfaction associated with certain activities and with undertaking optimal challenges" [Dec87, 1033]. "Intrinsically-motivated behavior [...] is [...] behavior done for its own sake. Intrinsic motivation [...] refers to being moved to do something because it is inherently enjoyable. Intrinsic motivation leads organisms to engage in exploration, play, and other behavior driven by curiosity in the absence of explicit reward" [Bar05, 1].

- *Extrinsic motivation* – "Extrinsic motivation is encouragement from an outside force. Behavior is performed based on the expectance of an outside reward, such as money or praise. Extrinsic rewards can be abused to bribe or coerce someone into doing something that they would not do on their own"

[Puc02]. In most cases extrinsic motivation is aiming on security and acceptance in our society.

In this paper it is tried to motivate users by applying an extrinsic reward – the accentuation of most active members. Nevertheless, members of a wiki must also have some sort of intrinsic motivation to contribute, because else the wiki phenomenon would not be as successful as it is now.

Maslow [Mas87, 15ff] developed a rising order of needs from physiological to self-actualization needs (Figure 8 on the next page). The order of needs starts from basic survival or lower order needs to higher order. As one type of need is satisfied, another higher order need emerges and becomes operative in life.

- *Physiological needs* – The most potent of all the needs are the physiological needs. They are hunger, thirst, sex, temperature regulation, and rest. According to Maslow, when these physiological needs are deprived for a long period, all other needs fail to appear.
- *Safety needs* – When the physiological needs are satisfied, safety needs become the dominant force for the individual. Safety needs are mainly concerned with maintaining order and security, to feel secure and safe, and out of danger.
- *Belongingness and love needs* – The needs of this category emphasize the basic psychological nature of human beings to identify with group life. These are needs of making intimate relationship with other members of the society. People want to be an accepted member of an organized group, need a familiar environment as in family. These needs are dependent on the fulfilment and satisfaction of previous categories of needs.
- *Esteem needs* – Esteem needs are divided into two categories:

  1) Self esteem, self respect, and self regard.

  2) Relating to respect from others, like reputation, status, social success, and fame. The need of self evaluation occurs in those persons who are comfortably situated and satisfied with the fulfilment of previous lower order needs. For example, a competent professional who has established a high reputation and does not have to worry about getting worse, may become quite discriminating about what type of work she/he accepts.

  Other types of Esteem needs are the need to achieve, to be competent, to gain approval, and recognition. The need to feel superior to others falls under this category, too. For fulfilling this, the people may buy good and costly clothes.
- *Self-actualization* – The highest need in the hierarchical system, proposed by Maslow, is self-actualization. It is the desire to utilize one's personal capacities, to develop one's potential to the fullest, and to engage in activities for

which one is well suited. One should realize and be satisfied that she/he has achieved what she/he is capable of [Mas87].

Figure 8: Diagram of Maslow's hierarchy of needs [Sar04, 2].

According to [Kim00, 8f], Maslow's hierarchy of needs can be transferred from offline to online needs of a user (summarized in Table 2).

need	**offline**	**online**
physiological	food, clothing, shelter, health	system access; the ability to maintain one's identity, and participation in a web community
security and safety	protection from crimes and war; the sense of living in a fair and just society	protection from hacking and personal attacks; the sense of having a "level playing field"
social	the ability to give and receive love; the feeling of belonging to a group	belonging to the community as a whole, and to subgroups within the community
self-esteem	self-respect; the ability to earn the respect of others, and to contribute to society	the ability to contribute to the community, and be recognized for those contributions
self-actualization	the ability to develop skills and fulfill one's potential	the ability to take on a community role that develops skills and opens up new opportunities

Table 2: Comparison of offline and online needs.

As it can be seen, contributing to a wiki is an action belonging to the two highest-level needs. Needs of system access, protection from personal attacks, and belonging to a community must be satisfied to contribute to a wiki. By generating a list of most productive authors they should be recognized for their useful contributions within the community. At best this incentives are motivating in such a way that leads a user to develop her/his potential to the fullest. Then a user develops skills which opens up new opportunities for her/him, thus making her/his membership even more valuable to the community.

Furthermore, 16 basic motivations which affect a human's life were identified by Steven Reiss (professor for psychology and psychiatry at the Ohio State University and author of several books) who analyzed thousands of people for years: power, independency, curiosity, acceptance, arrangement, saving/collecting, glory, idealism, connections, family, status, revenge/combat, sex, food, physical exercise, and emotional tranquility [Rei02]. As it was mentioned earlier this approach of user motivation in a wiki is focussing on criteria like power, acceptance, glory, reputation, connections, and status.

Other theories of human motivation developed by Douglas McGregor at the MIT Sloan School of Management in the 1960s are *Theory X* and *Theory Y* [McG85]. They describe two very different attitudes toward workforce motivation:

- *Theory X* – In this theory it is assumed that the average human being is inherently lazy and avoids work if she/he can. Because of this, people need to be closely supervised and comprehensive systems of controls have to be developed. A hierarchical structure with narrow span of control is needed at each level. According to this theory, people show little ambition without an enticing incentive program and avoid responsibility whenever they can.

- *Theory Y* – In this theory it is assumed that people are ambitious, self-motivated, anxious to accept greater responsibility, and exercise self-control and self-direction. It is believed that humans enjoy their mental and physical work activities. It is also believed that people have the desire to be imaginative and creative in their work if they are given a chance.

McGregor's work was based on Maslow's hierarchy of needs. He grouped Maslow's hierarchy into "lower order" needs (Theory X) and "higher order" needs (Theory Y). He suggested that management could use either set of needs to motivate employees, but that better results could be obtained by meeting the Theory Y needs. By mapping these principles to this field of research, it can be said that users who are contributing to a wiki certainly must be ambitious and self-motivated thus meeting Theory Y criteria.

By having a closer look at the psychological field of behaviorism, Burrhus Skinner[15] developed the idea of "shaping". If one controls the rewards and punishments which the environment gives in response to behaviors, one can shape behavior (commonly known as behavior modification).

Shaping as a learning strategy means that successively closer approximations to some target behavior are rewarded. The reward is increased the closer the behavior approximates to the target behavior. The intended target behavior needs to be as specific as possible. If people do not know what it is wanted them to achieve, they can not know whether they are getting closer to achieving it or not.

The behaviorists' basic mechanism of learning is *stimulus => response => reinforcement* [Nic05b, 4]. The idea is a further development of Pavlov's classical conditioning. Skinner was not satisfied that all behavior was based on reflexes. For him, it is the history of reinforcements that determines behavior. Reinforcement may be either positive (reward) or negative (punishment). The model of learning process became also known as S-R-R (stimulus-response-reward).

As a user's behavior is wanted to be shaped in an online community, the rewarding methods have to be controlled. By applying Skinner's ideas every user has to see for which behavior she/he is rewarded or punished. Therefore, the goal for users is as clear as possible and everybody knows what she/he has to do to be rewarded. When a user gets rewarded for an action, it is likely that she/he is going to stick to that behavior. As unwanted actions are punished, people are trying to be conform with group norms, thus making the wiki an enjoyable place for everyone.

At this point, it is worth taking a look at what Wilbur Schramm[16] calls the *Fraction of Selection* [Sch71, 32]:

$$\frac{\text{expectation of reward}}{\text{expected effort}} = \text{likelihood of selection} \tag{3}$$

It is based on Zipf's[17] *Principle of Least Effort* [Zip49]. Zipf took the view that, all other things being equal, human behavior tends to flow into a path of minimum

[15] Burrhus Frederic Skinner (born 1904, died 1990) was at lifetime Williams James Lecturer at the University of Harvard and founder of the so-called *radical behaviorism* [Rac95, 363].

[16] Schramm (born 1907, died 1987) was of German origin and a very famous researcher in many topics related to communication [Ste87].

[17] "George Kingsley Zipf, a Harvard linguistics professor, conducted empirical studies of word occurrences, observing that if words are ranked by the number of times they are found in the text of a particular novel, their rank will be proportional to their number of occurrences. Hence, if a logarithmic plot of the number of times one finds each word is drawn against the rank of such a word in one's favorite novel, one will see a straight line" [Pot05, 100].

effort. The expected effort[18] in Schramm's equation can only be estimated and influenced by a user her-/himself. Therefore, the expectation of reward must be increased to enhance the likelihood of selection. That means the higher the expected reward, the higher is the chance that a user contributes to a wiki.

From the context of behaviorism, social learning theories were developed which mix many approaches from different schools of psychology. While accepting Skinner's view that a human learns to do what she/he does because of the direct reinforcement of responses to stimuli, Bandura[19] adds that a human learns also by observing the consequences of other people's actions [Und07]. In his studies Bandura closely observes children who are shown a "model" behaving in a particular manner. The extent to which they imitate the model is measured then. The results show that children are more likely to imitate models

- who are similar to themselves,
- who are seen to be rewarded for their actions, and
- who have some kind of prestige (because of their possessions, their strength etc.).

It is evident that other factors also play a role:

- If the motivational set is increased, for example, by offering a reward for imitating the model, subjects are more willing to imitate.
- Previous experience influences the attention paid to particular aspects of the model's behavior, for example, children used to violence are more likely to pay attention to violent aspects of the model's behavior.

In this case it can be assumed that the contribution rate increases when, for instance, users are working on an article they are interested in, with people behaving similar to themselves, if they know that they are rewarded for their work, or when high ranked authors are integrated in writing an article. Best results should be achieved if all criteria are met.

2.8 Motivating Contributions in Online Communities

This chapter focuses on an article called "Using Social Psychology to Motivate Contributions to Online Communities" [Lin04]. In this article a group of professors

[18] Or "perceived expenditure of effort" as Schramm calls it.

[19] "Albert Bandura (born 1925) is at present David Starr Jordan Professor of Social Science in Psychology at the Stanford University and most famous for his work on social learning theory" [Ban06].

and students from different universities were investigating the problem of under-contribution in an online community called *MovieLens*[20] and how it can be solved using techniques from social science theory. MovieLens is a web-based movie recommender site where members rate movies, write movie reviews, and receive recommendations for movies. Design principles derived from social psychology theories were tested using four field experiments involving members of an online movie recommender community. The results of the article are summarized in this chapter.

Social loafing, or free riding, is the robust phenomenon that occurs when people work less hard to achieve some goal when they think they are working jointly with others than when they think they are working by themselves. The collective-effort model [Kar93] is a type of utility theory that claims that people work hard when they think their effort will help them achieve outcomes they value. Working in a group can influence how hard people work because it can change their perception of the importance of their contribution to achieving a specified level of performance, their likelihood of reaching the goal, and the value they place on the outcomes they gain by their efforts ([Har82] and [Ker83]).

The collective effort model identifies conditions under which people will socially loaf less. These include, among others: (1) believing that their effort is important to the group's performance, (2) believing that their contributions to the group are identifiable, and (3) liking the group they are working with.

Experiment 1

In experiment 1 of the article predictions from the collective effort model where tested which stated that people will contribute more to a group when they think their contributions are likely to be unique and when they like the group more [Kar93].

> Hypothesis 1: People will contribute more to online communities when given personalized information showing that their contributions would be unique.

People tend to like others who are similar to themselves [Byr97] and to dislike groups composed of dissimilar members [Wil98].

> Hypothesis 2: People will contribute more to online communities when they believe that they are similar rather than dissimilar to others in the group.

[20] `http://movielens.umn.edu`

Consistent with Hypothesis 1, subjects posted more messages in the uniqueness condition, when they were given personalized information about how their knowledge of movies differed from others. Hypothesis 2 was disconfirmed: Subjects posted fewer messages when conversing in groups constructed so that members had similar tastes in movies than in groups with heterogeneous members.

Experiment 2

> Hypothesis 3a: MovieLens users will rate more movies when the personal benefit they receive from doing so is made salient.

> Hypothesis 3b: MovieLens users will rate more movies when the benefit they provide to the community from doing so is made salient.

Contrary to Hypothesis 3a, participants who received the self-benefit message reduced their number of ratings. In addition, contrary to Hypothesis 3b, participants who received the other-benefit message also reduced their number of ratings. Participants who received both self- and other-benefit messages increased their number of ratings almost to the level of the control condition.

Experiment 3

Previous research has shown that when people are intrinsically motivated to perform some behavior, the promise of extrinsic rewards, such as money or grades, reduces their intrinsic interest in it.

> Hypothesis 4: Members who receive messages that increase salience of intrinsic motivation will rate more movies than those who receive messages that do not increase salience of intrinsic motivation.

An especially perplexing finding from Experiment 2 is that mentioning either self-benefit or other-benefit reduced ratings from a control condition, but mentioning both together did not. If mentioning a single benefit narrows focus, mentioning more should reduce this narrowing.

> Hypothesis 5: Members who are reminded about the multiple benefits that a contribution provides will rate more movies than those who are reminded of only a single benefit.

Hypothesis 4 could not be confirmed due to statistical insignificance and Hypothesis 5 was disproved by experiment.

Experiment 4

Hundreds of studies with over 40.000 subjects have shown that specific, challenging goals stimulate higher achievement than easy or "do your best" goals [Loc90].

> Hypothesis 6: Members who are assigned challenging, specific numeric goals will rate more than members assigned non-specific do-your-best goals.

The collective effort model [Kar93] predicts that individual goals and feedback will be more motivating than group goals, because in a group setting people may believe that their contribution is partially redundant and that if they shirk, others can take up the slack.

> Hypothesis 7: Members assigned individual goals will rate more than members assigned group goals.

Recent research has shown that goals which members view as overly difficult could reduce contributions.

> Hypothesis 8: Members assigned exceedingly difficult specific goals will rate less than members assigned difficult specific goals.

Hypothesis 6, which predicted that members given specific numeric goals would rate more than those given do-your-best goals, was affirmed. Subjects rated 27% more movies when given one of the specific goals than the non-specific do-your-best goals. Hypothesis 7 was disconfirmed, as subjects in the individual-goal condition rated only 42% of the movies they rated in group-goal condition. Hypothesis 8, predicting the highest challenge goal would lead to a decline in ratings, was weakly supported. Participants made the most ratings when they received intermediate goals, but made fewer when they were given the unchallenging goal of eight ratings or the most challenging goal of 64 ratings.

As in former chapters it was concentrated on the theory of user motivation to contribute in an online community, in this chapter an empirical study is presented which should proof theory. Although not all items from theory could be confirmed, a few criteria can be filtered out which are the most interesting to focus on when talking about motivation in online communities. Apparently, it is beneficial to emphasize unique contributions of users to a community and it is better not to interfere with group building processes. It is also very important to assign members of a community challenging and specific goals. Furthermore, goals should not be too difficult to achieve because then contribution can sink. In addition, members should be reminded of their intrinsic motivation from time to time to participate in an online community.

2.9 The Problem of Free Riding

A very common social dilemma of online communities is free riding. In a general social dilemma, a group of people attempts to utilize a common good in the absence of central authority. In the case of a system like Wikipedia one common good is the provision of a very large library of articles to the community. The dilemma for each individual is then to either contribute to the common good or to shirk and free ride on the work of others. For example, a user who is only reading articles but never posts one.

Since articles on Wikipedia are treated like public goods and the users are not charged in proportion to their use, it appears rational for people to view articles without contributing by making their own documents accessible to other users. If every individual did this way and free ride on the efforts of others, the whole performance of the system would degrade considerably, which makes everyone worse off – the tragedy of the digital commons [Ada00].

The free riders problem is easy to understand: Participation (like posting an article) in an online community like Wikipedia generates costs to a user. Therefore, it is easier for her/him to only consume the good provided (information) and not to take part in the process of generating it. So, the user gets something for free – she/he does not need to give anything in return. The only costs for free riders are charges for the Internet connection and their time spent to search for the desired information[21]. If all users are acting as free riders, Wikipedia would not have any articles in its free encyclopedia, thus making this online community obsolete.

2.9.1 Game Theory Approach

The problem of free riding is going to be demonstrated by making use of the very well-known game theory[22]. For simplicity reasons, the described game models has only two players, although in an online community thousands of users or even more are participating.

As shown in Table 3 on the following page, a game with two actors A and B is

[21] Search and information costs are subclasses of transaction costs. By gathering information using the Internet, transaction costs are almost falling to zero. For example, the time a user spends to search for a particular information can be lowered if a huge online community is providing such information in a structured way.

[22] "Any situation in which individuals must make strategic choices and in which the final outcome depends on what each person chooses to do can be viewed as a game. [...] Game theory models seek to portray complex strategic situations in a highly simplified and stylized setting" [Nic05a, 440]. "A game is described in terms of the players, the rules of the game, the payoffs of the game, and the information that players have about the details of the game" [Man04, 441].

assumed. Each actor has two possibilities: posting an article to the Wikipedia system or not. On the one hand, if actor A posts or does not post an article, this has an indirect effect on actor B. If actor A posts articles, but actor B does not, A has spent her/his time to do research on a specific topic and has posted the looked up material in an Wikipedia article. Actor A gains no profit from posting an article because she/he found the needed information and could use it without sharing it with others. That is why there is a high negative value (-3) for the posting of A. She/He has costs but gets nothing in return. On the other hand, B profits from the new article of A, because Wikipedia was filled with another resource which she/he can use. The same situation (only vice versa) is created if actor B actively participates in Wikipedia and actor A does not. If neither actor A nor B posts articles both gain a profit (1, 1), because acting as free riders they trust that someone else in the community posts articles which they can use.

		Actor B	
		post	**no post**
Actor A	**post**	$-2, -2$	$-3, 2$
	no post	$2, -3$	$1, 1$

Table 3: Problem of free riding.

In this scenario the worst case, in a collaborative view, is when both players post articles (-2, -2). It seems that posting articles is a very expensive operation which the benefit from generating information can not equalize. But exactly this situation is wanted and needed for a wiki to expand and live.

By looking at the interaction of both players, there are several possible scenarios, but only one strategy which is dominant (suits both players best). If actor A decides not to post an article (value of 2 or 1 depending of the choice of B), she/he can gain greater profit than if she/he posts one (value of -2 or -3). That is the reason why actor A in this game certainly does not post an article. The effect on B is that she/he also does not post an article, because by free riding she/he can obtain at least a value of 1 (compared to -3 if she/he does post). Therefore, nobody submits articles which is the best strategy for this game (collaborative value of 2).

An online community where nobody participates is certainly the worst case. The question is: How can users be motivated to contribute to an online community and do not only free ride? There have to be certain incentives which generate a surplus and motivate users to participate actively. These stimuli can turn the scenario described in Table 3 into a desired one shown in Table 4 on the following page.

In the game in Table 4 on the next page there must be some sort of motivations (social rewarding mechanisms, social punishments, etc.) that force actors to participate actively. If both actors do not post articles, there have to be some sort of sanctions.

These penalties must overlap the generated benefit of free riding (shown as values of $-1, -1$).

		Actor B	
		post	**no post**
Actor A	**post**	$2, 2$	$2, -1$
	no post	$-1, 2$	$-1, -1$

Table 4: Approach to solve the problem of free riding.

If actor A posts articles and B does not, again there is a negative value of -1 for B. Actor A gains a positive value of 2 for participating actively. Therefore, for both players free riding does not pay off, because of negative effects connected with it. In this game the best strategy for both A and B is to post articles which collaboratively gains the highest surplus.

Looking at the interaction of both players, one can see that it does not matter if a player posts an article or not, because the other certainly posts because of an invariably positive value of 2. As it was mentioned before, none of the members does not post because of the linked negative effects. Therefore, the dominant strategy is that both players post which gain the most benefit and which is also collaboratively the best strategy (for the whole community).

Unfortunately it is not enough to know how the perfect game for this situation looks like. The difficulty is the examination of strategies and incentives to motivate members of wikis to contribute. Different approaches to solve this problem are described in Chapter 3 on page 47.

2.9.2 Public Good Problem

As noted before, articles on Wikipedia are treated like public goods. Once they are made available to one person, they can be consumed by others at no additional marginal cost; this condition is commonly called jointness of supply or nonrivalness of consumption, because the consumption of a good by someone does not affect another. Therefore, in standard price theory, in which price tends to equate to marginal cost, such goods should have a zero price. But if they are priced at zero, they will generally not be provided. In essence, price theory commends free riding on the provision of such goods. This might sound merely like a logical problem but standard examples include radio broadcasts, national defense, and clean air. If any of these is provided for anyone, they are de facto provided for everyone in the relevant area or group.

Another feature of public goods is the impossibility of exclusion. Once supplied at

all, it is supposedly impossible to exclude anyone from the consumption of a public good, for instance, an article on Wikipedia can be viewed by anyone with an Internet access and a web-browser. It is often noted that this feature is analytically interesting but empirically often beside the point. Exclusion is merely a problem of technology, not of logic [Har07].

By looking at the free riding problem concerning public goods, it is formally assumed that individuals can only add positive value to a public good (so spammers and people who knowingly contribute false information are factored out). That means, in the sense of Wikipedia, that people generate a benefit even if they delete articles, because the information provided then seemed to be incorrect. Every person can decide if she/he wants to increase the amount of the public good or not. Then it can happen that one person decides that the amount provided by others is exactly right and prefers to contribute nothing (free rider).

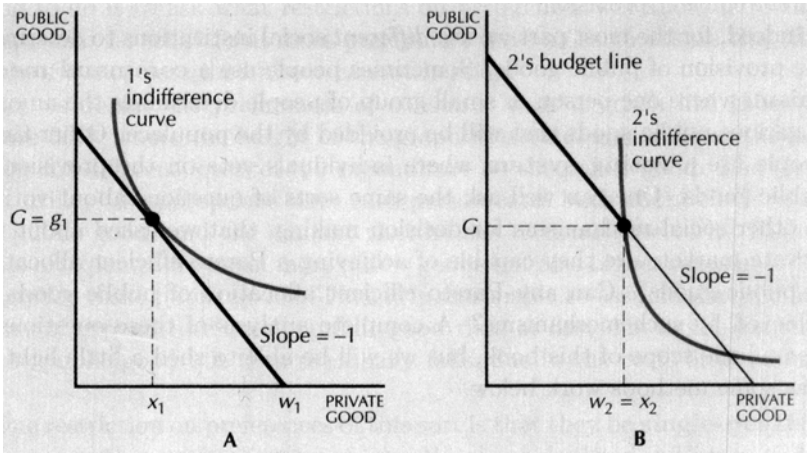

Figure 9: Problem of free riding [Var06, 683].

$G =$ total amount of public good
$g_1 =$ amount of contribution of Person 1 to public good

Such a case is displayed in Figure 9. A person's private consumption is displayed on the horizontal axis, while the public consumption is shown on the vertical axis. Every person has a "setting" assembled of the amount of her/his private goods (w_1) and the amount of public goods provided by the others (if this person decides to contribute nothing, this will be the available amount of the public good). Figure 9A displays a case where Person 1 is the only one contributing to the public good ($G = g_1$). If Person 1 contributes G units to the public good, Person 2's setting is assembled of her/his private goods w_2 and the amount of the public good G, because Person 2 can consume the public good regardless of contributing something or not. Person 2 can not reduce the amount of the public good (only increase it) therefore her/his budget constraint is the thick line in Figure 9B. By looking at the

given indifference curve in Figure 9 on the preceding page it is optimal for Person 2 to free ride on the costs of Person 1.

If a person offers a public good, this will lead to a reduction of other persons' contribution, because everyone can consume a public good in exactly the same quantum. Generally speaking will in a voluntary equilibrium a public good be insufficient offered, proportional to its efficient allocation [Var06, 681ff].

With this approach of social rewarding methods the problem of free riding in public goods is addressed, as well. Users will free ride less when they see that people are rewarded for their work. After all, a user has to be convinced that her/his contribution is generating a surplus not only for other members of the community but also for her-/himself.

2.10 Social Rewarding in Online Communities

In this chapter a closer insight to the structure of online communities – especially wiki systems – has been given. Motivations a user has to contribute to an online community have also been explained. In order to invent social rewarding mechanisms it is necessary to understand the behavior of a social group, such as the members of an online community. As it can be seen, there are many different member roles and types in online groups which interact in a social network. Different models exists to calculate the value of a network which, generally speaking, increases with the growth of members.

Users who contribute to an online community like Wikipedia are mostly intrinsically motivated. They do not share their knowledge with others for money but for their own sake. The only extrinsic factor (an outside reward) is to earn praise. And that is the point where social rewarding techniques should start. If a user contributes to an online community, she/he expects that her/his work is recognized and wants to be praised. In most cases, it is enough that the user who contributes knows that others are aware of what she/he has done. In most cases social rewarding belongs to accentuation of most active users. Her/His name must be present in the top contributors list and she/he has to know that. After all, users want to have something in return for their hard work and be it acknowledgements from others instead of money.

A user contributing to a wiki has another motivation as someone looking to get something to eat. Following Maslow's hierarchy of needs, it can be seen that a user wants to satisfy needs at the highest levels of the pyramid: *esteem needs* or even *self-actualization* [Mas87]. According to [Kim00] these offline needs can be mapped to online needs. This means that users must have the ability to contribute to a community and be recognized for those contributions. At the highest level a user

has to have the ability to take on a community role that develops her/his skills and opens up new opportunities for her/him. Following Steven Reiss' basic motivations [Rei02] some factors are found which certainly match a users' preferences (and are in perfect compliance with Maslow's hierarchy of needs): power, acceptance, glory, connections, and status. These factors (or at least some of them) have to be satisfied as best as possible by using social rewarding techniques.

As also mentioned in Chapter 2.7 on page 36, people tend to take the path of minimum effort. In connection with that the free riding problem as well as the public good problem occurs (Chapter 2.9 on page 41). From an economic point of view it can be said that to motivate a user the expectation of reward must be greater than the expected effort. A user has to think subjectively that her/his reward will be greater than her/his effort. Every user has her/his own preferences of rewarding methods. For a billionaire money will be less motivating than for a beggar. Here it is talked about needs at the highest level of Maslow's pyramid, therefore, money can be excluded as a pushing criterion. So it has to be stuck to social rewarding methods. After all, these are the factors on which the techniques introduced in the next chapter are focussing.

3 Analysis of Social Rewarding Techniques

As mentioned in Chapter 1.2 on page 9, social rewarding refers to something that causes a behavior to increase in intensity. In this chapter, different social rewarding techniques are analyzed which can be used in an online community. It is concentrated on the MediaWiki system, where as a next step selected social rewarding mechanisms are implemented. A quick overview of the presented social rewarding methods is available in Appendix A on page 111.

3.1 Top 10 Lists

The term "top 10 lists" summarizes any list based on some sort of ranking method which ranks the best in first place, the next in second and so on. This cardinal ranking mechanism is not – as the name may assume – limited to the number of ten items. Historically, most ranking lists only displayed ten or less items. Therefore, this section have been named this way, even though it is shown that in the MediaWiki system technically there is no limitation to the number of items in a list.

3.1.1 Top Contributors

One very common social rewarding mechanism in online communities is the overall listing of top contributors. In the sense of MediaWiki this would be a ranking of authors with the most posted articles or with the most articles edited. As the MediaWiki system is developed for cooperative work, every person can submit articles as well as edit existing ones. In a wiki there are several methods to calculate a top contributors list:

- *New articles* – Counting can be restricted to new articles only. That means new articles are counted only and increase a user's amount of metered contributions. Such a solution is not very satisfying because editing an existing article can be equally or even more valuable than creating a new one. It does not matter if another user already created a topic – the only thing that counts is the usefulness of the content. For example, one can contribute an article by simply typing two or three nonsense words. This behavior should certainly not increase her/his position in the top contributors list.

- *Editing existing articles* – As explained above, editing an article is as valuable as creating new ones. Editing implicates adding new or deleting wrong information, attach multimedia (photos, diagrams, audio and video elements etc.), correction of spelling, grammar, or punctuation mistakes, changing the structure or design of an article and so on. As it can be seen on Wikipedia, collaborative writing on articles is very common. It is very hard to find an article edited only by one user. Many authors have different views on a topic

and therefore write different theses, thus lightening the topic from different angles. Furthermore, every author is urged to represent not only her/his own opinion, but facts based on a neutral point of view.

- *Deleting existing articles* – Deleting an existing article which, for example, contains completely wrong information, is also some sort of contribution. For a list of top contributors, users who participate by "cleaning" false resources must not be forgotten.

Calculating a top contributors list using only the items above is a good starting point but might not be very satisfying. Additional conditions may be made, such as counting the number of words posted. For adding a new article or editing an existing one, this is a good quantitative measurement for the contribution rate. Nevertheless, there is no qualitative statement of the content, but it is assumed that most users behave as expected and submit only valuable content. As it can be seen in Wikipedia, the amount of spammers and people contributing wrong information intentionally is very low[23]. Bad manner does not pay off, because every user can restore an older revision of an article with absolutely no effort.

It has to be considered that in the MediaWiki system there is the possibility to preview articles before posting them. The preview shows how the article will look like after submitting it. Previewing is possible by both submitting new articles or editing existing ones. Therefore, it has to be assured that previews must not be count. Another issue of counting articles is that one user can write an article as a whole, while another is writing the same article piecewise. The second author then submits an article and edit it, for example, three of four times. Then there is the situation that author one has only submitted one article, whilst author two has submitted a much shorter article, but edited this one a few times (and therefore has a higher amount of measured contributions). This is an unfair setting because both authors have posted the same content, but with different methods. That is why there should be a restriction, that if an author edits an article a few times within a certain period of time, it should count as one contribution regardless of the number of edits. Another idea would be that a contribution should only count if it exceeds a certain amount of words.

Top contributors might not only be listed as an overall amount of all the postings made in the whole online community, but there might also be categorized top lists. As in MediaWiki each article should belong to a category[24], it would be obvious to provide ranking lists of top contributors for each category ("key players" in one category).

[23] A very big and active community is mostly a guarantee for quick sanctions for unwanted behavior.

[24] Although categorization is not compulsory.

To quickly identify top contributors there ought to be the possibility to link a graphical sign to a username. Every time the username is displayed, a certain sign should be displayed which indicates the participation rate of the user. A comparable method is used at the world's most famous Internet auction house eBay[25]. There, a graphic (stars in different colors) appears behind the username every time it is displayed indicating the sum of the user's feedback profile[26]. To see if a MediaWiki user is a regular contributor, a comparable graphical indication, like it is used on eBay could be useful. Contributors with 10, 50, 100, or even 1.000 submitted, edited, etc. articles (whatever counts as participation) should be identified by linking the username to a graphical sign with different color schemes characterizing the user as a poor, under average, over average, or as a top contributor[27].

3.1.2 Last Contributed Articles

The idea behind a list of last contributed articles[28] is that new articles should be put into the spotlight to emphasize them. On the one hand, users should be informed about new articles so that they know which topics were edited. On the other hand, new articles are often short and do not contain a lot of information. Wikipedia calls articles with little information which have still to be filled up *stubs*[29]. To pay more attention to presented articles in a list of last contributed articles, the listing has to be put on an easy to find special page or even on the front page of the wiki.

A list of last contributed articles can differ in the amount of articles shown. If the list is presented on a page especially designed for it, there can be ten, twenty, fifty, or even more articles displayed. The length of the list is connected to the size of the community. If there are only ten new articles a week, it should be enough to list ten or fifteen articles. In that case it makes no sense to display a list of hundred articles where the last were written two years ago – this would not be a list of *last* contributed articles. If a wiki has a large and active community and there are, for example, 30 to 50 new articles a day, it would make sense to display a list in the dimension of 100 to 200 articles. Of course, the articles are sorted in descending order by their contribution date and there should be the possibility to sort the listing by hits, hence making a list of most viewed articles.

[25] `http://www.ebay.com`, retrieved on 1 September 2007.

[26] For an explanation of eBay's rating system it is referred to Chapter 3.2.1 on page 60.

[27] There can be as many steps between a poor and a top contributor as one would like.

[28] In the MediaWiki system the same feature is implemented out-of-the-box and is called "new pages".

[29] Stubs are articles which have not yet received substantial attention from editors of Wikipedia, and as such do not yet contain enough information to be considered complete articles. The community believes that stubs are far from worthless; they are, rather, the first step articles take on their course to becoming complete [Wik07k].

A general list of last contributed articles, like it was mentioned above, must have a link to the authors of the corresponding articles. The authors' names have to be shown not only on the site where the full article is displayed, but also on the list beside the name of the article. Certainly, it would also be a good idea if there is the possibility to restrict the shown list to one or several authors. This can be done by using a drop-down list (if there are only a few authors) or by using a text field for inserting the authors' names.

A list of last contributed articles can also be placed in a specific user-page displaying only the last contributed articles of this author. Of course, the amount of items listed is fewer than in a general overall list. A good size would be displaying five to ten articles from the author, depending on the available space on the page and the date of the articles published. Then a user has an insight which other articles the author has recently worked on. For example, if a user thinks that an author has a good qualification in a certain topic or does her/his research very accurately, she/he can be interested in reading other articles from the same author.

Not only last contributed articles can be displayed, but there is also the possibility to generate a list of recent changes (in MediaWiki this feature is implemented out-of-the-box). Recent changes include every modification made in a wiki, for instance, creating, editing, deleting, restoring, etc. of articles. As a listing of last contributed articles only shows the newest submitted articles, a list of recent changes certainly displays many more items. In the end it has to be said that presented methods for last contributed articles can fully be applied to a list of recent changes.

3.1.3 Most Viewed Articles

The idea behind a list of most viewed articles is that when an article is viewed by many people it is either very informative and very well written with good background knowledge of the author, or it has a highly interesting theme for a broad range of people. If case one is assumed, it can be said that articles which have a high rate of hits[30] or visits[31] help to achieve a good reputation for their authors.

A list of most viewed articles can be an overall list of most viewed articles ever or separated in a certain space in time, or articles can be categorized by their topic. A list of most viewed articles ever can be a good idea, although there is certainly not much fluctuation among the top articles in the list. To avoid this behavior, counting only the views of the current month or week would be a solution. For example, if

[30] "A single access request made to the server" [One05].

[31] "A visit is a subset of consecutive page views from a user session occurring closely enough (by means of a time threshold or a semantical distance between pages)" [Tan04, 60].

there was an article about September 11th 2001 in the wiki, in a list of most viewed articles ever this article would range on top, although most of the hits or visits definitely were made a couple of months after the creation of the article. After six years passed by, the hit rate of this page would certainly have fallen to an average. If there are lists of most viewed articles per month, the article would range on top of these lists for only a few months, sinking further down from month to month. A categorization by the topic of an article is good for knowing which categories have more views than others and in these categories which articles are viewed most. Comparisons can also be made, for example, "Why does the top article in the medicine section have twice the hit rate of the top article in the category economics?".

The two categorization methods described in the former paragraph can be combined easily. Then it is possible, for example, to search for the most viewed article in the Information Science category for January this year. Additionally, the hit or visit rate of the most viewed article in any section from March this year can be compared to March last year. The categorization methods described in this chapter are not only restricted to a list of most viewed articles, but can be applied to nearly every method described in the "top 10 lists" section.

A list of most viewed articles only makes sense if there is a connection to the author(s) of a certain article. It is recommended displaying the author's name beside the title of the article in the list. A restriction to show only most viewed articles from one author or from a defined list of authors would generate a surplus to the users, too. Like in the chapter above, the amount of items in the list has to be well-chosen and a suitable solution has to be found.

Another approach for a list of most viewed articles would be displaying the list on a user-page showing only most viewed articles from this specific user[32]. With this method user-specific preferences for a topic or an author can be shown. The ranking of most viewed articles can also be generated using social network theories. If it is assumed that a user who is one of the top contributors of a wiki has a higher reputation than a poor contributor, it can be implied that an article viewed by her/him is worthier than an article viewed by a user with less participation. It must be supposed that a user with high participation in a wiki knows the system very well and therefore, also knows better if an article displays needed information right. Simply speaking it can be said, for example, that a hit from a top ten contributor counts ten credit points, while a hit from an average contributor counts only two or three. Summing up the "hit points", a weighted list of most viewed articles can be gener-

[32]Displaying such a listing without asking the user results in offending her/his privacy. Therefore, a user must have the option to turn off data collection and presentation. In fact, as a default setting, collection of data regarding a user's behavior must initially be turned off or must only be visible to her-/himself and not to the public.

ated. As this method is some sort of a credit point system, more information can be obtained by reading Chapter 3.2.3 on page 62.

3.1.4 Weighted Top List of Authors/Articles

A slight modification of the second approach for a list of most viewed articles shown in the former chapter, would be if hits from top/poor contributors are weighted more than from average users. A top list of articles can be created if the focus is on articles most viewed by top contributors (and therefore displayed in their most viewed list at their user-page) of the wiki. A ranking can be constructed in which an article has a better place if it is in the list of most viewed articles of, for example, three top contributors than an article which is only in the list of most viewed articles of one top contributor. Using this method it is assumed that articles which are viewed from top contributors very often have better information to offer than articles which are viewed less.

By connecting articles to authors it is possible to generate a ranking of authors using the described method above. If an author has many articles ranked very high on many top contributors' lists of most viewed articles, she/he can be sure to be ranked in a good place at a weighted listing of authors. Of course, there is the problem that one article might not be directly linked to only one author but to several ($1 : n$ relation). That is the reason why a high ranked article has an effect on the position in a listing of all authors involved.

At this point it is necessary to mention that described methods and approaches to problems characterized in a subchapter of this thesis (like this one) are not only limited to one subchapter, but can possibly be applied to techniques shown elsewhere, too. The approaches presented here can be, for example, transferred to another chapter and there usefully applied, as well. For instance, every article in a top list should have a link to the author(s) of the article. A commonality (nearly) every presented method for creating a top list has.

3.1.5 Extra Long Articles

Articles in a wiki system differ extremely in their length. Some topics need an in-depth explanation, causing these articles to be pretty long, while other topics may be presented in a very short way. This does not necessarily mean that the author invested less effort in creating the article, though it must be said that a longer writ-

ten article in most cases has higher costs than a short one[33]. Very short articles are in most cases stubs (for an explanation of stubs it is referred to Chapter 3.1.2 on page 49) and need more attention from the community. Most new articles also tend to be pretty short. That is why extra long articles should be emphasized and authors should be rewarded for their hard work.

A definition of extra long articles can be the amount of words or characters of an article or the size in (kilo)bytes. There is also the possibility to count not only characters but also multimedia content in an article. An article containing many pictures, graphics, video or audio streams, charts and so on addresses the attention of a user much more than only written text with no further material. This is why a definition of extra long articles might not be so strong as to only count characters, but also multimedia elements used in the article. Many articles need graphics for their basic understanding and for visualization of their content.

In the MediaWiki system a special page called "long pages" is available, displaying an ordered list of longest articles measured by their size in bytes. The list ignores multimedia elements as well as highlighting the authors of these long pages. An adaption of the listing including these two criteria in the measurement process would be desirable. Another attempt to emphasize authors would be summing up all written articles of an author and, for instance, displaying a list starting by the author with the most written words counted as an overall of all her/his articles.

Methods shown here can only be used to generate quantitative listings, because there is no attempt to measure the usefulness of the content of an article and no effort is made investigating if the content of an article makes sense or not. An additional problem in connection with that is the handling of "spam", where information can make sense (e.g., advertisement links or sales promotion), but is in the wrong context and therefore unwanted. Most spam entries are generated automatically, that is why it is very likely that articles with spam entries have pretty much the same structure and/or content. Filtering some standard spam-phrases (like HTML tags) and prohibiting the contribution of articles containing such information can be a starting point. Another idea would be the restriction to submit only a certain amount of articles per minute. However, in the end spam can only be defeated with the help of the community's members by deleting or rewriting articles containing spam entries.

[33]This problem concerns also a list of most edited articles. If an article has been edited a thousand times (from different authors), it can be said that some effort has been made. This does not necessarily mean that the information provided in the article is useful or more useful than an article with only a couple of edits or none at all. Nevertheless, in most cases an article with many edits from many authors is much more detailed and informative because of the collaborative writing and the different points of view the authors have.

3.1.6 Often Used Search Terms

As a wiki system has no hierarchical structure where a user can find information top down or bottom up (there is no strict way to find desired information), many are using a search engine to find articles. Often used search terms could be linked to articles (which themselves are linked to authors) thus displaying articles most visited through the help of the search engine. For example, if a user searches for "information visualization examples" and another for "information visualization definitions", for both search terms the same page could appear in the result list (for example, on the same position in the list, as well). If a user visits the page with the desired information on it by clicking on the displayed link in the result list, a counter could recognize this event by increase its hit value for this page by one. It can be assumed that if many users are ending up on a specific page by using the search engine, this page offers information which are useful to a lot of people. A further presumption would be that the article has been written very well and the topic is very interesting or an up-to-date one. So the author must have spent either a lot of energy in writing the article or a lot of time researching an up-to-date story or even both.

A listing with often used search terms increases user visits for the top articles in the result list of top search terms. A listing of most visited articles through the use of the search engine does the same, but results in more visits on less different pages. A listing certainly has to contain information about the author(s) of an article. Most found articles from a certain author can also be obtained, thus creating a ranking of authors which articles where most viewed through the help of the search engine. With this method it is possible to find articles from authors which are interesting to a lot of people. Articles from authors or even authors themselves can also be linked to used search terms to identify different categories where the author is active. That means if articles of an author are found most likely by using the search term "information visualization" (and terms which differ slightly), this author seems to be very engaged in the field of information visualization.

3.1.7 Amount of References

In the case of the Wikipedia encyclopedia – as in every other scientific publication – the value of an article grows with the amount and the quality of used references. It is possible that in such a free encyclopedia wrong information has been published. The author need not necessarily know that she/he has submitted wrong information. Therefore, it is likely that the usefulness of an article increases with the number of used references and with their esteem and trustworthiness.

If an author refers to a couple of references all showing the same information, a user trusts the article more than if there is only one reference or none at all. The number

of references can be counted with no effort[34], but the quality of the references is not as easy to estimate. If a user has found the information to an article in a book, it may be more reliable than a reference to a web-page. An explanation could be that information on a web-page can be published from nearly everyone at almost no costs. Printing a book is expensive and only people with an in-depth knowledge of a specific topic do that because many people have to buy the book to cover the expenses.

An approach to an automated quality check of Internet resources can be realized by the help of a very well-known search engine which describes itself as the world's leading search engine with the largest index of web-pages: Google[35]. The search algorithm of Google is named *PageRank* and it "relies on the uniquely democratic nature of the web by using its vast link structure as an indicator of an individual page's value. In essence, Google interprets a link from page A to page B as a vote, by page A, for page B. But, Google looks at more than the sheer volume of votes, or links a page receives; it also analyzes the page that casts the vote. Votes cast by pages that are themselves 'important' weigh more heavily and help to make other pages 'important'. Important, high-quality sites receive a higher PageRank, which Google remembers each time it conducts a search. [...] Google's complex, automated methods make human tampering with our results extremely difficult" [Goo07].

So how can Google help to check the quality of references in an article? It has only to be checked how much sites are linked to the one used as a reference by using Google's search engine[36]. If only a couple of sites are linked to an Internet resource used as a reference and this links themselves have only some sites linked to them, the resource seems not very reliable. But if many Internet sites link to one site and this links themselves have a high number on links to them, information displayed on this site must have at least a basic level of plausibility. Why should a site have thousands of links pointing to it, when there is no reliable information displayed?

The output of Google's search can, for instance, be interpreted as follows: Summing up the numbers of web-pages pointing to all references used in an article generates a quality index number. Such a quality index number can be used for a basic classification of the references as more or less credibly. So at least an initial quality check of Internet resources can be realized by using Google's PageRank technique.

[34] The number of references used in an article must be set in relation to the length and the topic of the article.

[35] `http://www.google.com`, retrieved on 1 September 2007.

[36] Using "`link:http://www.tuwien.ac.at`" as a search term results in showing all pages linking to the URL `http://www.tuwien.ac.at`.

Using many references with high quality index numbers can indicate a well written article with correct information. A list of articles using these criteria as computation basis can be generated. A listing can also be produced ranking those authors topmost that submitted articles having a lot of references with high quality index numbers. This attempt tries to rank authors not only by means of quantitative measurements but also qualitative ones.

A more global idea would be counting the links to an article from outside the wiki system. By using Google, not only references within articles can be checked, but it can also be figured out how much sites outside the wiki are pointing to an article. If there are thousands of links to an article, it is likely that this article is valuable to many people. Furthermore, the higher the amount of links to an article is, the higher is the frequency of visitors and readers. Many links are therefore an indicator for good quality of an article. Summing up all links to articles of an author can generate a listing where the author with the most linked articles is placed first.

3.1.8 Most Linked User-Pages

In the MediaWiki system every user has the possibility to create her/his own user-page. Information provided on such user-pages can be anything the user would like to reveal about her-/himself, although displayed information need not necessarily have a relation to the user.

As suggested in former chapters, one can utilize user-pages for some of the presented social rewarding mechanisms, for example, for different types of top 10 lists. If not only personal information is provided on a user-page, other users can have a surplus by reading such web-sites. Sure, personal facts can also be interesting, but if there are rankings and links to contributed articles, everybody has a quick overview of the topics a user is involved in. Sometimes raw statistics of a user's productivity can tell more than her/his self description.

If it is assumed that on user-pages interesting material is presented, a listing of most linked user-pages can be reasonable. The more articles and special pages link to a user-page, the more it is likely to be expedient. The mechanism is a little like Google's PageRank, where the importance of a site is measured by the links of other sites to it.

MediaWiki supports its users with an out-of-the-box tool named "what links here". At this web-page a user can view which sites are linked to the site directly viewed. The technique can be executed on nearly every page of the MediaWiki system – no distinction is made, for instance, between articles and user pages. With the help of this function it would not be too difficult to generate a list of most linked user-pages.

Certainly, a listing of most linked articles could also be created, which answers the question "Which article in a wiki has the most links to it?". Summing up all linked articles from a specific author can also lead to a ranking of authors with most written and linked articles. Connecting the number of linked articles from an author to the amount of links to the user-page of the same author can result in an overall statistic of linked resources of a specific author.

3.1.9 Productivity of Authors

Social rewarding should not only target authors which were very productive in the further past, but should cover also over average contributions of authors at present. Measuring the productivity of authors over time is a good instrument for time oriented social rewarding. If a list of most viewed articles ever is published, it may happen that the top articles never change because at a particular point in time there were so many hits or visits that no other article could make up. That reason along with others (described in Chapter 3.1.3 on page 50) are the matter for splitting such top lists, for instance, in monthly published parts.

The productivity of authors can be measured in different ways. An abstract of various methods is described as follows:

- *Monthly list of top contributors* – A listing of top contributors of the month (or even of the year) could be presented. It has to be defined if only submitting new articles should count or every change made (editing, deleting, restoring articles and so on).

- *Monthly list of top participants* – Strictly speaking, this ranking would differ from the list of top contributors for one particular reason: Contribution implicates submitting data to the community, for instance, creating new articles, editing, or restoring old ones etc., while participation can also be achieved by only viewing articles or using the search engine. In general, contributors administer something to the community which generates a surplus, but participating in an online community can merely be the registration to a wiki, for example. If no further action is taken, the community does not profit from such a behavior. A listing of monthly top participants could be calculated using only the time a user spends on reading articles and using the wiki (time spent surfing on the web-site)[37].

- *Monthly list of most viewed articles* – In Chapter 3.1.3 on page 50 a suggestion for a monthly published list of most viewed articles is described.

[37] Privacy plays an important role and therefore it has to be possible that a user who does not want to reveal private information about her/his behavior does not have to.

- *Monthly list of extra long articles* – The ideas from Chapter 3.1.5 on page 52 can be taken and they can be displayed in a monthly listing of extra long articles.

- *Monthly list of often used search terms* – The ideas from Chapter 3.1.6 on page 54 can be modified to a monthly listing of often used search terms.

Presented techniques are only a short excerpt of ideas available. To put the author of an article in the foreground, links and rankings have to be generated. The last three suggestions would list only the article instead of the author. Either the author can be named beside the article or – the better solution – calculating a new author-based ranking adapted from these lists can be made. Taking a monthly list of most viewed articles as basis, a ranking of authors whose articles were viewed most can be generated. Calculation can be done in different ways: Either all articles from an author are collected and the sum of all views are computed or the sum of all views of all articles of an author is divided by the number of her/his articles (average view-rate of an article of a specific author)[38]. With these data monthly listings can be produced that rank the author with the most overall views of her/his articles or with the highest average view-rate per article on top.

The productivity of authors cannot only be measured over time, but can also be split up in different categories or for different languages. In the MediaWiki system every article should belong to a specific category, therefore it would be self-evident to generate category-based rankings (e.g., monthly list of extra long articles in the category "information systems"). The same can be done, for instance, if articles have been written in different languages. Once again, graphics can be used to illustrate the productivity of authors.

3.2 Rating Systems

Users show their attitude towards something by taking part in a rating which is mostly adapted from a standardized scale. For this purpose, there is no need for a long questionnaire, a simple question, such as "Did you like the article?" is sufficient. Possible answers range from "Yes/No" to scales ranging from -5 (did not like it at all) over 0 (neutral) to $+5$ (liked it much). Open questions do not play a major part because they cannot be evaluated automatically (or not as satisfying as wished for). A characteristic of rating systems are credit point systems which are explained later in this chapter.

[38] A slight modification would be an algorithm which weights newest contributions most with a decreasing weighting for older articles.

Every rating system must have some sort of protection against multiple votes[39]. In general, every user has one vote and multiple votes should be restricted or should not be considered while generating the result. Multiple votes can be prohibited in different ways: by comparing usernames, IP addresses of computers, or by using session variables. Thoughts about excluding automatic voting applications should be made, too. If a user knows the voting process, she/he might be able to write an application which votes different articles automatically without human interaction.

As an article's content changes with every revision, rating must be linked to exactly one revision. It is not useful to view a rating of an older revision if an article was completely rewritten. If there are only minor corrections, like deleting two line breaks where there has to be only one, the results of the rating from the former revision should be transferred to the new one. In all other cases, by submitting a new revision, rating should start from the beginning.

3.2.1 Rating of Articles

To distinguish good written articles from bad ones, the user should have – at the end of every article – the possibility to vote for or against it. This should be done by asking only one simple question with a few standardized answer alternatives (to automatically evaluate answers). Questions could look like this: "How relevant was the information shown in this article to you?", "Did you find the information you had been looking for?", or "Was the information presented in this article easy to understand?". There can be a pool of five questions which are randomly asked each user viewing an article. Using this method, answers can be, for instance, differentiated against language or grammar skills of an author.

As it was mentioned before, possible answers could be: "Yes/No" ("Did you like the article?"), assigning points from -5 to $+5$ ("How relevant was the information shown in this article to you?") or something similar to that. It is also practicable to use graphics to count users' answers. Questions which are using non standardized answer possibilities, like "What did you like/dislike in the article?" are normally not be asked because of the bad automatic analysis of the answers. A combination of both types of answers could be realized: A question such as "Did you like the article?" can have a "Yes/No" answer choice. Furthermore, a little text field can be inserted giving the user the chance to write in short form what she/he liked or disliked. So the "Yes/No" rating points can quantitatively be calculated while the

[39] But there might be situations where more than one vote is tolerated, for example, if a very high participation rate is desired (but then results can get falsified).

author also gets the change of a personal qualitative feedback[40].

For the question "Did you like the article?" a primitive listing can be produced by counting every "Yes" vote as $+1$ and every "No" vote as -1. The same can be done with answer ratings ranging from -5 to $+5$. Rankings can be displayed as an extra page or the result of every article is displayed on the top or at the bottom of the article[41].

For a combination of quantitative and qualitative ratings an extra site linked to an article is a good solution (in addition to a special page only listing the results of the quantitative questions). Users and especially the author her-/himself can have a quick overview why users rated the article positively or negatively. As a next step the author could rewrite the article based on the ideas of the users (certainly other users can do this also on their own). If a special page is displaying the voting results like it is described here, the author should also have the possibility to answer on comments, thus giving her/him the chance for a justification. This would be a step in the direction of a discussion page (which is implemented in the MediaWiki system out-of-the-box).

The rating of articles as it is described here is much like eBay's user rating system[42]. If an eBay user purchases a good via auction from another user, both have the possibility to rate each other. They can give a positive, negative, or neutral feedback along with a short comment why they rated the other user that way. Positive valuations increases, negative decreases a user's profile by one – neutral ratings have no effect. Graphics (stars in different colors) are linked to a username indicating the status of the user's profile. For example, if a user's profile which has fourteen positive and three negative valuations is summed up, the output would be eleven and a yellow star would appear next to the username every time it is displayed. Users with 50, 1000, or even 10.000 positive valuations all have stars of different colors, thus their trustability can be seen very quickly (if it is assumed that positive and negative ratings are an indicator for a user's reliability – the more positive ratings a user has the more she/he is reliable).

[40] A combination of a simple and a detailed rating could also be imaginable where a user chooses if she/he votes using a shorter (and therefore simpler) or a longer (detailed) form. At first, there should only be the possibility for a simple vote so that every user who wants to give further information has past the standard rating procedure. After that, the user can submit further personal feedback by filling out a much more detailed form. The short rating should concentrate on criteria measured quantitatively, while the detailed rating should leave enough space for qualitative feedback.

[41] A modification would be if a user has to vote for an article first and only after that has the possibility to have a look at the rating points of this article. In that case she/he would not be influenced by a positive or negative voting result.

[42] For a detailed description of eBay's feedback system it is referred to [eBa07].

Ratings on articles can be split up in different categories, like information/content, language, grammar etc. Various questions in combination with different possible answers can gather diverse types of information about an article. That is why there can be more than one form of presenting voting results, thus leading to different kinds of listings. As ratings of articles can be linked to the corresponding authors, lists of authors in different categories, like "usefulness of the information of the articles" or "language skills of the author" can be generated. Again monthly top lists of authors can be produced as well as rankings for different wiki categories.

It has to be thought about the number of minimal votes for a representative result. If there is only one vote from one user, the result should not be published because one vote reveals nothing about the opinion of the whole community. A listing of articles which got the most votes can also be interesting (and with that a ranking of authors whose articles got most votes).

3.2.2 Estimated Effort

It is hard to estimate the efforts an author has to write an article. Some authors are better in writing articles than others and hence do not need spending so much time on a topic. Some authors have more Internet search practice and find information quicker than others.

If a system should represent the estimated effort, there certainly should be two sides: the author and the user side. Authors can vote for their article by submitting it and rating her/his efforts, for instance, from 0 to 10 (where 0 means "no effort" and 10 "very high effort"). Users can then read the article and every user has the possibility to vote for it the same way as the author. So there should be estimations from the author as well as from users. Again a number of minimal user votes according to the size of the community should be chosen. The results of author and user votes can be displayed at the same page as the article or on a separate one. Adding some text to a vote should also be made possible and authors should be able to answer on users' votes.

A listing of authors whose articles – in the users' view – have the highest effort can then be generated. Rankings of authors which have the same opinion as the community should be produced, too (authors which can estimate their effort quite like the community). But if the estimated effort targets the real effort of the author, cannot be answered. It is possible that the author has rated her/his article lower (no too high costs), because she/he does not want anybody to know that she/he has spent many, many nights on the article and the result is quite weak. If the community votes the same, the real effort does not match the estimated effort.

As a conclusion it can be said that an estimated effort can be very tricky to calculate

and may lead to false results not reflecting the real costs an author has for writing an article. It must also be considered whether estimating efforts an author has for writing an article is really a good indicator for quality. Some people can do the same things in the same quality faster than others. Therefore, a high estimated effort does not implicate a good written article because of the very subjective kind of the concern.

3.2.3 Credit Points

Credit point systems are further developments of rating systems. When a user signs up for a community she/he gets an account where, for every action taken, credit points are booked. There are many actions for which a user can receive positive or negative credit points[43]:

- *Sign up* – Even when a new user signs up for a community, she/he can already get some basic credit points. This is done because new users may do things that are not accepted by the community and earn negative points for that. New users must become accustomed to the system and a negative credit point account from the very beginning of their membership would not be very helpful.

- *Read articles* – A user may possibly earn some credit points for only surfing the community and for spending time on the web-site. Such a behavior certainly increases the usage of an online community.

- *Rate articles* – For rating articles credit points can be assigned regardless of a positive or negative review – only participation counts.

- *Submit articles* – Credit points for submitting articles must have some sort of distinctions. The length, the estimated effort, the quality, etc. of an article should influence the amount of credit points earned. Either there is a fixed value for a certain length of an article (e.g., 10 credit points per 500 words) or users rate the amount of credit points an author gets[44].

[43] Presented list is not exhaustive.

[44] If users rate articles, there is the problem when to give an author her/his credit points. Theoretically, the rating could never be closed and users can rate an article even two years after its publishing date, but then an author does not get credit points because every vote can change the amount of credit points assigned. The problem can be solved by measuring the fluctuation of rated credit points. At a certain point in time a single vote of a user has so little effect on the bottom line that the vote can be closed and credit points can be booked to the user's account. Definitely, an open vote is only closed after a certain amount of votes because ratings of articles should reflect the community's opinion (and that can only be achieved by counting as much votes as possible). Closing a vote after only a couple of ratings certainly produces wrong results which are not matching the opinion of the community (even if there are no fluctuations for a long time).

- *Edit articles* – Editing articles is much like submitting ones. For editing an article an amount of credit points should be booked based on user votes or the complexity of the changes made. Editing articles can also include deletion of wrong information – even whole articles.

- *Other forms of participation* – Users should also earn credit points when they participate in surveys or questionnaires or, for instance, if they give constructive feedback how to improve the platform. The meaning of a credit point system is that users should actively participate in a community, so every action which generates a surplus for the community should be rewarded.

If a community implements a credit point system, it is obvious to show a sorted list of users with the highest amount of credit points. There can also be listings, like "Most credit points earned during the last week" or "Most active rookies". Another idea would be that loyal users with a high amount of credit points have more rights on the system. A hierarchical structure could be implemented, thus giving the most active users more privileges. These users could also have outweigh in surveys and therefore more influence on the result. For a detailed description of so-called "career systems" read Chapter 3.9 on page 69.

If the calculation of credit points is based on a fair algorithm, the right amount of credit points are computed for the right action. Keeping the estimated effort in mind, a credit point system could reflect users' real participation in an online community very well and comparisons between different users can also be drawn easily.

3.3 Survey Systems

Survey systems are much like rating systems. The differentiation is that survey systems are not necessarily linked to an article (and therefore do not exist as long as the article exists), but can be launched and stopped whenever wanted. Moreover, on most survey systems rating is done only by selecting the item one most likes without any scale or point scheme. Every voting result should be displayed as an automatically generated chart (beside the text-only version with simply numbers and percentages).

A survey system might be a quick poll on, for example, the design of the web-site or if users must first register to submit/edit articles or not. A survey should be placed at the start page of an online community or in a prominent place where it fits logically. Such polls certainly have much more votes than ratings of articles because they are placed on important locations in an online community (possibly more then once). Therefore, surveys should be changed every week or every second week because useful results can already be obtained in such a short period and users should not be bored with the same question. It is also possible to have a pool of several questions

which are randomly displayed. To not distorting the result, every user should only have one vote.

Another poll might be rating the best article of the week. Therefore, the top five visited articles (counted by users' hits or visits) of last week should be nominated and displayed as a poll where the articles can be read and voted. Every week the results of the past week should be published and the poll should be replaced by a new one. The possibility to vote the best author of the week/month should also be given. Therefore, the visits of all articles of an author should be summarized and a ranking should be generated where the top five authors are published to be voted for.

Another approach to vote the top article of the week would be by listing the five best-rated articles. So users who read an article should have the possibility to vote for it and at the end of the week the five best-rated articles should be displayed for voting (to open the vote to a greater public). Or voting for the best article of the year should be generated by taking the best-rated article of every month of the past year as the basis for the poll. Votes should be archived and made accessible to users. Then a "hall-of-fame" could be generated by calculating, for example, the most number-one-voted author[45] (overall, per decade, per year or so).

3.4 Recommender Systems

"Recommender systems are programs which attempt to predict items [...] that a user may be interested in based on some information about the user's profile" [Res97]. Typically, these programs are implemented as collaborative filtering algorithms. "The goal of collaborative filtering [...] is to predict the preferences of one user, referred to as the active user, based on the preferences of a group of users. For example, given the active user's ratings for several movies and a database of other users' ratings, the system predicts how the active user would rate unseen movies. The key idea is that the active user will prefer those items that like-minded people prefer, or even that dissimilar people do not prefer" [Pen99].

Recommendation systems work by collecting data from users, using a combination of explicit and implicit methods. The recommendation system compares the collected data to similar data collected from others and calculates a list of recommended items for the user [Sto06]. Examples of explicit data collection include the following:

- Asking a user to rate an article on a sliding scale.
- Asking a user to rank a collection of articles from favorite to least favorite.

[45] Or, for instance, a list of the top 10 most voted authors.

- Presenting two articles to a user and asking her/him to choose the better one.
- Asking a user to create a list of articles that she/he likes (personal recommendation).

Examples of implicit data collection include the following:

- Observing the articles that a user views in an online community. To generate a page like Amazon's[46] "who viewed article A also viewed article B" recommendations.
- Keeping a record of the articles that a user submitted/edited ("users who submitted this article, also submitted article X and article Y").

Concrete implementation of a recommendation system in a wiki could be presenting a list of four or five links to other articles with the hint that other users, who viewed this article also looked at these. Or, if a user likes an article, she/he should have the possibility to view a list of all articles from this author. Another link should not only display articles submitted by this user, but also her/his personal recommendation list of articles not written by her/him (if the user has generated such a list).

By applying these methods users can find interesting articles in several ways and also identify other articles from authors they like. If a user likes an article and views the personal recommendation list of the author, she/he gets to know other articles from authors this user prefers. With the personal recommendation list users recommend other users by presenting their articles as outstanding ones.

With the information which user views which article, not only a list, like "user who viewed this article also viewed article X and article Y" could be displayed. By matching the results of viewed articles to the authors of these articles a listing, like "other authors which may be interesting to you" can be generated.

3.5 Reminder Systems

A user's participation can also be enhanced, for example, by reminding her/him of the services an online community offers and the benefits she/he gets by participating. If a user thinks her/his articles are useful and honored by the community, her/his participation rate certainly grows.

Reminder systems target this behavior by sending automated messages with personalized information to a user. It is important to send the right amount of messages –

[46] `http://www.amazon.com`, retrieved on 1 September 2007.

not too less and not to much. For example, would it certainly be annoying for a user if she/he gets an automated email with only some text, like "Thank you for your contribution!" every time she/he contributes or edits an article. This phrase can be shown on a web-site after posting a new article, but it is certainly the wrong content for an email. Top contributors would get hundreds of emails thanking them for their participation[47].

Most reminder systems are push-based, this means that the user does not have to interact to get the information. The medium best fitting the needs of a reminder system is obviously email, because usually every user must have an email address at her/his point of registration (related to the MediaWiki system). Receiving information displayed on a cell phone by using SMS should be another way, but due to restrictions in the length of the message (maximum of 160 characters) the transportable information is very low. Probably, (1) there could only be a note that the user has to gather more information on the community's web-site or (2) that the user will receive further information per email. First method is not pull-based, because the user has to react to get information. The second method is making sending an SMS obsolete because "real" information is transferred by email[48].

If a user is not very active in participating in an online community, an email with new features of the community or services which might be useful to her/him can be sent[49]. This method is similar to a newsletter where (mostly) on a monthly basis news and announcements are sent to the members of a community. In the sense of social rewarding techniques that could be: monthly top-contributors, monthly list of extra long articles, naming award of best article of the month, last ten contributed articles etc.

Another option would be to get help of a recommender system by using the history of the user's viewed articles. With the data of all users of the community new articles with similar topics could be found and a personalized list can be sent to each user. An alike method would be sending new contributed articles of the last viewed authors. Reminder services can also be linked to real-life events. If a user has viewed an article (or several articles – to be sure that a user is really interested in the topic) about Mozart in the past, she/he could be reminded of an article describing activities on the occasion of Mozart's 250th anniversary.

By using statistics a reminding email message could probably say that a user has

[47] Of course, there must be an option that allows a user to stop receiving automated email messages from the community system.

[48] It is assumed that a user checks her/his mailbox regularly.

[49] "Useful services" can be services someone has required before or new services that has connections to former needed services.

to contribute five more articles this month to be in the top 20% of the monthly top-contributors. Or, if a user earns another 100 credit points, she/he is awarded the next rank in the hierarchy (in a career system). Reminding a user of next-reaching goals is very motivating: the goals are intermediary, so next month there are other goals, and they are reachable at passable costs.
Reminding a user of new articles can also be done on the community's platform on a special user-page. This approach would be pull-based because a user has to log in to the system to view the information. If a user has logged in, she/he is already participating in the online community (not because of a reminder, but of other reasons) and that is actually wanted to be achieved by reminding her/him with some information. Consequently, reminding a user of something can only make sense by using push-techniques where she/he has not to be the first to act.

3.6 News Systems

News systems are displaying latest events, announcements, and (as the name may let assume) news from the whole community platform. This can be done, for example, by displaying a list of new users in the last 24 hours or the latest submitted articles. Such a list changes rapidly (with every new registered user or every new submitted article) so that this technique provides only a quick overview of latest events.

In news systems the following statistics can, for instance, be involved, where each item should be a link to a web-page displaying the appropriate listing:

- Users currently online
- New users (of the last 24 hours)
- Registered users
- Number of articles
- New articles (of the last 24 hours)
- Articles read (in the last 15 min)

Not only automated news lists can be generated, but also moderated news entries can be displayed on the first page of an online community informing users about latest events. Editors of moderated news entries can be administrators of a community or users with special rights (rights are granted through an administrator or in a career system they may be achieved with a special rank).

Along with these platform-wide identical news, it is possible to display also user-specific. The facility to manage a so-called "watchlist" (an out-of-the-box feature

of the MediaWiki system) should be given to a user. In a watchlist of articles a user should, for example, have the possibility to add articles from which she/he would like to receive news. Subsequently, the user should be informed about changes made to these articles or to similar articles in the same section. Of course, the submission of new articles or the deletion of existing ones similar to the ones in the watchlist should also be displayed. In another watchlist user changes should be shown, for example, new articles or changes of articles of this author, user account updates, or when the user reaches a higher rank in the hierarchy (in career systems). There is also the possibility to connect a watchlist to a reminder system, so that changes made by a user who was added to the watchlist is, for example, sent by email to the user's email address.

3.7 Personal Feedback Systems

A special form of a feedback system is a rating system (described in Chapter 3.2 on page 58) which, for example, rates an article by means of a defined scale. The combination of quantitative and qualitative feedback, for instance, by adding a text field for comments has also been discussed. The following personal feedback methods refer to such qualitative feedback.

Personal feedback is subjective and cannot be satisfactorily evaluated automatically. Nevertheless, it is a good instrument to get information a rating system can not collect: users' attitudes, opinions, ideas etc. Every article in a wiki system should give the user the possibility to submit her/his personal feedback (possibly in conjunction with some rating functions). This feedback should be listed on the author's user-page and should be readably for everyone.

For feedback that is only intended for the author(s) and which should not be read by others, the authors name and her/his email address must be known by the system[50]. There is also the possibility to give feedback for an article by using a web-based form with two options: feedback readable by everyone and feedback only readable by the author(s). So every user can decide if others can read her/his statements. If feedback to an article is displayed in connection with the author, she/he must have the chance to answer (sort of discussion page).

[50] Other contact information of a user, like her/his address or telephone/fax number can be shown, too, in order to get in contact. Certainly, not ever user might be willing to publish these very personal information.

3.8 Hyper-Linking Systems

Hyper-linking systems are based on some sort of relation between two (or more) connection points. For example, if a user views an article, other articles in the same category can be displayed to introduce these articles with similar topics. One event should lead to another, so that users are able to see and read all related articles without the need to search for them.
If an online community is tracking every visit of a user, it is possible to find users with same interests because of the articles they read[51]. Therefore, it can be useful, while a user is reading an article, to display other members of the community which have read the same article or related ones to get in contact with them. A user would then be able to contact other users with same interests based on articles they have viewed or actions they have taken ("other users with same interests"[52]). As such a public list would offend a user's privacy, she/he has to agree that her/his collected data is used to generate a listing viewable by others. Otherwise users' data must not be presented to the public.

Another idea would be displaying random user-pages[53]. The only thing needed is a link called "random user-page", which navigates the user to a randomly calculated user-page of a member of the online community. With this method the popularity of user-pages can be increased and members are getting to know each other regardless of their participation rate, rank (in career systems), or date of registration.

The former described idea can be adapted and extended by randomly taking a single user every day and introducing her/his user-page at the start page of the online community. The possibility should be granted to display a picture of the user, statistics about her/his community behavior, her/his interests, hobbies and so on.

3.9 Career Systems

Career systems are mostly based on credit point systems where users get credit points for participating in the community in some way (like contributing articles, editing existing articles, rating of articles, etc.). A user starts at the lowest rank in the career system and needs to work one's way up to get a higher rank and with that a higher reputation from the community and more rights.

The career system has to be constructed in such a way that a new member can easily climb the ladder a few steps, each step getting harder to reach as the former one.

[51] If the community is separated in different languages, hyper-linking over language borders is possible, too.

[52] One might call such a functionality "users next door" or "other users near you".

[53] Equivalent to presenting random articles.

Here it is assumed that a career system has ten ranks starting at rank one as a new member. Reaching rank three has to be nearly twice as hard as reaching level two. Therefore, a member has to be very constructive and has to participate in the community actively to achieve a high rank. A typical pyramid-system will be the result with many users at ranks under three or four (mostly 70% – 80% of the members) and only a few reaching rank eight, nine, or even the tenth rank. It must be visible which rank a user has, at best symbolized with some sort of graphics next to the member's name (like stars in different colors). A list of all members and their ranks (with ranking filter options) has also to be implemented. With every step taken on the ladder of the career system, more rights should be granted to a user. Certainly the tenth rank is the administrators level who are able to do what they like. A good graduation-scale has to be found for every rank starting from the lowest to the very highest.

One can be sure that users with a very high rank in the career system participate actively in the community, therefore have much experience and in nearly every case the user has a membership that already lasts for a couple of years. Connecting a career to a rating system, ratings of such high-ranked users can have more effect on votes than ratings from lower-ranked users. Such high-ranked users would have more influence because in most cases they know best what is good for the community. But attention has to be paid that new users (or lower-ranked ones) have enough rights and that they are not dominated by higher-ranked users.

It has to be discussed what a user has to do to stay at one level in the career system. Certainly, for wrong actions in the community negative credit points have to be booked, so that a user can fall from a higher rank to a lower one. If it is wanted that users stay active over the complete time of their membership, at every rank a user has to fulfill a certain amount of participating actions over a certain period in time to maintain her/his current rank. That means that a user at rank three, for example, has to submit five articles a month while a user at rank four has to submit ten and rate another five. If a user does not complete her/his work, she/he falls back to a rank right under her/his current position[54]. The described system is a very strict one and – if implemented – users must have the chance to take a downtime, for instance[55].

3.10 Award Systems

Awards are a good instrument to emphasize outstanding performances. They can be assigned either by vote or by users with special privileges. Most awards are assigned

[54] Of course, only after some warnings and one or two months time to do the work.

[55] A certain workload over a period of time seems in some cases only meaningful on the very highest ranks to ensure a permanent administration and support of the system.

through surveys and polls available to all members of an online community, for example, "article of the month" or "user of the month". Members can vote for a user and the user with the highest amount of votes is assigned the award, for instance, "user of the month 5/07"[56]. Awards have to be displayed graphically at the user-pages of awarded members.
Another category of awards could be a "creativity award" for superb articles with a high amount of creative potential or an award for a user's "life work". These awards can as well be assigned either through the community or by users with special rights. Administrators, users with a high rank (in career systems), or top-contributors can have the right to vote for or assign an award in these categories.

It should be possible that a user can get more than one award even in the same category. Awards are acting as figureheads for users and therefore have to be put on prominent places on user-pages. A "hall-of-fame" presenting all award-winning users may be implemented.

3.11 User-Centric Search

The idea of a user-centric search extends the built-in search engine of the MediaWiki system by filtering authors of articles. On the one hand, searching for community members must be made available. Not only the nickname, also the email address, or the real name of a user have to act as search criteria. Constraints to search only for one criteria must be implemented (by using dropdown lists of different criteria).

On the other hand, searching for articles must be limited to only the name of the author (or authors, if there are more than one). It has to be possible to search for other articles of the same author(s) as well as for other articles in the same category. If a user is searching for a document by using a text phrase (like "Mozart") and no restriction on the author criteria, it must be possible that after the result is shown, a limit to display only articles from one author may be made. To ensure that, a dropdown list of all authors of the displayed articles must be generated so that the user can choose one (or more) author(s) to display only articles from this/these author(s). It might also be useful to restrict search results to a specific group of authors, for example, only articles of top-contributors or users with a high rank (in career systems).

It has to be considered that in the MediaWiki system the search engine does not index all pages at real-time, therefore it can be possible that new articles are not being displayed in a result list.

[56] I.e. the user of the month May 2007.

3.12 Documentation and Help Systems

To ensure that every user knows the functionality of an online community a complete documentation of offered services is essential. Moreover, one must be informed of benefits by participating in the community (i.e. pages which document and explain implemented social rewarding mechanisms). Therefore, special portal pages which introduce these social rewarding methods can be generated. Documentation should be updated through users of the community to assure topicality and completeness.

When problems arise, the first contact point should be the documentation and FAQs. If a problem cannot be solved by reading the documentation, personal help should be available. In a good help system users with some sort of problems can ask other (experienced) users for help. This can be done, for instance, by email, by a discussion list, or by a categorized forum.

The same reasons that gets somebody to participate in an online community gets her/him to help other users. If a user knows she/he is rewarded for her/his help (for example, by getting credit points or by being a member of the "help team"), her/his motivation certainly increases.

3.13 Motivation through Punishment

Up to here all explained social rewarding mechanisms are based on a positive reinforcement (except loosing points in a credit point system – described in Chapter 3.2.3 on page 62 – and falling from a higher rank to a lower one in a career system – described in Chapter 3.9 on page 70) because with these techniques best motivation can be achieved. When a user knows she/he gets punished for an unwanted action, she/he does not directly get motivated to do the opposite, but she/he certainly thinks twice about her/his attitudes. A combination of social rewarding with punishment mechanisms can be imagined so that users omit unwanted behavior.

As the content of an article cannot be inspected automatically, every user can write what she/he wants. Even if the article only consists of a line like "blah blah blah". Sure, members of the community have the right to delete this article, but in the meantime the user can submit another article with the same useless content. If there are not one but a couple of users behaving in the same manner, it can be annoying to search for useless articles or (worst scenario) articles which tend to be based on the truth but consist of false information which cannot be identified as such on first sight. Therefore, the community must have tools to sanction this attitude. This can be achieved by the possibility to vote users off. If a certain amount of the community members vote against a user because of her/his behavior, she/he gets banned.

Likewise, in a credit point system a user looses credit points for unwanted actions. Blocking of a user for a certain time period (a week, a month) based on her/his offence can be a solution, too. A restriction to allow only editing her/his published articles is also a good instrument to prevent spam-attacks. Another method would be that an article gets published only if about ten other community members have reviewed it and voted for its publishing. Otherwise, the author has to rewrite the article.

Other punishment techniques are, for example, a list of users who never published a single article or never participated in anything. Such a list can be unwanted because there are certainly members which read only articles and do not post anything. Therefore, applied punishment methods depend strongly on the community.

Nearly every social rewarding technique introduced can be twisted negatively: a list of never viewed articles, list of shortest articles, worst article of the month, users with least credit points, a survey about the worst author and so on. It is self-evident that public discredits of users are not wanted. It is not a good motivational factor to publish all failures of a user to all other members of the community. Therefore, punishment methods should be applied carefully.

3.14 MediaWiki's Out-of-the-Box Social Rewarding Techniques

The MediaWiki system is bundled with some already built-in social rewarding mechanisms. Most of the functions are not "real" social rewarding techniques but with a slight modification they can be.

- *Popular pages* – A list of popular pages ranked by the number of users' views.
  Possible modification: a list of popular user-pages.
- *Current events* – Events related to the current date. Suggestions are made for news stories on the main page.
  Possible modification: current news depending on user events.
- *Recent changes* – A listing of recent changes on articles.
  Possible modification: a listing of recent changes on user-pages.
- *Random page* – A randomly picked article.
  Possible modification: random user-page.
- *What links here* – Links connecting to the site currently being viewed.
  Possible modification: which user-page links here.
- *Related changes* – A list of related changes in connection with the site currently viewed (to pages linked from this page).
  Possible modification: related changes on user-pages.

- *Gallery of new files* – A picture gallery of new added files.
  Possible modification: a gallery of new added user pictures.
- *List of blocked IP addresses and usernames* – A listing of currently blocked IP addresses and usernames with a statement why they were blocked.
- *Long pages* – A list of extra long pages (measured in bytes).
  Possible modification: a list of extra long user-pages.
- *Most linked to pages* – A list of pages to which most links are pointing.
  Possible modification: a list of most linked user-pages.
- *My watchlist* – A list of articles in the user's personal watchlist.
  Possible modification: a watchlist of user-pages.
- *New pages* – A list of recently added articles.
  Possible modification: a list of recently added user-pages.
- *Oldest pages* – A list of oldest articles.
  Possible modification: a list of oldest user-pages.
- *Statistics* – Pages with statistics about the community including user-specific statistics.
- *User list* – A list of all members of the community.

3.15 Social Rewarding Techniques Used by Wikipedia

Wikipedia uses all of the functionality described in the former chapter[57] except the method of *popular pages* and the *list of blocked IP addresses and usernames*. Furthermore, there are some extended functions which are described in short now:

- *Featured content* – Wikipedia's users can nominate articles which are "well written, comprehensive, factually accurate, neutral, and stable" [Wik07d] to be featured articles. This means that these articles are reviewed and evaluated against others and, if good enough, are published on the main page of Wikipedia as "Today's featured article".
  Possible modification: "Today's featured user".
- *Articles with most revisions* – A list of articles with most revisions.
  Possible modification: a list of user-pages with most revisions.

[57] Again most functions are not "real" social rewarding functions unless a slight modification is performed.

- *Selected anniversaries* – On the main page of Wikipedia anniversaries of famous people as well as historical events are published (according to the current date).
  Possible modification: anniversaries of community's members.

- *Statistics* – Wikipedia has a much more detailed statistic section than the MediaWiki system offers: traffic analysis, more in depth users, articles, links, database, and daily usage statistics, size of Wikipedia, search engine statistics and much, much more (it is referred to [Wik07b] and [Wik07j] for details).
  Possible modification: partly adaptation of user-specific statistics.

3.16 Social Rewarding Techniques to Implement

As said in former chapters of this thesis some social rewarding mechanisms are implemented in the MediaWiki system. Four of the introduced techniques are selected which are representing a good mixture of all methods. Another point is that the methods have to fit somehow in the MediaWiki structure, although it definitely has to be extended. The chosen mechanisms are:

- *Amount of References* – Does a quality check on the amount of references used in the article.

- *Rating of Articles* – Users can rate an article and optionally submit a comment.

- *Most Viewed Articles* – Hits or visits from users are counted.

- *Recommender System* – Due to visits of users other recommended revisions, articles, authors, or users with same interests are calculated.

These methods are a compound of top 10 lists, rating systems and recommender systems (with little influence of hyper-linking systems) and do also use third party software. The first three (Amount of References, Rating of Articles and Most Viewed Articles) are used to compute a ranking of authors[58]. Chapter 5 on page 80 focuses on these four social rewarding methods, but before that an introduction to information visualization is given which is used to present results.

[58]The fourth method (Recommender System) is something of a stand-alone technique and is not part of calculating the ranking.

4 Information Visualization

Modern society is confronted with an explosion of information. Acquisition devices like MRI-scanners, large scale simulations, and also stock trading at stock exchanges produce very large amounts of data. Visualization of data provides for researchers, analysts, engineers, and the lay audience to obtain insight in these data in an efficient and effective way. With the unique capabilities of the human visual system, interesting features and patterns can be detected in short time [Wij05, 79].

The field of computer-based information visualization draws on ideas from several intellectual traditions: computer science, psychology, semiotics, graphic design, cartography, and art. The two main fields of computer science important for visualization are computer graphics and human-computer interaction. The areas of cognitive and perceptual psychology offer important scientific guidance on how humans perceive visual information. Design deals with the process of creating artifacts well-suited for their intended purpose. Cartographers have a long history of creating visual representations that are carefully chosen abstractions of the real world.

Information visualization has gradually emerged over the past fifteen years as a distinct field with its own research agenda. The standard argument for visualization is that exploiting visual processing can help people explore or explain data. It is an active field of study because the design challenges are significant and not fully understood [Mun00, 3].

The field dealing with visualization of data that concerns people or is somehow people-centered is called *social visualization*. Judith Donath from the MIT characterizes social visualization as follows: “visualization of social information for social purposes”. Example domains for social visualization can depict conversations, email patterns, web activities, social networks, or life histories. For each domain there are many models and tools for visualizing social information [Sta05, 2f].

As this paper focuses on social rewarding techniques rather than information visualization methods, only a short description of two applied visualization mechanisms are presented in this chapter. This thesis tries to motivate users to participate actively in an online community, which is achieved by generating a ranking of authors accentuating most productive members. The rank of a user is visualized with stars while her/his participation rate is shown with sparklines. The following subchapters explain the ideas beyond the decision to use these two visualization techniques.

4.1 Rank Visualization with Stars

Using stars to generate a ranking is a traditionally well-known and an established way to give a quick indication on how good or bad something is. It is not an inven-

tion of the Internet community. Public companies, airlines, hotels, restaurants etc. are all rated by using a predefined star scale. In most Internet communities a five star ranking is applied and therefore users are very familiar with it.

Many well-known Internet platforms use this star scaling method (or a slightly different one), such as eBay, Amazon, or an Austrian price comparison web-site called Geizhals[59]. After winning an auction buyer and seller on eBay can rate each other and the overall score of these ratings is displayed by stars (for a more detailed description of eBay's rating system it is referred to Chapter 3.2.1 on page 60). On Amazon's C2C market users can rate each other, as well, and the result is displayed using a five star scale with a half star interval. Furthermore, consumers can rate a product by writing reviews annotated with stars. As far as the web-site Geizhals is concerned users can judge the retailer from which they bought a product. The judgement can be done for different categories such as availability of the product, price, shipment, delivery time etc. Grades are assigned ranching from fully satisfied to not satisfied, which are displayed by head icons that show different facial expressions from a happy smile to an angry face. Then an overall score is calculated along with the associated grade and the appropriate face is displayed beside the retailer's company logo. All rating methods provide the user with text areas for placing their personal comment as it always is a good idea to justify one's reviews regardless if it is a good or a bad one.

Another field of application is the use of stars in discussion forums. Nearly every larger forum software has some sort of ranking system implemented. As there are many forums published under an open-source license, developers can extend or modify these ranking systems, such as swapping images or create new ranking levels. Although the number of stars varies, the method is always the same. Displaying stars is in most cases related to the number of posts a user submits to the forum. A popular option is also the implementation of a career system in which members can climb the ladder by posting numerous articles. Another not so widely used method is to allocate a rank in the hierarchy according to the amount of time a user belongs to the community. That means a user can reach a higher level only by being a registered member of the forum (and not necessarily a very active one).

It was decided to implement a five star scaling to display the participation rate of a user (Figure 10 on the following page). The scaling reaches from zero to five stars – so there are six intervals. Displaying zero stars does not mean that no stars at all are shown. If, for example, a user has an amount of points that would lead to a score of two stars, five stars are shown whereas only the left-handed two of them are colored. The other three are empty showing only the outline of the stars. Displaying half stars can be activated, as well, showing only half of a star colored and the other

[59] `http://www.geizhals.at`, retrieved on 1 September 2007.

half colorless (if the amount of points leads to this result).

Figure 10: Display of three and a half star.

As stars are computed in relation to the participation rate of all other members of the community, they are a good indication for the overall contribution rate of a user. However, this does not state anything about the temporal dispersal of the contributions made. It is possible that a user submitted numerous articles many months or even years ago. For visualizing these information sparklines are used.

4.2 User Activity Visualization with Sparklines

Sparklines are "small, high-resolution graphics embedded in a context of words, numbers, images. Sparklines are data-intense, design-simple, word-sized graphics. Sparklines have obvious applications for financial and economic data, by tracking changes over time, showing overall trend as well as local detail" [Tuf06, 7f]. Sparklines are used to indicate values and trends, not for exact visual data representations. They give a good and quick overview of information compared to a long data table. The possibility to place sparklines directly in a context of words makes them very handy, too.

In this paper sparklines are used to show the participation rate of a user over a certain period of time split by predefined intervals. The contribution rate is calculated using the three social rewarding mechanisms described earlier. The darker sparklines indicate a value which is higher than the average, a brighter sparkline indicates a lower one. The appearance, intervals, heights, widths, spaces, and colors of the sparklines can be customized by the user.

Figure 11: Example of displaying sparklines.

Regarding Figure 11 one can see that user Christian has his most active period at the end of the first third and at the very end of the examined time frame[60]. With absolutely no effort another user can see that user Christian is a frequent contributor, having only a few very high contribution time frames. For the rest of the time he participates actively but not with too much effort. During his most active time there

[60]The time space displayed with sparklines can be set, for example, the whole time since the online community was launched until now or the last year or only the last days etc.

was either a topic which interested him very much or he had a lot of free time to spend by actively participating in the community. Nevertheless, by having a look at the sparklines one would characterize Christian as an average active user with frequent contributions – in other words a perfect member for online communities.

An excerpt from the data table which is displayed as sparklines in Figure 11 on the preceding page could look like this:

	score
month 1	0.2
month 2	0.4
month 3	3.8
month 4	8.2
month 5	0.4
month 6	14.9
month 7	27.1
month 8	16.4
month 9	4.0
month 10	0.5
month 11	0.1
month 12	6.6
month xx	...

Table 5: Assumed input data table for sparklines.

As it is wanted to give a quick overview of the participation rate of a user over a certain period of time, it is not a good idea to display data as it is done in Table 5. Firstly, it is impossible to integrate such a data table in a context of words without a significant loss of clearness. Secondly, it takes too long for a user to read the table. Thirdly, there is no graphical illustration which can be useful for an overview of highs and lows, outliers etc. Fourthly, the values used for calculation reveal nothing about the real contribution rates – they are only used for internal computations. Therefore, it makes no sense to display the scores.

Information visualization techniques illustrate the results of a community member's contribution rate. They are only used in addition and are displayed in a way that does not overstrain the user. As a result stars and sparklines are displayed next to a user's name giving a quick indication of the activeness of a user.

5 Implementation of Social Rewarding Techniques

In this chapter the process of implementing the chosen social rewarding techniques within the MediaWiki software along with a detailed description of the computation methods is going to be explained.

5.1 MediaWiki Software

MediaWiki is a free software wiki package originally written for Wikipedia. MediaWiki is a freely available server-based software, that is licensed under the GNU General Public License (GPL). It is designed to be run on a large server farm for a web-site that gets millions of hits per day. MediaWiki is an extremely powerful, scaleable software and a feature-rich wiki implementation that uses PHP to process and display data stored in a MySQL database.

When a user modifies a page, MediaWiki writes the change to the database, without deleting the previous versions of the page, which allows for reverting a prior version in case of vandalism or spamming. MediaWiki can manage image and multimedia files, too, which are stored in the filesystem. For large wikis with lots of users, MediaWiki supports caching and can be easily coupled with any proxy server software ([Med07b] and [Med07e]).

5.1.1 Installation of Extensions

Extensions to MediaWiki extend the functionality of MediaWiki in some way. As the MediaWiki software is open-source everybody can enhance it. The social rewarding techniques are implemented in MediaWiki as an extension called *SocialRewarding*. That means that everybody with a working MediaWiki installation could extend it by installing the extension provided.

All installed extensions can be found in the directory `[MediaWiki]/extensions` where `[MediaWiki]` is the path to the local MediaWiki installation. That is the place where all files from the SocialRewarding package belongs to.

Installing a MediaWiki extension is straight forward and does not need in-depth knowledge of the software system. For a closer look at the installation requirements and the installation process of the SocialRewarding extension it is referred to Appendix B on page 122 to read the file `SocialRewardingREADME` where needed information can be found. Beside other important things one will read that an installation script exists which will guide a user through the installation process and does the data initialization.

Installing the SocialRewarding extension should be done in a couple of minutes. The only procedure that can be very time consuming (depending on the size of the database) is the optional initialization of already existing data of the MediaWiki installation. Initialization of already existing data as well as collecting and linking new data is necessary as the SocialRewarding package can only work with data provided after its installation.

It is highly recommended that one initializes existing data because otherwise the SocialRewarding package starts with absolutely no data and therefore needs some time to deliver good results. As data is collected only by the participation of users it can take some hours or even days (depending on the contribution rate of community's users) to reach a level at which enough information is collected to display representative results.

Attention: Before starting the initialization process one has to set up the SocialRewarding extension to meet one's requirements. This can be done by editing the configuration file named `SocialRewardingConfigDetail.php` or `SocialRewardingConfigShort.php` (depending on wether one would like to have a short or a detailed description of the configuration parameters). To enable one of the configuration files one has to edit the file named `SocialRewarding.php` and un-/comment lines 43 or 44 where the short and the long versions of the configuration files are loaded[61]. In Appendix C on page 124 an example of a detailed configuration file is shown.

5.1.2 Special Pages

Special pages are pages that do not fit in the regular MediaWiki structure (no articles or discussion pages). Some special pages are created by the MediaWiki software on demand. Everybody can extend MediaWiki by developing her/his own special pages. They are located in their own namespace *Special:* and are not directly editable such as other pages. Some special pages depend on the preferences that have been set by a user, for example, the number of titles which is displayed on a user's watchlist [Med07a]. There are numerous special pages created by the MediaWiki installation, for example, a user can view articles having the highest number of revisions, a gallery of new files, oldest pages, short pages, a user list, or a random page and many more.

The pages created by the SocialRewarding extension (e.g., "Ranking of Authors", "Recommender System" etc.) are all special pages and accessible through the list on the special pages site. The names of the special pages are all starting with *Social*

[61] The short version is loaded by default.

Rewarding. As the list of special pages is sorted alphabetically the sites from the SocialRewarding extension are all next to each other. In detail the SocialRewarding package implements the following special pages[62]:

- *Social Rewarding: Amount of References of Articles (unweighted, directly from DB)* – This page displays the calculated points for the social rewarding mechanism Amount of References for articles. The listing is retrieved directly from entries in the database and is *not* weighted with the algorithm from the social rewarding technique Amount of References. The articles can be weighted according to the number of references. This means that the calculated points are divided by the number of references of the article. Therefore, for example, a reference with 100 points is weighted stronger than two references with 50 points each (although both articles would have an unweighted sum of 100 points).

- *Social Rewarding: Amount of References of Revisions (unweighted, directly from DB)* – The same method as described before except that revisions are taken as calculation basis instead of articles.

- *Social Rewarding: History of Ranking of Authors* – As there is a caching mechanism which stores all computed data at a certain point in time, a history of ranking of authors can be generated[63]. This site displays a list quite similar to the special page *Social Rewarding: Ranking of Authors* which is described below, but with an option to select a date in the past to display calculation results.

- *Social Rewarding: Installation* – After implementing the SocialRewarding package as described in Appendix B on page 122 the page shows the instructions for setting up the extension. After a successful installation this special page disappears.

- *Social Rewarding: Most Viewed Articles (unweighted, directly from DB)* – This special page shows a list of most viewed articles which is retrieved directly from the database and is *not* weighted with the algorithm from the social rewarding technique Most Viewed Articles. Visits or hits are displayed just as set up in the configuration file.

- *Social Rewarding: Most Viewed Revisions (unweighted, directly from DB)* – Here the same listing is generated but with revisions as computation basis instead of articles. Revision numbers are displayed next to the page title using the following syntax: `[R1234]` where `1234` is a wildcard for the revision number.

[62] On every special page of the SocialRewarding package self-explaining display parameters can be set which are not explicitly mentioned here.

[63] Only when the cache is activated and it uses the database.

- *Social Rewarding: Ranking of Authors* – The main site presenting a ranking of authors calculated with the developed social rewarding methods. The user can select wether stars, sparklines, and/or score points should be displayed next to a user's name. Furthermore, all combinations of the three social rewarding techniques Amount of References, Rating of Articles and Most Viewed Articles can be chosen as computation basis of the listing.

- *Social Rewarding: Rating of Articles (unweighted, directly from DB)* – As users can vote for or against an article these points are summed up, divided through the amount of voters, and displayed here. This mechanisms does *not* weight points using the social rewarding method Rating of Articles.

- *Social Rewarding: Rating of Revisions (unweighted, directly from DB)* – The same technique as for Rating of Articles but on a revision basis.

- *Social Rewarding: Recommender System* – This special page lists recommended revisions, recommended articles, recommended authors, or authors with same interests for all members of the community in a defined timeframe.

- *Social Rewarding: User List* – This page was adapted from the MediaWiki special page `SpecialListusers.php`. As an extension to the original site, stars are displayed beside a user's name.

Figure 12 on the next page is exemplarily displaying the relations of the special page Most Viewed Articles[64]. The class `SpecialPage` manages all special pages within the MediaWiki software. As the special page Most Viewed Articles generates a list the class `QueryPage` is used which is designed especially for listings. The object `SpecialSocialRewardingMostViewedArticles` is a subclass of the class `PageQueryPage` (designed for very simple queries) which itself is a subclass of `QueryPage`. Therefore `SpecialSocialRewarding-MostViewedArticles` inherits all behaviors from both superclasses.

Most special pages within the SocialRewarding package are designed like this, with a few exceptions:

- Classes `SpecialSocialRewardingAuthorsHistory` (history of rankings of authors), `SpecialSocialRewardingAuthorsRanking` (ranking of authors), `SpecialSocialRewardingInstall` (installation page) and `SpecialSocialRewardingRecommenderSystem` (recommender system) are not using `QueryPage` or subclasses of it because they do not

[64]For a simpler displaying of the figure some method parameters are not shown. Three dots (...) are indicating that this function has further parameters. Moreover, objects in Figure 12 on the following page are referring to MediaWiki 1.6.7. Newer MediaWiki versions can have slightly different methods and parameters.

need it (in case of the installation page) or they implement an own listing procedure.

- Class `SpecialSocialRewardingListUsers` (lists users and displays stars) inherits only behavior from class `PageQuery` (and not from `PageQueryPage`).

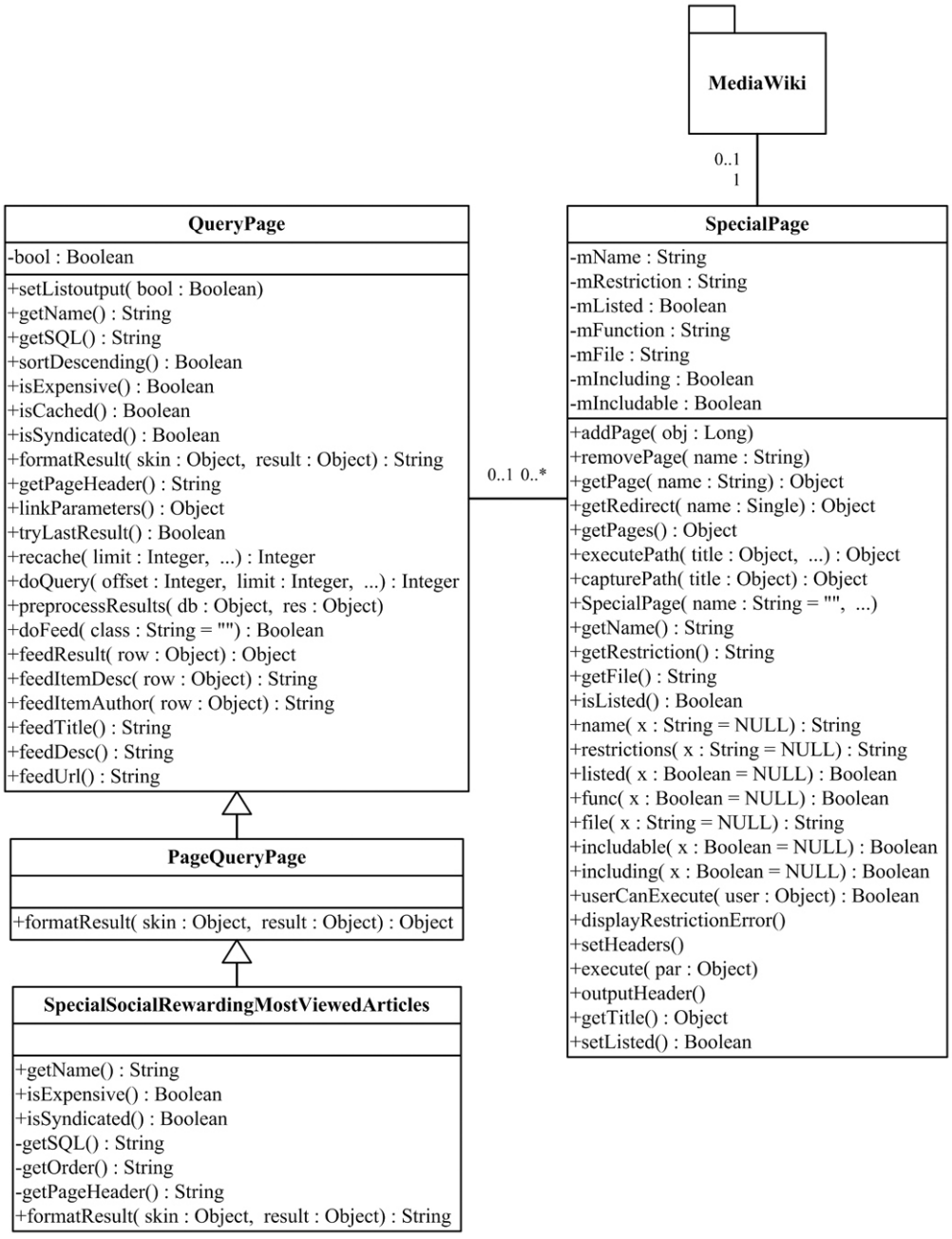

Figure 12: UML class diagram of related special page objects.

5.1.3 Hooks

"MediaWiki provides several hooks that can be used to extend the functionality of the MediaWiki software. Assigning a function (known as an event handler) to a hook will cause that function to be called at the appropriate point in the main MediaWiki code, to perform whatever additional task(s) the developer thinks would be useful at that point. Each hook can have multiple handlers assigned to it, in which case it will call the functions in the order that they are assigned, with any modifications made by one function passed on to subsequent functions in the chain" [Med07c]. The SocialRewarding package makes use of hooks for several actions, for example, if an article is saved, a function is called where references in this article are found and checked using Google's SOAP search API (Beta)[65]. Then the results are stored in the database.

5.1.4 Markups

"The wiki markup is the syntax system you can use to format a Wikipedia page" [Wik07m]. As MediaWiki is open-source everybody can extend the wiki markup by creating her/his own tags. The SocialRewarding extension defines some wiki markups which load predefined text in an article, for example, a voting form where a user can submit points and a comment. The SocialRewarding package defines XML-styled tags which can be used in the wiki editor as follows:
`<tagname> some text </tagname>`

5.1.5 Database

As the SocialRewarding extension does not affect all tables in the database, only significant ones are explained. In fact, there are only four MediaWiki tables which are related to the SocialRewarding tables: `revision`, `page`, `user` and `text`. By having a look at Figure 13 on the next page the relationship between these tables can be seen[66].

On creating a new article a record is stored in the table `page` where the title, the counter and so on is placed. As one article can have many revisions also a new revision is created with an entry in the table `revision` which has a link to the specific page (`rev_page -> page_id`). The revision record stores, for example, the timestamp or the author of the revision which is a link to the table `user` (`rev_user -> user_id`) where all user-specific information is located (user-

[65] `http://code.google.com/apis/soapsearch/`, retrieved on 1 September 2007.

[66] Database tables in Figure 13 on the following page are referring to MediaWiki 1.6.7. Newer MediaWiki versions can have slightly different table columns.

name, real name, password, email address etc.). The article's text from every revision is stored in table `text` with a link to the `revision` table (`rev_text_id -> old_id`). Therefore, every revision belongs to one page and must have an author and a modified text in relation to the old revision[67]. If a new revision is created, only entries in the table `revision` and `text` are made.

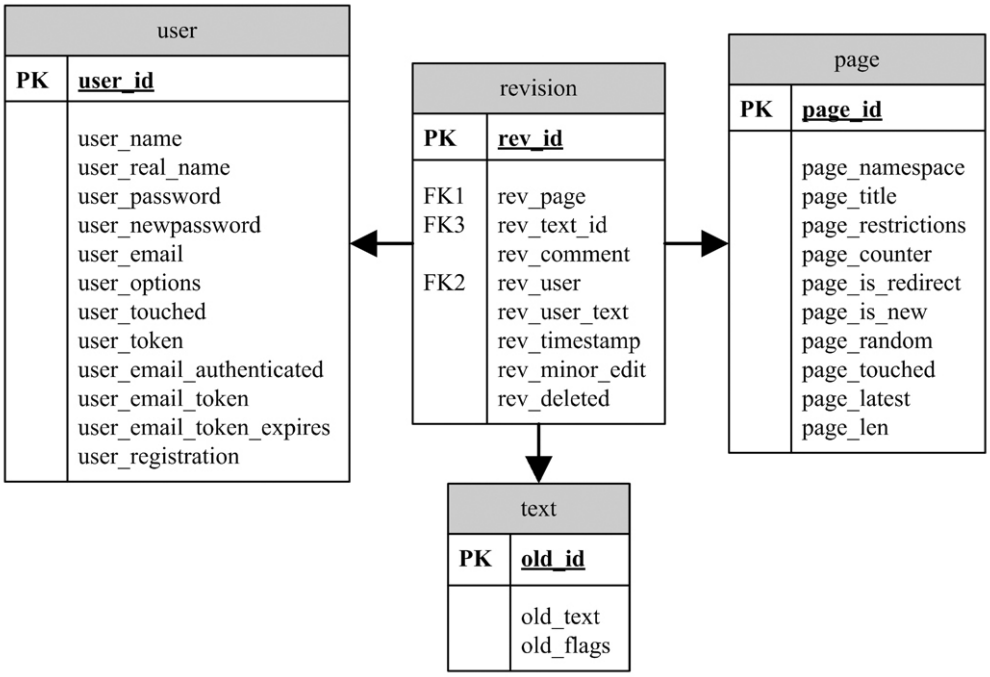

Figure 13: ER model of significant MediaWiki tables.

The PKs of all tables (displayed bold in Figure 13) are automatically generated, serially numbered integers. The FKs in table `revision` refer to the other three PKs by storing their value. Important columns to know are `page_namespace` which defines the namespace of an article. That means it defines the article as a main page, a discussion page, a user-page and so on. The column `page_counter` is only important for initializing data and is not used for storing users' visits or hits for the SocialRewarding extension (separate tables are defined). The current revision is saved in the field `page_latest`. The field `page_len` stores the length of the current revision in bytes. The column `rev_timestamp` stores the creation date and the time of the revision in the following format: `YYYYMMDDHHMMSS` (for example `20061102181936`). All other columns are self explaining or not important as far as it concerns the SocialRewarding package.

[67] MediaWiki prevents saving a new revision without any changes made.

5.1.6 Objects

The MediaWiki is an object oriented software. The most important objects regarding the SocialRewarding extension are displayed in Figure 14. For simplicity reasons, only important relations between objects are displayed and methods and variables are left out[68].

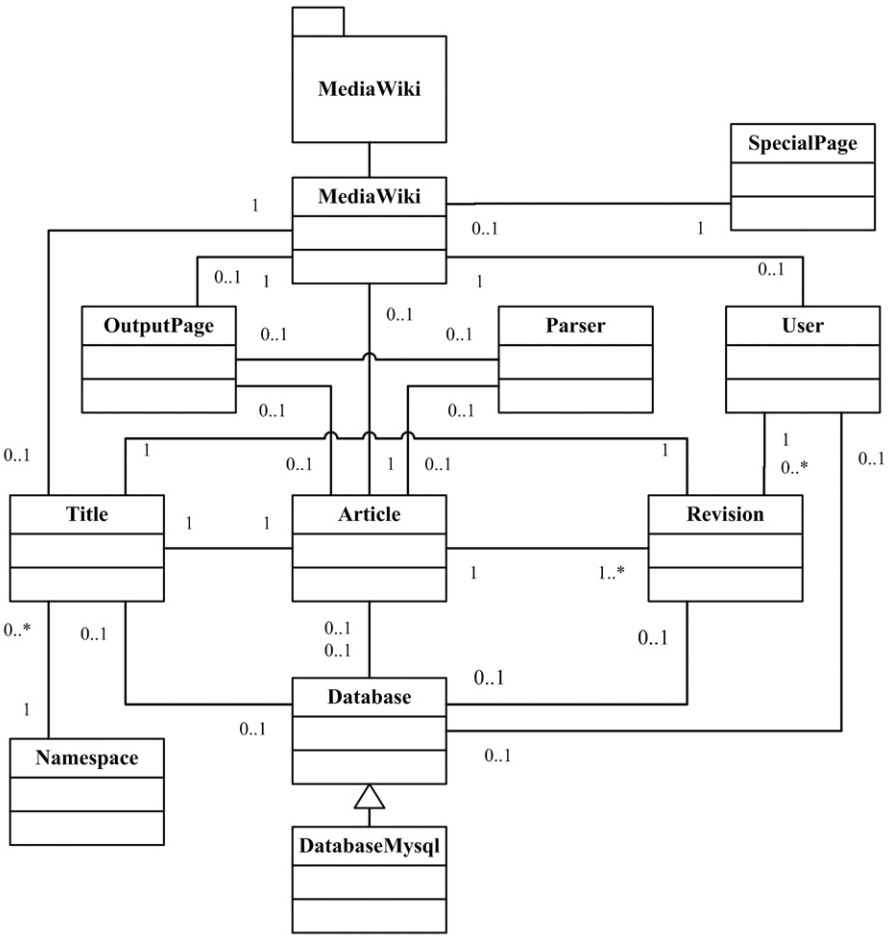

Figure 14: Simplified UML class diagram of significant MediaWiki objects.

The `MediaWiki` package contains the base class called `MediaWiki` which is the starting point for loading all other classes. When a user hits the MediaWiki start page, the `MediaWiki` object instantiates an object `Article` which does all the necessary routines to load the front page, for example, load the text from the database[69] by using class `Database` (a generalization of the class `DatabaseMysql`

[68]For a full documentation of MediaWiki's source code (files, classes, namespaces) it is referred to [Med07d].

[69]If caching is enabled, it can happen that loading data from the database is not necessary.

because of additional support for PostgreSQL and experimental support for Oracle databases). As every article consists of at least one revision there is a strong link between these classes. The `Title` object handles names of articles for which different spellings for different domains are possible (there is a difference between displaying a title of an article as an URL or as a headline in the article). `Title` objects can be linked to many different namespaces which is handled through the class `Namespace`. Every text a user contributes to the wiki is parsed using the `Parser` object to formate the text in wiki style, for example, internal/external links, images, markups, or simple text formatting (bold, italic, headlines etc.). For embedding the text to be displayed in HTML a class `OutputPage` is used. Every article contributed or every revision edited must be linked to a user. These user interactions are handled by the `User` object. Finally, there is the class `SpecialPage` which registers and handles all special pages.

Classes displayed in Figure 14 on the previous page are not limited to the functionality described above. Only some example routines have been picked out from every object to get a better understanding. For a full description of the MediaWiki classes it is referred to [Med07d] or at the source code of relevant files.

5.2 Amount of References

As described in Chapter 3.1.7 on page 54 counting the amount of references used in an article is quite a common way to get information about the quality and publicity of the article. In the MediaWiki software a technique has been implemented which counts Internet references specified in an article according to three different criteria:

- *Size of the reference* – Count all web-pages of which the entire referred website consists of.
- *Link size of the reference* – Count the number of links pointing to a reference.
- *Link size of the article* – Count the number of links pointing to the article.

All three criteria can be weighted so to weight a specific criteria stronger than the other or to completely disable one if it should not be used[70]. Checking the references is done using Google's SOAP Search API (Beta) at the time a user posts an article or submits an edit. The corresponding method is called `SocialRewardingReferences()` and is invoked by the hook `ArticleSaveComplete`[71]. As the Google search is queried in realtime, a minimal delay can be recognized.

[70] For a complete configuration list it is referred to Appendix C on page 124.

[71] To find the files in which these functions are defined and also for further localization of classes and methods it is referred to Appendix D on page 141.

The class handling the Google query is called `GoogleSearch` and is displayed in Figure 15.

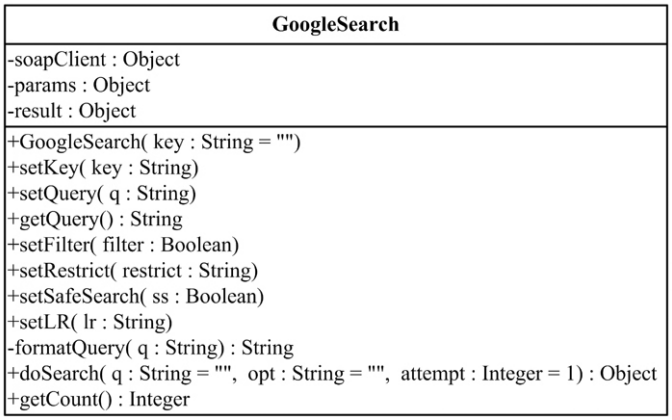

Figure 15: UML class diagram of object `GoogleSearch`.

The method `SocialRewardingReferences()` stores collected data (through class `GoogleSearch`) in the database. Figure 16 displays the ER model for the table `sr_references` which is the data container for the method Amount of References. For all social rewarding mechanisms the according data storage is based on revisions, therefore the PK is the revision ID (`rev_id`) of the article. In columns `size` the size of the reference, in `link` the link size of the reference and in `self_link` the link size of the article are stored. As `size` and `link` sum up all web-pages from all references as a single number, the amount of references must also be stored to compute a mean value which is done in column `count`.

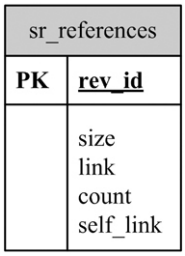

Figure 16: ER model of table `sr_references`.

The calculation process for assigning points to authors is split up in two parts. First of all, for all three social rewarding methods points are computed on a revision basis and in a next step these scores are assigned to an author where certain criteria (like length of the article, weighting of the specific social rewarding technique etc.) are crucial for the amount of points assigned. An introduction of the first part of the computation process is explained in the particular subchapters of the social rewarding methods. Chapter 5.6 on page 99 explains the second part of the computation

process and focuses on the mathematical methods used.

The entire configuration of the whole SocialRewarding package (that includes also the parameters of the computation process) can be done in the config file where also an in-depth explanation of all variables is given. An example configuration file is displayed in Appendix C on page 124. If one wants to know which parameters are available and how they must be configured, she/he should have a look at these pages. In the specific subchapters of the social rewarding methods no further explanation of these settings are issued.

For calculating points for a specific revision the value for each of the three criteria of this revision is compared against the specific average value for all revisions. The result is a percentage value for every criteria. If, for example, the factor *size* of a variable has a value of 0.7 then the size of the reference of this revision has an amount of 70% compared to all sizes of references of all revisions existing in the wiki. So this revision has web-pages as references whose sizes are shorter than the average size of all web-pages used as references. Therefore, this revision gets less points for this criteria (for the remaining two criteria the calculation method is the same) according to a configurable scale. In the example configuration file in Appendix C on page 124 the scale for computing points for the social rewarding mechanism Amount of References looks like this:

$< 15\%$	0 points
$>= 15\% \ \& < 33\%$	1 point
$>= 33\% \ \& < 66\%$	2 points
$>= 66\% \ \& < 100\%$	3 points
$>= 100\% \ \& < 150\%$	4 points
$>= 150\%$	5 points

Table 6: Sample Amount of References pointing scale.

Therefore, in the example above the criteria *size* would get 3 points. These points and the others from the remaining two criteria have to be weighted and summed up according to the settings in the config file. The outcome of this computation process is the amount of points for one revision for the social rewarding technique Amount of References.

For calculating points according to the technique Amount of References (like it is described in the former paragraphs) a class called `RewardReferences` (Figure 17 on the following page) is used. At first, all data needed is loaded from the database and stored in a multi-dimensional array. To calculate the score the data from the revision for which the computation is performed has to be retrieved as well as the average values from all revisions in the database. Then the points for the revision have to be calculated (as it was explained earlier). Class `RewardRefer-`

ences handles exactly this process and method getArticlePoints() returns the points for the particular revision ID (rev) which has to be passed.

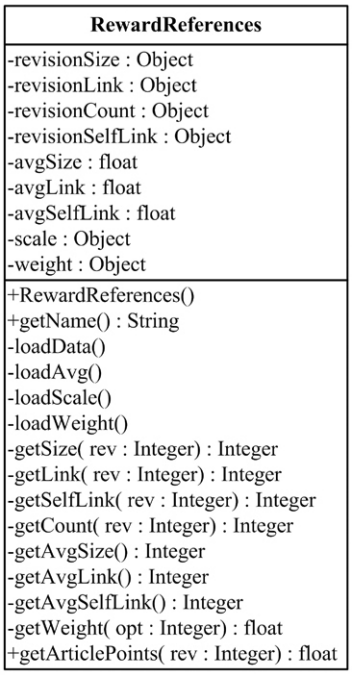

Figure 17: UML class diagram of object RewardReferences.

The values stored in the database for the social rewarding method Amount of References can also be displayed in every article by making use of markups[72]. By inserting the tag <SocialRewardingReferences /> (short form of opening and closing tag) a text is added to the article which can be set up in the config file. The text displayed can, for instance, look similar to the following: "There is a total of 4582 links pointing to the references (4) in this article. The total size of the sites used as references in this article is 21719. 583 links are pointing to this site (until 16:23 11 May 2007)." The format of displayed text can be configured in the config file as a general template for every article. Furthermore, every author of a revision can change the text individually:
<SocialRewardingReferences>Links to this site: $4
</SocialRewardingReferences>.

This results in an text output such as "Links to this site: 583". The variable $4 is thereby substituted with the value of links to the site retrieved from the database. The variables are: $1 (link), $2 (count), $3 (size), $4 (self_link), and $5

[72]Beside using markups all other outputs of all calculation processes are displayed using special pages. It is referred to Chapter 5.1.2 on page 82 to find more details on which special page one can find which information.

(current timestamp).

Hooks are used to format embedded wiki-markups. In this case the hook named `SocialRewardingReferences` calls method `SocialRewardingReferencesMarkup()` to generate the result. As every social rewarding method hast some output to be displayed in an article markups exists for each of them (see underlying subchapters). In the configuration file it can be set up whether these markups are to be inserted at installation time in all existing articles and on every newly created article automatically. Therefore, a hook called `ArticleInsertComplete` exists which calls function `SocialRewardingAutoMarkup()` to handle the auto-markup insertions.

As the complete UML model for all classes of the SocialRewarding package is too big to display, every class is presented on its own in its specific subchapter (like Figure 17 on the preceding page). A simplified UML class diagram of significant objects belonging to the overall package is displayed in Figure 18 to show their dependencies[73].

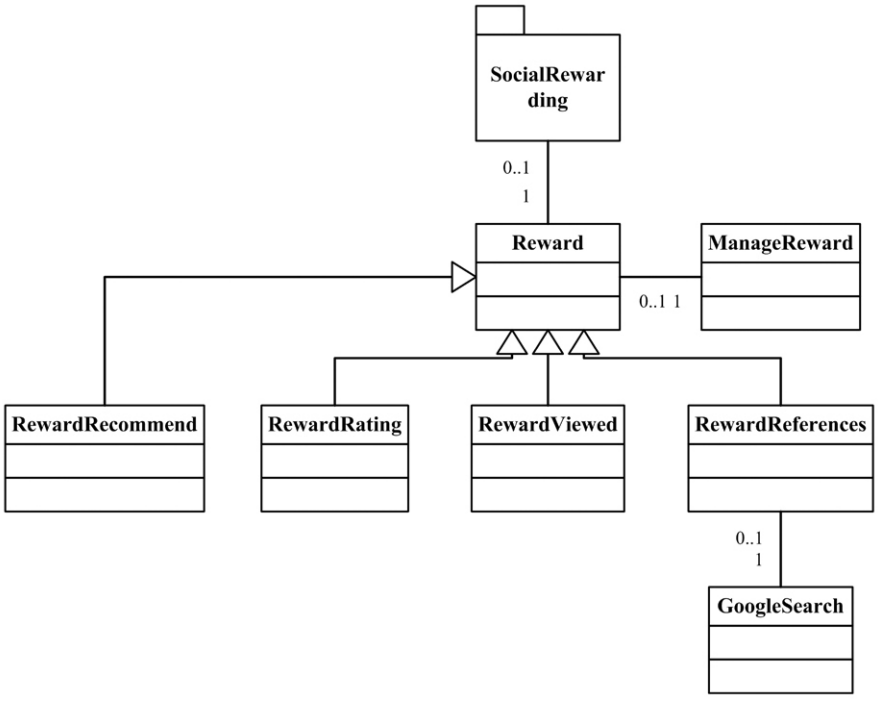

Figure 18: Simplified UML class diagram of the SocialRewarding package.

[73]Figure 18 disregard developed MediaWiki related functions such as hooks, markups, or special page classes and therefore does not display them.

The main class of the SocialRewarding package is called `Reward`. There, all computations for all authors are performed as well as the calculation of stars and sparklines. That means object `Reward` collects, computes, displays, manages, and stores all data[74]. The class `ManageReward` handles the caching algorithm (which caching method is used, when to cache etc.) – basically the caching of class `Reward`. For every social rewarding method there is a subclass of `Reward` which collects and calculates data for its own mechanism. In the case of `RewardRecommend` (Recommender System) everything (collecting, computation, and displaying of data) is done in this class. For anything else the subclasses `RewardRating` (Rating of Articles), `RewardViewed` (Most Viewed Articles), and `RewardReferences` (Amount of References) compute points for their own mechanism on a revision basis. These points are then collected in class `Reward` where the calculation process takes place on the basis of authors. Displaying results from these processes are also handled there. As it can be seen in Figure 18 on the previous page object `RewardReferences` uses a class called `GoogleSearch` which manages all Google related methods, such as the SOAP interface, search querying, or result handling.

5.3 Rating of Articles

A user-centric evaluation of published articles is still missing in the MediaWiki software. Therefore, an open rating system is implemented where users can vote for or against an article (and optionally add a comment) by making use of a predefined pointing scale. The ideas behind a social rewarding method of rating an article is described more precisely in Chapter 3.2.1 on page 59.

The SocialRewarding extension inserts a voting form in an article where a user has the possibility to assign points and optionally add a comment. Comments are then displayed on the discussion page, where the author has the chance to answer. Inserting the voting form is again done by using the MediaWiki markup language, for instance:

```
<SocialRewardingRatingOfArticles comment=true size=20
maxlength=70 buttoncaption="VOTE" popup=true popupmsg=
"Thank you for your vote"></SocialRewardingRatingOfArti-
cles>
```

This example results in displaying a voting form where comments can be inserted (`comment=true`) with a text field length of 20 (`size=20`) and a maximum text length of 70 characters (`maxlength=70`). The button caption is set to "VOTE" and after voting a pop-up appears (`popup=true`) displaying the text "Thank you

[74] In the meaning of storing all data in an object not in the database or filesystem.

for your vote". With this markup instrument every author can adjust the design of the voting form individually. If an author does not insert a markup or omits some markup tags, the extension searches for master variables in the configuration file which can be set to adjust the voting form design globally.

Results of the voting scores can be displayed in two ways. Firstly, if comments are activated in the configuration file, it can also be set if rated points of an author are displayed next to her/his comment. Secondly, an overall score can be shown in the article by inserting the following markup text:
`<SocialRewardingRatingPoints>This article was rated $1 time(s) with a score of $2 point(s) (until $3).</SocialRewardingRatingPoints>`

This will insert text and substitute the three variables as following: `$1` how often this revision has been rated, `$2` overall average of assigned points, `$3` time and date limit of data used for the calculation. If no text is inserted between the tags, typing `<SocialRewardingRatingPoints />` has the same effect (but this loads the standard markup text specified in the configuration file).

The markup for displaying the rating form is transferred into real text in method `SocialRewardingRatingOfArticlesMarkup()` which is invoked by hook `SocialRewardingRatingOfArticles`. The points are displayed in an article by function `SocialRewardingRatingPointsMarkup()` and hook `SocialRewardingRatingPoints`.

Furthermore, the whole design of the voting form can be defined in the configuration file (there are even more variables to be specified). Voting points are collected on a revision basis, therefore, every time a new revision is committed voting points are set to the initial status. The function for storing data into the database is called `SocialRewardingRating()` and is invoked by hook `SkinTemplateContentActions`[75]. The according database table schema is displayed in Figure 19.

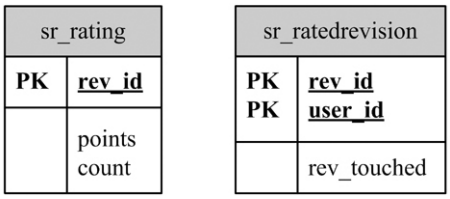

Figure 19: ER model of tables `sr_rating` and `sr_ratedrevision`.

[75] A hook is needed that does not produce an endless loop, but is called every time an article is loaded.

Table `sr_rating` stores the aggregated points (column `points`) that all users have assigned to a revision. To calculate average points for one revision the number of votes have also to be stored (column `count`). As there is the possibility to activate mechanisms for preventing multiple votes, the voting time of a specific user and revision is stored in table `sr_ratedrevision`. Beside other options in the configuration file, variables can be set which define a timeout between two votes of the same user. Multiple votes can be turned off completely, as well.

Every social rewarding method used to generate a ranking of the most active authors must be comparable, thus producing an numeric output between 0 and 5. In the examples a scale for the technique Rating of Articles is used which already matches these requirements[76]. The class for the overall computation process for this mechanism is called `RewardRating` and is displayed in Figure 20.

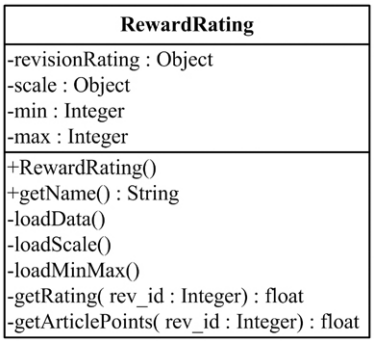

Figure 20: UML class model of object `RewardRating`.

Calculation is performed by collecting all necessary data from the database (method `loadData()`) and storing it in a multi-dimensional array. The method `getArticlePoints()` returns the average rating points for the submitted revision. If necessary, normalization is also applied.

5.4 Most Viewed Articles

The third criterion for generating a listing of the most productive authors is the social rewarding method named Most Viewed Articles. There, visits or hits of users on a revision of an article are counted (for more information it is also referred to Chapter 3.1.3 on page 50).

In the configuration file an administrator can specify the counting method (`hits` or `visits`). If the method `visits` is chosen, a timeout within which another visit

[76] If any other rating scale is defined in the configuration file, the extension normalizes points to this schema (minimum-maximum normalization).

from the same user is not counted has to be set, as well. The option whether hits or visits from the author should be counted has to be specified, too. Collected data are stored in the database schema displayed in Figure 21.

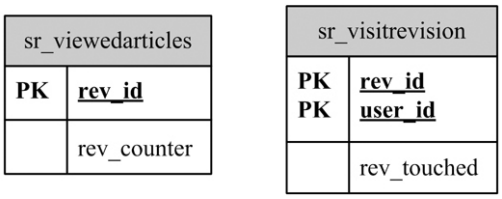

Figure 21: ER model of table `sr_viewedarticles` and `sr_visitrevision`.

Every hit or visit increases the counter in column `rev_counter` in table `sr_viewedarticles` by one. The table `sr_visitrevision` provides information on who has accessed which revision at which time. Data storing is done by method `SocialRewardingMostViewed()` and invoked by hook `ArticlePageDataBefore`.

For this social reward technique again an output of points between 0 and 5 is needed. Therefore, the collected data of one revision is compared with the overall data collected for the whole community. This means the hits or visits generated by one revision is compared to the average value of all revisions. The outcome is a percentage number which has to be looked up in a mapping table for the according point value. In this example the configuration file in Appendix C on page 124 includes the scale for calculating points for the social rewarding mechanism Most Viewed Articles as follows:

$< 25\%$	0 points
$>= 25\% \ \& < 50\%$	1 point
$>= 50\% \ \& < 100\%$	2 points
$>= 100\% \ \& < 200\%$	3 points
$>= 200\% \ \& < 300\%$	4 points
$>= 300\%$	5 points

Table 7: Sample Most Viewed Articles pointing scale.

This means that if a revision, for instance, has more than two and less than three times the hits or visits of the mean hits or visits of all revisions in the wiki, it is assigned 4 points. Of course, this scale can be adjusted in the configuration file.

Class `RewardViewed` (Figure 22 on the following page) manages the computation process for this social rewarding method. Again all necessary data must be loaded from the database and are stored in run-time variables. As the design of the technical implementation of all social rewarding methods is nearly the same,

this technique also provides a method `getArticlePoints()` which returns the points for a specified revision.

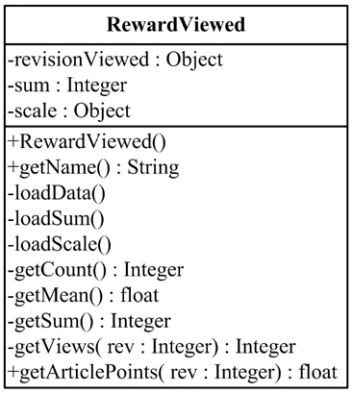

Figure 22: UML class model of object `RewardViewed`.

Hits or visits of an article can be displayed in the article by using the wiki-markup once again:
`<SocialRewardingMostViewedArticles show=true></SocialRewardingMostViewedArticles>`

Argument `show` must be set true to display an output like "This article has been accessed 18122 time(s) (until 17:10, 11 May 2007)". To change the standard output defined in the configuration file, one can insert a self-written text between the two tags. Attention must be payed that by inserting another text there are two variables which can be defined and are substituted by the right value. Variable `$1` defines the access rate and `$2` date and time. The markup function for this technique is named `SocialRewardingMostViewedArticlesMarkup()` with corresponding hook `SocialRewardingMostViewedArticles`.

By now the three social rewarding methods which are used to generate a ranking of the most productive authors has been discussed. As mentioned before there exists a fourth technique which does not interfere with the ranking calculation, but is rather a stand-alone solution. This mechanism – the Recommender System – is explained in the next chapter.

5.5 Recommender System

The implemented Recommender System works by implicitly collecting data from users while they are browsing the wiki. By comparing the data from one user to data from the others similarities are retrieved to calculate a list of recommended items for that particular user. As an outcome recommended revisions, articles, or authors and authors with the same interests can be displayed. For more information

on recommender systems it is referred to Chapter 3.4 on page 64.

For this social rewarding method another database table is needed which is displayed in Figure 23. There, the revision and user ID is stored along with a timestamp which denotes the time a user has visited an article. This is necessary for limiting the output of the recommendations in the special page to a smaller amount of recommendations for the last hour, last day, last week, or last month. The data insertion is again handled through a function called `SocialRewardingRecommend()` which is invoked by hook `ArticlePageDataAfter`.

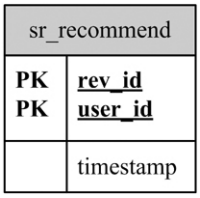

Figure 23: ER model of table `sr_recommend`.

The basis for the calculation is a list of top-articles of an author. The amount of top-articles the calculation is based on can be configured. Top-articles of an author are articles which she/he has visited most often. Every article in the top-article list of an author is compared with articles in the top-articles list of all other authors to find similarities. The outcome is a list of articles which are weighted with the following criteria: their ranking in the top-articles list of the specific author and their position in the top-articles list of other authors. If this list is sorted in ascending order, the most recommended articles, revisions, or authors are at the top. It can be configured how many recommendations are displayed. To get a list of users with the same interests the recommended list of a specific user has to be compared with other recommended lists to find similarities. Users with the same interests are users with the most matches, whereas the match criteria again are weighted according to configured settings. Finally, the number of items to be displayed in the list can be defined.

The UML model of the class `RewardRecommend` implementing the social rewarding method can be seen in Figure 24 on the next page. After the usual data fetching process, calculation is performed according to settings in the configuration file. If the output of a calculation is to be displayed in an article the following markup has to be inserted:
`<SocialRewardingRecommend method=author> </SocialRewardingRecommend>`

The attribute `method` determines one of the following techniques: `revision`, `article`, `author`, or `interestedAuthor`. If the attribute is omitted, the according setting in the configuration file is applied. Again, a user-defined text can be

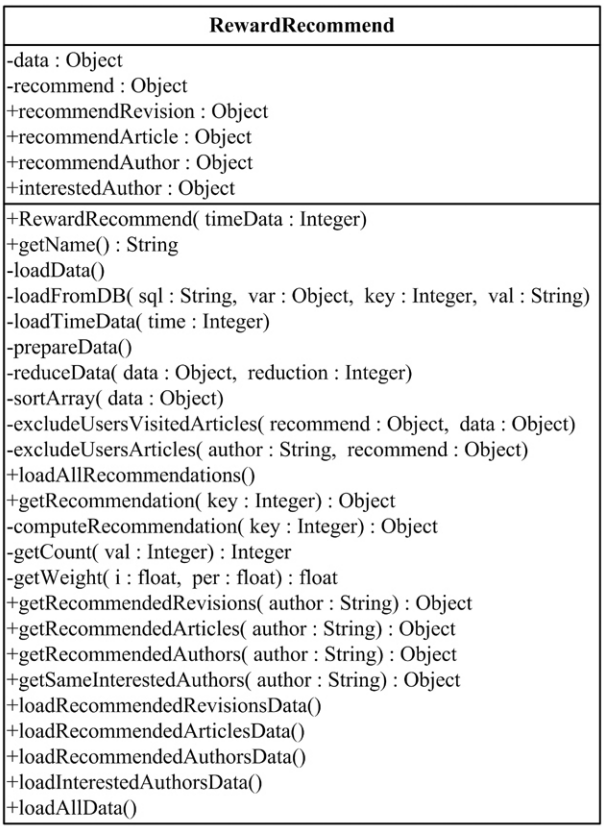

Figure 24: UML class model of object `RewardRecommend`.

inserted between the two tags which is displayed then. The method `SocialRewardingRecommendMarkup()` and hook `SocialRewardingRecommend` are responsible for this particular functionality.

5.6 Ranking of Authors

5.6.1 Technical Design

To generate a ranking of authors the results of the formerly discussed social rewarding methods Amount of References, Rating of Articles and Most Viewed Articles are merged to generate a single score[77]. The main class which handles these methods is called `Reward` and is displayed in Figure 25 on the next page. The class `Reward` is the superclass of the four social rewarding techniques (the fourth is the Recommender System). In this class the results of all other calculations of other classes are collected and processed as well as the output of stars and sparklines.

[77] The used computation process is described in Chapter 5.6.2 on page 102.

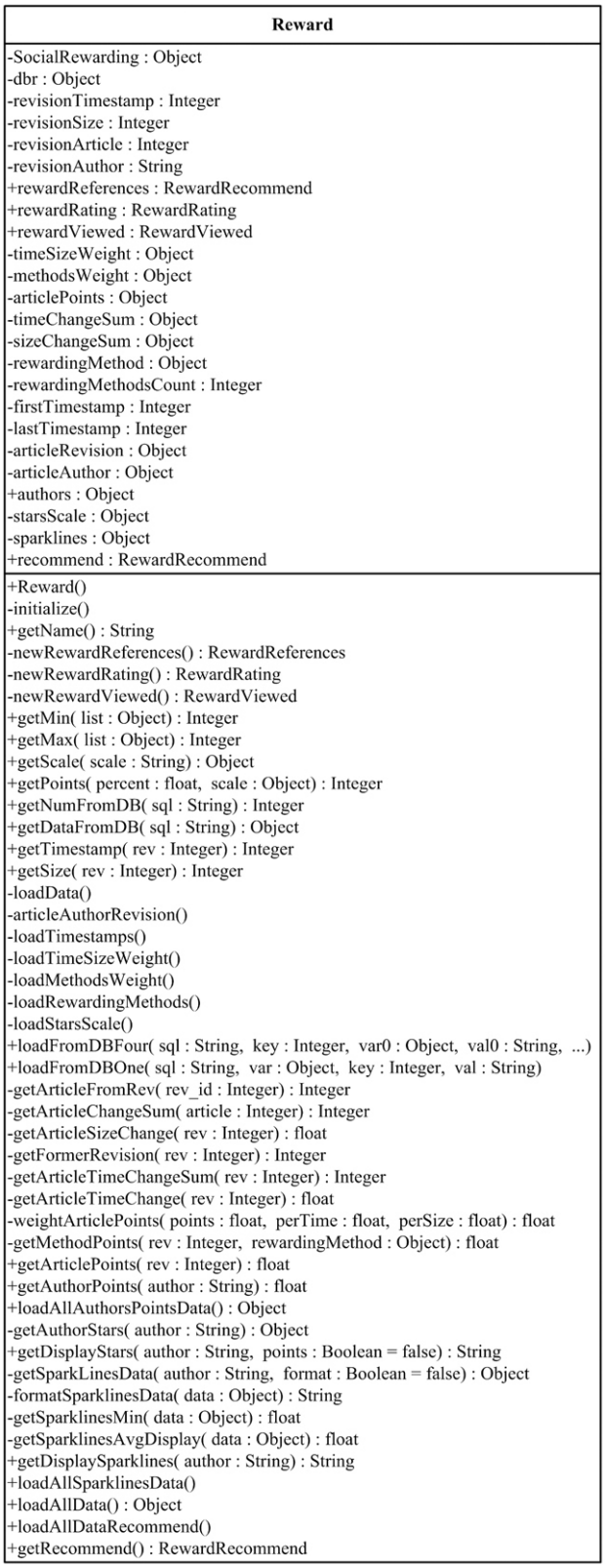

Figure 25: UML class model of object `Reward`.

As the computation process strongly depends on the number of users, articles, and revisions of a wiki there also is a caching algorithm omitting recalculations upon every single request[78]. According to the configuration file there are two different caching methods: either writing the cache file onto the filesystem or in the database. As every data needed for the calculation is stored in runtime variables and objects, PHP can generate a storable representation of these values which can be saved later on. If caching is done using the filesystem, the cache file is overwritten when it becomes too old. The cache refresh time must be specified in the configuration file. If the caching method makes use of the database, every cache file is stored along with a timestamp (Figure 26). Therefore, it is possible to generate a history of rankings of authors which provides for tracking the most active users over time.

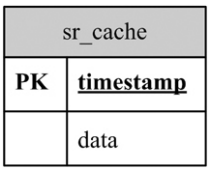

Figure 26: ER model of table `sr_cache`.

The class for handling the cache algorithm is displayed in Figure 27. It depends on the configuration which methods are called. Different functions exist for caching based on either the filesystem or the database (such as two read and write methods each).

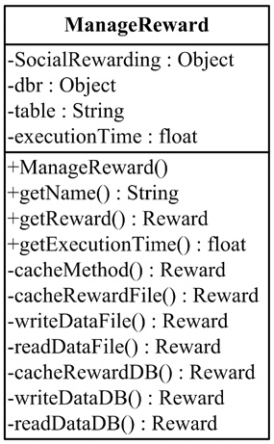

Figure 27: UML class model of object `ManageReward`.

The whole calculation process can be very time consuming. If the computation lasts too long, one may have to increase the `max_execution_time` parameter in the

[78] Caching must be enabled and set up in the configuration file.

`php.ini`[79]. Another important fact is that if the database is used as the cache storing location, one probably has to increase the `max_allowed_packet` parameter of the MySQL database server, as the single SQL statement sent to the server to store the cache can be too big. Moreover, if the file system is used for caching data, the executing program must have write access to the according directory.

5.6.2 Calculation Process

The developed social rewarding techniques focuses on accentuation of the most active members in a community, which is achieved by highlighting the most productive authors in a ranking. In the former chapters the social rewarding methods has been introduced which are used for the calculation of such a listing. Now it is time to explain the two-step calculation process.

Each of the three social rewarding mechanisms computes points for a single revision of every article. This is done by comparing the value of the specific revision with the average value of all revisions in the wiki (Equation 4).

$$avg_{R_j} = \frac{\sum_{i=1}^{n} r_{ij}}{n} \tag{4}$$

R is a set of all revisions where r_{ij} is the value of the social rewarding mechanism j of revision i:

$$R_i = \{r_{1j}, r_{2j}, r_{3j}, \ldots, r_{ij}; 1 \leq j \leq 3\} \tag{5}$$

For example, for the social rewarding technique Most Viewed Articles all visits to a revision are counted. Assume article A in revision 7 has 20 views. The average value of views of all revisions is 30. So revision 7 of article A has only 66.67% of the overall average views. As every revision should be credited with a certain amount of points according to its visits, a scale has to be predefined for the mapping. In this scale a value of 66.67% would be graded with 2 out of 5 points[80].

This example of point assignment is done for every revision and for every social rewarding method. For the technique Rating of Articles users vote for an article by assigning 0 to 5 points. For the technique Amount of References the number of links pointing to a reference, the size of this reference, and the number of links pointing to

[79] By installing the SocialRewarding extension and perform a first data initialization the script tries to set a never-ending maximum execution time by putting `set_time_limit(0)`. If one is running PHP in safe mode, this has no effect and one has to edit the `php.ini` and set the `max_execution_time` parameter by oneself.

[80] These examples use a scale from 0 (worst) to 5 (best) points, but it can be defined as wanted.

the specific article are used as parameters. These three criteria are weighted according to users' settings and are compared to a mean value calculated over all revisions.

At the end of the first computation step for every social rewarding method and for every revision points are assigned according to predefined scales. These values are weighted and summed up to an overall value per revision. By looking at Equation 6 it can be seen that p_{r_i} represents the summed up points for revision i for every social rewarding method j weighted against w_j (which has to be defined in the configuration file).

$$p_{r_i} = \sum_{j=1}^{3} \frac{r_{ij}}{avg_{R_j}} \times w_j \tag{6}$$

In the next step the allocation of revision points to authors is done. As a revision is related to exactly one author, it is possible to sum up all points of every revision an author has written. This is done by using two methods to weight the result: the length of the edit process and the creation time of a revision. Therefore, a modified set of R is created where r'_{ik} is revision k of article i (Equation 7).

$$R'_i = \{r'_{i1}, r'_{i2}, r_{i3}, \ldots, r'_{ik}\} \tag{7}$$

It is assumed that the more different a new revision is compared to the former one, the more important the changes must have been. It does not matter, whether a new revision is extended or shortened – a surplus in content quality is assumed[81]. The changes from one revision to another are measured in bytes. Applying Equation 8 an overall value of size changes from all revisions k from an article i is got (where $s_{r'_{ik}}$ is the specific size change from one revision to the former one).

$$S_{r'_i} = \sum_{k=1}^{n} \underbrace{\left| size(r'_{ik}) - size(r'_{i(k-1)}) \right|}_{s_{r'_{ik}}} \tag{8}$$

The second assumption is that newer revisions count more than older ones. It is believed that newer revisions have up-to-date topics and therefore should be weighted higher than ones written long ago. Equation 9 sums up the relative amount of time for all revisions k for an article i ($t_{r'_{ik}}$ is the relative amount of time for one revision).

$$T_{r'_i} = \sum_{k=1}^{n} \underbrace{time(r'_{ik}) - firstTime(r'_i)}_{t_{r'_{ik}}} \tag{9}$$

For all revisions of an article, the changes to the former revision, and its age with regard to the creation date of the article are saved (Equation 8 and Equation 9).

[81] That means flamers, trolls etc. are factored out.

Equation 10 defines a subset A of revisions belonging to one author. This means that only revisions from a specific author for whom the calculation is performed are considered. For example, $s_{a_{ik}}$ (in Equation 11) is the size change from one revision of an author which is divided by the overall size change of all revisions of the article to obtain a percentage value.

$$A \subseteq R' \tag{10}$$

In Equation 11 for every revision belonging to an author and every criterion (size and time) percentage values are generated which are weighted using a predefined scale (w_S and w_T). Then these two values are multiplied with the specific points calculated in the first step for this revision (p_{r_k}) and both values are summed up. The result is a newly weighted value for every revision ($p_{a_{ik}}$) which has to be summed up for all articles belonging to an author (p_A).

$$p_A = \sum_{i=1}^{n} \sum_{k=1}^{m} \underbrace{\left(\frac{s_{a_{ik}}}{S_{r'_j}} \times w_S + \frac{t_{a_{ik}}}{T_{r'_j}} \times w_T \right) \times p_{r_k}}_{p_{a_{ik}}} \tag{11}$$

This procedure has to be repeated for all authors, so that finally every author has one single value assigned which is the basis for displaying the ranking. Figure 28 gives an overview of the two-step calculation process described in this chapter. As already mentioned, points are computed on a revision basis using the three developed social rewarding methods. In a second step points are weighted according to time and size factors and summed up for one author.

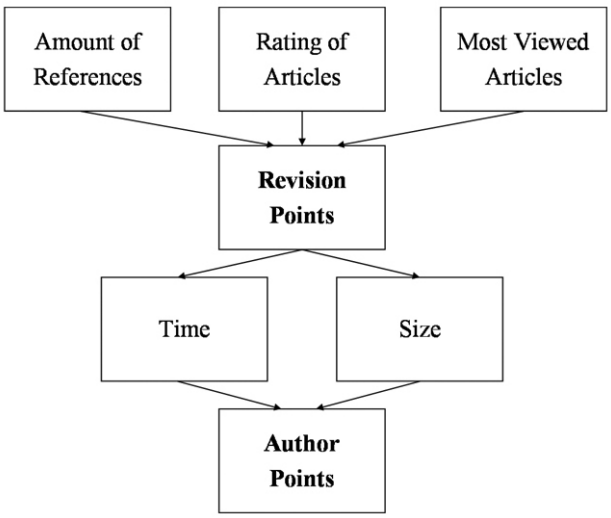

Figure 28: Two-step calculation process of ranking of authors.

5.6.3 Visualization

For the representation of results, various authors' rankings can be generated with the most active ones at the top (Figure 29). The output is displayed using a special page offering a user the possibility to adjust some calculations and display settings. To highlight displayed results, two well-known data visualization techniques are used: stars and sparklines[82] (for more information it is referred to Chapter 4 on page 76). The numbers on the right represent the achieved scores according to the calculation of the social rewarding methods. The ranking of authors is the main output of the SocialRewarding package. The top of the list shows the most productive authors according to the former described computation process.

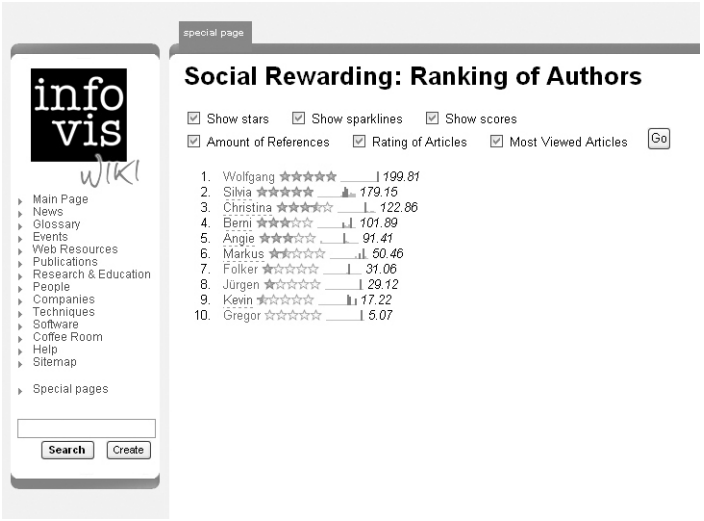

Figure 29: Screenshot of ranking of authors.

Amongst others, the following information can be retrieved from Figure 29:

- User *Christina* is the third most productive author according to the calculation.
- The productivity of *Christina* is about 60% of the most active author (*Wolfgang*)[83].
- *Silvia* has a more constant participation rate than *Wolfgang* who has contributed only in the last time period (but then very actively).

[82] For displaying sparklines the *Sparkline PHP Graphing Library* in version 0.2 from 2 June 2005 is used. Copyright of this software is held by James Byers. It is referred to the special license file located in the directory `[SocialRewarding]/sparkline-php-0.2` of the SocialRewarding package. Further information can be found on `http://sparkline.org`.

[83] This is calculated by comparing the score points.

- As *Markus* has only one and a half stars his overall contribution rate is not very high although it increases steadily (but with pauses).

If data caching is activated and a database is used as data storage system, another special page can output the history of ranking of authors (Figure 30). This means a ranking of authors at a specific point in time (when cache data was saved).

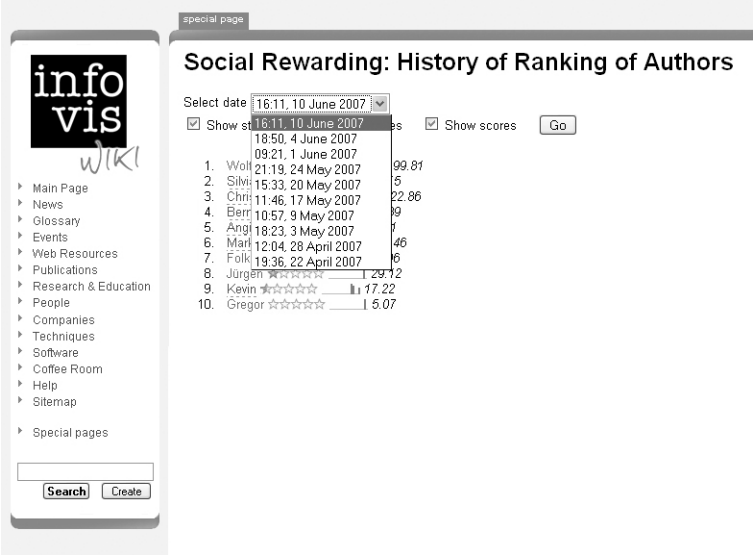

Figure 30: Screenshot of history of ranking of authors.

6 Conclusion and Future Work

As under-contribution is a serious problem for many online communities, in this paper it was tried to give an insight on how to motivate users by means of social rewarding techniques. At the beginning, an insight in online communities and user motivation in general is given, then different social rewarding methods are analyzed, and in a further step wikis and their implementation process is described. The implementation is based on the accentuation of most active members in a wiki and in the case of the Recommender System on displaying recommended authors. To find the most productive members a ranking of authors is generated by making use of calculated points of three developed social rewarding mechanisms: Amount of References, Rating of Articles, and Most Viewed Articles. Several parameters influence the ranking, of which some are configurable, while others rely on the quantity, quality, and novelty of authors' contents. Apart from the ranking, stars and sparklines are used to visualize the results. As an implementation platform MediaWiki has been chosen in which the social rewarding mechanisms are integrated.

These concepts are based upon the fact that accentuation of most active members can increase the community's contribution rate. By investigating several different social rewarding techniques it was tried to focus on the problem from different angles in order to get diversified results. It was thought that developed mechanisms are adequate to face the problem of under-contribution. The implementation in the MediaWiki software has been technically challenging and thus has been very time consuming. Actually, MediaWiki can easily be extended, the bigger problem has been to find an appropriate calculation algorithm. As the amount of data which needs to be processed can be huge, a time and memory efficient calculation algorithm is essential. It has been fundamental to implement a caching method to increase the performance of the computations' final results. However, this means that the bottleneck of this implementation is only the final output and not the data retrieving or storing process. Therefore, a normal user who contributes to a wiki has no disadvantages if caching is disabled and the calculation of authors' ranking takes more time than expected. In such an environment a delay can be recognized only when viewing the final output, for example, the list of ranking of authors. This is the result of separating data collection and result computation (which was a vital aspect of the implementation).

As the developed social rewarding techniques are completely based on data retrieved from the community, they may take some time and the active cooperation of users to deliver representative results, which depends strongly on the size of the involved community and their participation rate. This circumstance and the fact that many variables can be set in the configuration file can lead to a shorter adaptation rate of these settings for an administrator. Installing the SocialRewarding extension with an initial set of data can significantly improve first results. The longer the social rewarding methods are in use the better the results and the fewer adaptations

have to be performed by an administrator.

Improvements of implemented social rewarding methods can be done in two ways: by extending existing or by developing new ones. For example, the method Amount of References can be extended by making use of another search engine. When combining results from two or more search engines the output can be enhanced. As a result, data retrieval does not depend on one data source only. That means that data collection works if one search engine is not accessible. It may also be useful if the method Rating of Articles would allow more than one question to be asked. As voting for a revision is assumed to reflect the community's opinion about an article, another improvement could include ignoring points of a rating until a predefined threshold of actively voting members is reached. Moreover, the technique Most Viewed Articles can rely on other or further criteria to rank an article. For example, one could store how often a user has visited an article in addition to who has visited it. Then it would be possible to weight visits from more productive authors more than other visits. The outcome would be a weighted list of articles which have been visited. Interesting results could also be delivered by implementing a social rewarding mechanism based on extra long articles found by often used search terms. To get a more dynamic ranking of authors, their productivity can be pooled in weekly or monthly lists which then certainly have a high fluctuation. Another attractive idea would be to integrate a career system into MediaWiki by extending the existing (and insufficient) group permissions.

Furthermore, a ranking of most productive authors could result in users having acquired such a huge amount of points that other "normal" users cannot reach a better ranking as one or two stars. Therefore, within future work, it could be applicable to factor out outliers by making use of 1.5 times the interquartile range or something similar to that. This does not mean that these users are completely eliminated from the ranking, but rather that their points are not considered in the computation process.

As it is described in this paper, every revision of an article gets credit points assigned according to the implemented social rewarding methods. These points are then weighted against size and time factors and transferred to an author's account. In the computation process the difference in size from one revision to the former one is measured. It is assumed that deletion of text generates a surplus, too. This method may be discussed controversially, but despite the fact that it is not wanted that humans are involved in the calculation process, it is the best solution. It would be a huge effort to manually check for the quality of every change applied to an article. As a future work some artificial intelligence algorithms could be implemented to automatically check edits. If such an algorithm is not able to evaluate the quality of an article, but to sort out spam, troll, or flamer edits from other "good" entries, this would certainly be useful, as well.

The result of a ranking of authors is a list which is supported visually by stars and sparklines. These two methods are a good way to show the productivity of an author, but other information visualization techniques can be inserted, too. A further option could be to display stars and/or sparklines beside every occurrence of an author's name – not only on special pages. Such an extension would integrate social rewarding methods in the overall picture of a wiki.

This approach can be seen as a starting point for the development of mechanisms that focus on motivating users to participate actively in a wiki system. In no other online community the participation rate of users is more important than in a wiki, as there producers and consumers of the good (information) are the same. If too few users produce content and do only free ride, a wiki community cannibalizes itself. The implementation of social rewarding techniques presented herein creates qualitatively high results, due to the mixture of several awarding methods, which are necessary to generate non-monetary incentives for users. Nevertheless, the mechanisms described in this thesis might not be sufficient to motivate enough people to form an active community that participates in *every* wiki. Besides all, users must have an intrinsic motivation to contribute, since the developed techniques stimulate only the extrinsic factors such as earning praise, glory, and status, which are outside rewards.

The project has not been publicly released yet and therefore empirical data is not available. Consequently, it is currently planned to evaluate the implemented concepts in a larger setting.

Appendix

A Overview of Social Rewarding Techniques

Name	Category	Potential	Problems	Implementation[84]
Top contributors	Top 10 lists	Long term motivation; ranking method; battle against other users; every user can see top ranked users; good for linking users to specific online community	Long way for newbies to get on top of ranking; not good for quick motivation; frustrating if user does not climb the ladder	Easy/Medium: adaption of already existing user list (special page) necessary to display results; regarding which method to implement data already exists or has to be collected separately
Last contributed articles	Top 10 lists	Short term motivation; ranking method; put new articles into the spotlight; motivation to contribute many articles	Motivation lasts shortly because of the quick disappearance of own article in ranking; stubs are produced more frequently to get more articles in the ranking	Easy: data exists; adaption of existing special page necessary

[84] Factor indicating the complexity for implementation in the MediaWiki system.

Name	Category	Potential	Problems	Implementation
Most viewed articles	Top 10 lists	Ranking method; long term motivation; rankings for different categories and/or time separation; good indication for well written articles; calculation of reputation of authors possible	Counting method: hits or visits; counting per article or per revision (to link to the author of a revision); author can influence ranking by visiting her/his own article very often (has to be restricted); articles with an up-to-date topic are visited more often (Really an indicating factor for the quality of an article?)	Easy/Medium: if counting is based on articles, data exists, else data has to be collected (revision based); adaption of existing special page necessary
Extra long articles	Top 10 lists	Ranking method; motivation to write long articles or to extend existing ones; good for linking users to specific online community; multimedia content can be considered	Hard to gather from length of article to its quality; hard to interpret quality of multimedia content	Easy/Medium: data exists for counting only the length of an article (without multimedia content); if multimedia content should be considered, it has to be interpreted in some way; adaption of existing special page necessary

Name	**Category**	**Potential**	**Problems**	**Implementation**
Often used search terms	Top 10 lists	Ranking method; if users are visiting one article very often although searching for different terms, this can be an indication for the quality of the article; long term motivation method	Author can influence ranking by submitting many search queries finding her/his own article (has to be restricted); implies that user use search engine regularly	Medium: data has to be collected; adaption of special page necessary
Amount of references	Top 10 lists	Ranking method; initial check of quality of articles can be realized; method cannot be influenced by the author; amount of references are a good alternative approach to find out the quality of an article (in combination with other methods)	Book references are hard to handle (More valuable than Internet resources?); need third party software (e.g., Google service must be available); checking many references on a server with low bandwidth can be time consuming; reference can be declared in article but has nothing to do with it (checking if a reference fits to the topic of an article is very difficult)	Medium/Hard: data has to be collected using third party software (e.g., Google's SOAP search API), adaption of special page necessary

Name	Category	Potential	Problems	Implementation
Most linked user-pages	Top 10 lists	Long term motivation; amount of links are a good approach to find out the name recognition of an article; personal information and statistics about the user are collected on one page	Not every user creates a user-page although being a very active contributor; users may do not want to reveal statistics about their contributions	Medium: mechanism "what links here" exists and have to be adapted; data for statistics partially exists (but further information has to be collected); listings for displaying results has to be adapted
Productivity of authors	Top 10 lists	Short term motivation; good indication for present productivity of authors; many different techniques; time-based and categorized; the chance for an author to be displayed on a top position in a ranking is higher; users are linked closer to the online community	User has to contribute a lot of articles in a short period of time (e.g., per month) to be ranked; can be very time consuming; contributions made in the past are ignored completely; may lead to a behavior where users only have a look at the quantity of articles but not on the quality of them	Easy/Medium: depending on which method to implement, data has to be filtered (or even collected) on a time or category basis; adaption of special page necessary
Rating of articles	Rating systems	Present other users' opinions best; combination of quantitative and qualitative methods possible; author gets good feedback about her/his article	Number of minimal votes needed to get a representative result; asking open questions is time consuming for a user and cannot be measured automatically	Medium: ranking form has to be generated; data has to be collected; the possibility for manipulation has to be eliminated; a special page has to be designed to display results

Name	Category	Potential	Problems	Implementation
Estimated effort	Rating systems	Tries to represent the author's estimated effort best; both authors' and users' points of view	Estimated effort is hard to determine; real costs may not be considered because of the very subjective kind of the concern; author may tend to understate or overstate her/his effort; high estimated effort does not implicate a good written article	Medium: rating system must be implemented (authors and users); a special page displaying results has to be designed
Credit points	Rating systems	Long term motivation; users try to participate actively to climb the latter; strong linking of users to the online community; good representation of users' participation; easy comparison between different users; motivating users by giving more rights to top ranked users (or users at a certain level)	Long way to the top; early registered users have a healthy margin; if a user is inactive and credit points get lost, achieving a high amount of points can be very time consuming; users participate only in services rewarded with credit points	Hard: complete credit points system must be implemented; collecting data (depending on which action should be rewarded with credit points); designing special pages to display results

Name	Category	Potential	Problems	Implementation
Survey systems	Survey systems	Good way to present opinions of users; receives many votes because of positioning on a prominent place; variety of polls are possible (also through randomization); time and/or category based; simplicity through preselection; combination with other methods possible	Polls can be displaced, e.g., on the front page of the online community; human action is needed if no randomization is implemented; too much voting possibilities can annoy users; representative results are questionable because of preselection	Medium: poll form has to be generated; inserting new questions automatically (randomization) and saving prior results must be implemented; special pages to display results over time and category based must be designed
Recommender systems	Recommender systems	Good reflection of users' attitudes; many sources to collect data; calculate recommendations based on data of the whole community; bringing together users with same interests	Can produce wrong recommendations (especially with few data); users' preferences may have changed over time; protection of data privacy must be observed	Hard: depending on which methods to implement data has to be collected; creating the implementation of calculation routines for a representative recommender system; designing special page to display output

Name	Category	Potential	Problems	Implementation
Reminder systems	Reminder systems	Reminds users of online community and/or new features; should link users to online community; can be push-based (users do not have to do the first step) or pull-based (users have to act to get information); link to real-life events possible	Sending emails as a reminder can be annoying to the user; option to turn feature of must be implemented; reminder has to contain well suited information for the user to be effective; only makes sense on low participating users	Medium: data has to be collected, interpreted, and formatted; information has to be sent by using email/SMS etc.; information of return-rate of users getting a reminder message must be counted
News systems	News systems	Up-to-date information is displayed on a central place; can save user's time; general and user-specific news possible; information can substitute other sources where a user has been looking for news so to link the user to the online community	Moderated news articles need human resources; news must really be up-to-date; categorized news only makes sense on big online communities where information is nearly unmanageable	Easy/Medium: depending on which systems are implemented; newest data has to be collected and displayed; user-specific news options must be available; designing special page to display output; link to reminder system possible (e.g., sending news from an article by email)

Name	Category	Potential	Problems	Implementation
Personal feedback systems	Personal feedback systems	Qualitative high personal feedback; direct contact with author; good for discussing complex problems; personal feedback can be combined with other methods (e.g., rating systems); differentiates feedback only viewable for the author and for the public	Giving personal feedback and answering on it is time consuming; many authors may not want to be contacted directly; there is no good way to analyze personal feedback automatically	Easy: discussion page exists already which can be adapted for personal feedback; contact information of authors must be displayed
Hyper-linking systems	Hyper-linking systems	Linking together different users with same interests; can be category based; does not care about participation rates; therefore, put users in the spotlight who are not top users in a ranking; good to get in contact with new people; good for creating a strong community; intercultural exchange possible	Community must have exceeded critical mass of users to make a hyper-linking system useful; data privacy must be protected; “random user-page” only makes sense if user-page is not empty; every user has the same chance to be put in the spotlight, so motivation to participate actively could not be as high as expected	Medium: data has to be collected; implementing a routine for a representative hyper-linking system; existing random page function can be used; special page has to be designed to display results

Name	Category	Potential	Problems	Implementation
Career systems	Career systems	Long term motivation: get on the highest level; short term motivation: reach next level; good for participating in an online community over a long time; good representation of users' participation; good for link users to specific online community; motivating users by granting more rights to users at a certain level; can be linked to a ranking system; by taking wrong actions a user could also fall down a level (punishment possible)	New users may get frustrated if it takes too long to climb the ladder; early registered users have a healthy margin; if punishment is implemented, a user may fall down a level if she/he does not participate over a certain period of time (What to do if user is on holiday?); very strict hierarchic system; user participates only in services which help reaching a higher level	Hard: complete credit points and career system must be implemented; collecting data (depending on which action should be rewarded with credit points); designing special pages to display results

Name	Category	Potential	Problems	Implementation
Award systems	Award systems	Emphasizes outstanding performances (which must not necessarily be long term performances); can be awarded through the community or through users with special privileges (e.g., administrators); hall-of-fame possible; motivates users because of their uniqueness	When assigned through the community polls have to take place (time consuming); if assigned through one person, it is not easy to determine who has the rights to assign an award and how to handle false decisions; awards must be displayed graphically and have to be designed	Easy/Medium: polls must take place if the community decides who will be awarded; data has to be collected (escapes if only one person has to decide); awards must be designed graphically; special page has to be developed to display award winners
User-centric search	User-centric search	Focuses search on authors' names, nicknames, email addresses etc.; puts authors in the spotlight; connecting authors from same articles	If caching is enabled, search engine does not index all pages at real-time; problem if there are hundreds or even thousands of authors of an article (dropdown list impossible)	Easy: extension of already existing search engine necessary

Name	Category	Potential	Problems	Implementation
Documentation and help systems	Documentation and help systems	Good documentation releases help system (and therefore save human resources); structured documentation saves users' time; a user gets more motivated if everything is explained to her/him in a way she/he understands; if credit points are assigned, users get motivated to help others; motivation through assigning a user to the "help team"	No one likes writing documentation and FAQs; human resources must be spent on writing and updating the documentation and on the implementation of an efficient help system	Easy/Medium/Hard: it depends on which help system is implemented (email, discussion list, or categorized forum) and if the help system is developed in conjunction with a credit point system
Motivation through punishment	Motivation through punishment	Unwanted behavior can be stopped; disturbing users can be blocked from the online community (over a period of time or forever); wrong information can be prevented from being published	Punishment alone does not motivate users; only useful in combination with positive social rewarding methods; too strict consequences discourage users; public discredit is not wanted	Easy/Medium/Hard: it depends on which punishment systems are going to be implemented and if these systems already exists and if adaption is necessary

Table 8: Overview of social rewarding mechanisms.

B File SocialRewardingREADME

```
1  == SocialRewarding extension for MediaWiki ==
2
3  Version: 0.9
4  Date: 2007-07-21
5
6  Copyright (C) 2007 Bernhard Hoisl <berni@hoisl.com>
7
8  This package is the SocialRewarding extension for MediaWiki. The
9  focus is laid on generating benefits for users in order to
10 achieve a higher contribution rate in a wiki system. Therefore,
11 social rewarding mechanisms are implemented to reward people
12 contributing to an online community. In an online community
13 social rewarding is in the majority of cases based on
14 accentuation of the most active members. For example, ranking
15 of users' activities ("top contributors"). In this extension
16 four techniques has been designed: "Amount of References", "Most
17 Viewed Articles", "Rating of Articles" and a "Recommender
18 System". An in-depth explanation of social rewarding methods,
19 there design principles, and a technical description can be
20 found in the document named "SocialRewardingDocumentation.pdf"
21 which is included in the package and should be located in the
22 same directory as the extension.
23
24 The SocialRewarding extension is licensed under the terms of the
25 GNU General Public License, version 2 or later (see
26 http://www.fsf.org/licenses/gpl.html). You should have received
27 a copy of the GNU General Public License along with this program
28 (have a look at file "SocialRewardingCOPYING").
29
30
31 == Installation requirements ==
32
33 * A web-server (extension was tested on Apache 2.0.54).
34 * MediaWiki 1.6 or higher (extension was tested with 1.6.0,
35   1.6.7, 1.7.1 and 1.10.0; with some little adaptations the
36   extension should work on older versions, too).
37 * PHP 4.3 or higher; 5.x is recommended (extension was tested
38   with PHP 4.3.5 and 5.1.4). If you use a PHP version prior 5
39   social rewarding method "Amount of References" is disabled
40   automatically because of a nonexistent SOAP interface.
41 * A MySQL server 4.0 or higher (extension was tested with MySQL
42   4.0.18).
43
44 The SocialRewarding extension was developed on Apache 2.0.54 with
45 MediaWiki 1.6.7 and 1.10.0 respectively, PHP 5.1.4, and
46 MySQL 4.0.18.
47
48 The extension may work with older software products, but then
49 it can be that not all features are available. However, to
50 minimize problems it is recommended that you always use the
51 latest stable release of all software products.
52
53 For security reasons please turn register_globals = Off in your
54 php.ini. Neither MediaWiki nor the SocialRewarding extension
55 need register_globals to be enabled.
56
57
58 == Installation ==
59
60 * Download a copy of the zipped SocialRewarding package.
61 * Unpack all files to the extension directory of your MediaWiki
62   installation (e.g. [MediaWikiPath]/extensions/SocialRewarding/).
63 * To activate the SocialRewarding extension in MediaWiki you have
```

```
64    to edit your [MediaWikiPath]/LocalSettings.php. Insert the
65    following at the end of the file:
66    require_once("extensions/SocialRewarding/SocialRewarding.php");
67  * Configure the SocialRewarding package by editing the file
68    SocialRewardingConfigDetail.php or SocialRewardingConfigShort.php
69    depending wether you would like to have a detailed or only a
70    short description of the configuration parameters. To enable one
71    of the configuration files you have to edit the file
72    SocialRewarding.php and un-/comment lines 43 or 44 where the
73    short and the long version of the configuration files are loaded.
74    The short version is loaded by default. Be sure to have your
75    copy of the SocialRewarding extension configured correctly before
76    moving on to the next step.
77  * For setting up social rewarding method Amount of References you
78    have to obtain a license key from http://www.google.com/apis to
79    be able to use Google's SOAP Search API (beta). The license key
80    has to be inserted into the variable
81    $SocialRewarding["references"]["googleKey"] in your configuration
82    file.
83  * Now start your favorite web-browser and navigate to the
84    installation script. The URL should be something like
85    http://www.YourWebServer.org/MediaWiki/index.php/
86    Special:SocialRewardingInstall.
87    Alternatively, you can also reach the installation page if you
88    browse the special pages of your MediaWiki and click on the link
89    called "Social Rewarding: Installation".
90  * If you get an error while trying to reach the installation
91    special page you probably have to copy all special pages of this
92    package to the [MediaWikiPath]/includes/ directory. To identify
93    special pages they are all starting with
94    "SpecialSocialRewarding".
95  * Follow the instructions on the installation page. After you
96    completed the installation successfully your SocialRewarding
97    extension should work.
98
```

File 1: SocialRewardingREADME

C File SocialRewardingConfigDetail.php

```
1  <?php
2
3  # Copyright (C) 2007 Bernhard Hoisl <berni@hoisl.com>
4  #
5  # This program is free software; you can redistribute it and/or modify
6  # it under the terms of the GNU General Public License as published by
7  # the Free Software Foundation; either version 2 of the License, or
8  # (at your option) any later version.
9  #
10 # This program is distributed in the hope that it will be useful,
11 # but WITHOUT ANY WARRANTY; without even the implied warranty of
12 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
13 # GNU General Public License for more details.
14 #
15 # You should have received a copy of the GNU General Public License along
16 # with this program; if not, write to the Free Software Foundation, Inc.,
17 # 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.
18 # http://www.gnu.org/copyleft/gpl.html
19
20
21 /**
22  * @package MediaWiki
23  * @subpackage extensions
24  * @subsubpackage SocialRewarding
25  */
26
27
28
29 /**
30  * Detailed config file of the SocialRewarding package. If you feel the
31  * need to configure something, do it here.
32  */
33
34
35
36 /*****************************************************
37  * GLOBAL SOCIAL REWARDING CONFIGURATION
38  *****************************************************/
39
40 /**
41  * Activate SocialRewarding extension?
42  */
43
44 $SocialRewarding["reward"]["active"]                    = true;
45
46
47 /**
48  * The path to the SocialRewarding extension directory (where you
49  * copied all files of the SocialRewarding extension to).
50  */
51
52 $SocialRewarding["reward"]["extensionPath"]             = "extensions/";
53 $SocialRewarding["reward"]["extensionPath"].            = "SocialRewarding";
54
55
56 /**
57  * Activate caching?
58  */
59
60 $SocialRewarding["reward"]["cache"]                     = false;
61
62
63 /**
```

```
64   * Caching for social rewarding mechanism "Recommender Systems" can be
65   * activated separately. $SocialRewarding["reward"]["cache"] must be
66   * set true for this to take effect.
67   */
68
69  $SocialRewarding["reward"]["cacheRecommend"]            = true;
70
71
72  /**
73   * Caching method: "db" or "file". By selecting "db" a history of
74   * calculated rankings of authors is kept. Attention: Data
75   * to store in the database can exceed "max_allowed_packet"
76   * size of MySQL. Then the limitation has to be extended.
77   */
78
79  $SocialRewarding["reward"]["cacheMethod"]               = "db";
80
81
82  /**
83   * Cache refresh in seconds, e.g. 1 hour = 3600, 1 day = 86400.
84   */
85
86  $SocialRewarding["reward"]["cacheTime"]                 = 600;
87
88
89  /**
90   * Name of file for caching (only necessary if caching is based on file
91   * system (that means $SocialRewarding["reward"]["cacheMethod"] = "file").
92   */
93
94  $SocialRewarding["reward"]["cacheFile"]                 = "SocialRewardingCache";
95
96
97  /**
98   * Computing the size of a revision using methods "LENGTH" or
99   * "CHAR_LENGTH" of MySQL. For an explanation consult the MySQL
100  * documentation.
101  */
102
103 $SocialRewarding["reward"]["sizeMethod"]                = "LENGTH";
104
105
106 /**
107  * All floats are rounded to x decimal places.
108  */
109
110 $SocialRewarding["reward"]["round"]                     = 2;
111
112
113 /**
114  * Delimiter for string manipulation. Every string which should be
115  * tokenized has to be formatted using the character beneath as
116  * delimiter.
117  */
118
119 $SocialRewarding["reward"]["delimiter"]                 = ",";
120
121
122 /**
123  * Timestamp from which to begin calculating all social rewarding
124  * mechanism ("" = first timestamp in database).
125  */
126
127 $SocialRewarding["reward"]["beginTimeInterval"]         = "";
128
129
```

```
130 /**
131  * Ending timestamp for calculation ("" = last timestamp in
132  * database).
133  */
134
135 $SocialRewarding["reward"]["endTimeInterval"]           = "";
136
137
138 /**
139  * Percentage of weighting of time and size calculation methods
140  * (in this order).
141  */
142
143 $SocialRewarding["reward"]["timeSizeWeight"]            = "50,50";
144
145
146 /**
147  * Calculation method for computing the size of a revision. "mean" is
148  * faster and calculates the size of a revision by using an averaged
149  * value. "repr" is a little slower but highly recommended because
150  * the size changes from one revision to another is calculated using
151  * exact values.
152  */
153
154 $SocialRewarding["reward"]["sizeCalcMethod"]            = "repr";
155
156
157 /**
158  * Calculate article points with decimal places? Only for calculation,
159  * displayed values are rounded using
160  * $SocialRewarding["reward"]["round"].
161  */
162
163 $SocialRewarding["reward"]["articlePointsDecimalPlace"] = true;
164
165
166 /**
167  * Weighting for the three social rewarding mechanisms in percent.
168  * Order: "Amount of References", "Rating of Articles" and "Most
169  * Viewed Articles".
170  */
171
172 $SocialRewarding["reward"]["methodsWeight"]             = "33,33,33";
173
174
175 /**
176  * Basis for calculating social rewarding mechanisms: "all",
177  * "articles", or "user_pages".
178  */
179
180 $SocialRewarding["reward"]["calcBasis"]                 = "all";
181
182
183 /**
184  * Corrects calculation basis, that means excluding system pages
185  * (=pages with namespace like
186  * $SocialRewarding["reward"]["calcBasisCorrectionNS"]).
187  */
188
189 $SocialRewarding["reward"]["calcBasisCorrection"]       = true;
190
191
192 /**
193  * Namespace for system pages.
194  */
195
```

```
196 $SocialRewarding["reward"]["calcBasisCorrectionNS"]      = 8;
197
198
199 /**
200  * Namespace for articles (main pages).
201  */
202
203 $SocialRewarding["reward"]["calcBasisArticlesNS"]        = 0;
204
205
206 /**
207  * Namespace for user-pages.
208  */
209
210 $SocialRewarding["reward"]["calcBasisUserPagesNS"]       = 2;
211
212
213 /**
214  * Calculation method for authors' stars. "min_max": obtain a minimum
215  * and a maximum value from all authors. By taking points from one
216  * author with this range a percentage value can be generated. "mean":
217  * getting an overall average value from all authors. By relating
218  * points from one author to this average value percentage data can
219  * be generated.
220  */
221
222 $SocialRewarding["reward"]["calcMethodStars"]            = "min_max";
223
224
225 /**
226  * Calculate social rewarding mechanisms only on registered users
227  * (without "MediaWiki default" user)?
228  */
229
230 $SocialRewarding["reward"]["calcUsersOnly"]              = false;
231
232
233 /**
234  * Scale for calculating stars for authors in percent, e.g. from 0
235  * to 15 percent 0 stars are displayed, from 15 to 35 percent 1 star
236  * and so on. Scaling must be configured regarding which calculation
237  * method is set in $SocialRewarding["reward"]["calcMethodStars"] and
238  * the structure of the wiki community.
239  */
240
241 $SocialRewarding["reward"]["starsScale"]                 = "0,15,35,55,75,90";
242
243
244 /**
245  * Should half stars be displayed? If set to true then
246  * $SocialRewarding["reward"]["starsHalfBorders"] and
247  * $SocialRewarding["reward"]["starsHalf"] must be set.
248  */
249
250 $SocialRewarding["reward"]["starsDisplayHalf"]           = true;
251
252
253 /**
254  * Between which range (in percent) should a half star be displayed?
255  * For instance: "25,75" means that between a value of 2.25 and 2.75
256  * two stars and a half is displayed.
257  */
258
259 $SocialRewarding["reward"]["starsHalfBorders"]           = "25,75";
260
261
```

```
262 /**
263  * Rounding method for stars, only if
264  * $SocialRewarding["reward"]["starsDisplayHalf"] is set false. "floor":
265  * rounding down to the previous integer; "ceil": rounding up to the
266  * next integer; "round": rounding to the nearest integer (e.g. .5:
267  * rounding up).
268  */
269
270 $SocialRewarding["reward"]["starsRound"]                = "round";
271
272
273 /**
274  * Directory pointing to stars images.
275  */
276
277 $SocialRewarding["reward"]["starsDir"]                  = "SocialRewardingImg";
278
279
280 /**
281  * File name of image for a full star.
282  */
283
284 $SocialRewarding["reward"]["starsFull"]                 = "star_full.gif";
285
286
287 /**
288  * File name of image for an empty star.
289  */
290
291 $SocialRewarding["reward"]["starsEmpty"]                = "star_empty.gif";
292
293
294 /**
295  * File name of image for a half star (only needed if
296  * $SocialRewarding["reward"]["starsDisplayHalf"] is set true).
297  */
298
299 $SocialRewarding["reward"]["starsHalf"]                 = "star_half.gif";
300
301
302 /**
303  * Name of directory to sparkline package.
304  */
305
306 $SocialRewarding["reward"]["sparklinesPackageDir"]      = "sparkline-php-0.2";
307
308
309 /**
310  * Displaying interval for sparklines.
311  */
312
313 $SocialRewarding["reward"]["sparklinesInterval"]        = 20;
314
315
316 /**
317  * Minimum percentage value for sparklines that are certainly be
318  * displayed. Set this variable so that a thin line is displayed through
319  * the whole sparkline even if no or to few data is given for an interval.
320  */
321
322 $SocialRewarding["reward"]["sparklinesMinPercent"]      = 10;
323
324
325 /**
326  * Width of sparklines in pixel.
327  */
```

```
328
329 $SocialRewarding["reward"]["sparklinesWidth"]           = 2;
330
331
332 /**
333  * Spacing of sparklines in pixel.
334  */
335
336 $SocialRewarding["reward"]["sparklinesSpacing"]         = 0;
337
338
339 /**
340  * Maximum height of sparklines in pixel. You should try to match
341  * height of stars.
342  */
343
344 $SocialRewarding["reward"]["sparklinesHeight"]          = 12;
345
346
347 /**
348  * Color for under average sparklines.
349  */
350
351 $SocialRewarding["reward"]["sparklinesUnderAvgColor"]   = "C0C0C0";
352
353
354 /**
355  * Color for over average sparklines.
356  */
357
358 $SocialRewarding["reward"]["sparklinesOverAvgColor"]    = "808080";
359
360
361 /**
362  * Background color for sparklines.
363  */
364
365 $SocialRewarding["reward"]["sparklinesBGColor"]         = "FFFFFF";
366
367
368 /**
369  * Set automatic markups active? Auto-markups are markups which are
370  * added automatically on the end of each new submitted article. There
371  * exist auto-markups for every social rewarding mechanism which has to
372  * be set active on its own. This is a global switch to turn all
373  * auto-markups on/off with one move.
374  */
375
376 $SocialRewarding["reward"]["autoMarkup"]                = true;
377
378
379 /**
380  * On which extra namespaces should auto-markups also be set? For
381  * example "100,102" means that on all articles with namespace "100" or
382  * "102" auto-markups are actived. Only even namespaces are
383  * accepted because odd ones are restricted to talk pages.
384  */
385
386 $SocialRewarding["reward"]["autoMarkupExtra"]           = "";
387
388
389
390 /*******************************************************
391  * AMOUNT OF REFERENCES CONFIGURATION
392  *******************************************************/
393
```

```
394 /**
395  * Activate social rewarding mechanism "Amount of References"? This
396  * technique is only available using PHP >= 5 (because of a missing
397  * SOAP interface in PHP versions prior 5). If you are using an older
398  * PHP version $SocialRewarding["references"]["active"] is set
399  * to false automatically.
400  */
401
402 $SocialRewarding["references"]["active"]                = true;
403
404
405 /**
406  * Compute the size of a reference and use this value in the calculation
407  * process?
408  */
409
410 $SocialRewarding["references"]["siteSizeFactor"]        = true;
411
412
413 /**
414  * Count the number of links pointing to references in an article and use
415  * this value in the calculation process?
416  */
417
418 $SocialRewarding["references"]["siteLinkFactor"]        = true;
419
420
421 /**
422  * Count the number of links pointing to an article and use this value in
423  * the calculation process?
424  */
425
426 $SocialRewarding["references"]["siteSelfLinkFactor"]    = false;
427
428
429 /**
430  * Weighting of the three factors in percent. Order: "siteSizeFactor",
431  * "siteLinkFactor" and "siteSelfLinkFactor". If you disable one factor
432  * set weighting to zero, e.g. disabled "siteLinkFactor" (by setting
433  * $SocialRewarding["references"]["siteLinkFactor"] = false): "50,0,50".
434  */
435
436 $SocialRewarding["references"]["siteWeight"]            = "25,25,50";
437
438
439 /**
440  * Scale for computing points of an article in percent, e.g. if an article
441  * has over 150 percent of points relating to the average points calculated
442  * for all articles it gets five points.
443  */
444
445 $SocialRewarding["references"]["articleScale"]          = "0,15,33,66,100,150";
446
447
448 /**
449  * Basis for calculating social rewarding mechanism "Amount of References":
450  * "all", "articles", or "user_pages".
451  */
452
453 $SocialRewarding["references"]["calcBasis"]             = "all";
454
455
456 /**
457  * Corrects calculation basis, that means excluding system pages
458  * (=pages with namespace like
459  * $SocialRewarding["reward"]["calcBasisCorrectionNS"]).
```

```
460   */
461
462   $SocialRewarding["references"]["calcBasisCorrection"]    = true;
463
464
465   /**
466    * Should the search of references (links) be limited to a specified "section"
467    * or the whole "article"?
468    */
469
470   $SocialRewarding["references"]["textMode"]               = "section";
471
472
473   /**
474    * If limitation of searching links is set to "section", a list of
475    * delimiter words can be defined to recognize reference sections (the
476    * section to search for links).
477    */
478
479   $SocialRewarding["references"]["textSection"]            = "References,";
480   $SocialRewarding["references"]["textSection"].           = "Bibliography";
481
482
483   /**
484    * If limitation of searching links is set to "section" here delimiter
485    * symbols can be defined to indicate headings, e.g. that sections starting
486    * with "== References ==" are found.
487    */
488
489   $SocialRewarding["references"]["textDelimiter"]          = "==";
490
491
492   /**
493    * Start text for indicating a link (recommended to do not edit).
494    */
495
496   $SocialRewarding["references"]["linkStart"]              = "http://";
497
498
499   /**
500    * Delimiter characters to know the ending of a link, e.g. spaces, ">", "]"
501    * and so on are indicating that a link ends here (also because these
502    * symbols are not allowed in URLs).
503    */
504
505   $SocialRewarding["references"]["linkDelimiter"]          = " <>[]'\"";
506
507
508   /**
509    * If set true strips beginning "www" from links (better results
510    * for $SocialRewarding["references"]["siteSizeFactor"]).
511    */
512
513   $SocialRewarding["references"]["stripWWW"]               = true;
514
515
516   /**
517    * Should Google search on whole domain instead of subdirectories only? For
518    * instance, link "http://domain.org/MediaWiki/index.php" results in a
519    * Google search in whole "http://domain.org".
520    */
521
522   $SocialRewarding["references"]["googleWholeDomain"]      = true;
523
524
525   /**
```

```
526   * License key for using Google's SOAP Search API (Beta). You have to
527   * register on http://www.google.com/apis to obtain a key for your own.
528   * "Your Google account and license key entitle you to 1.000 automated
529   * queries per day" (according to Google's web-site), but it seems that this
530   * limit is not checked by Google. The search interface was tested with a
531   * lot more than a thousand queries per day (but that does not mean that
532   * the limit is not going to be checked in the near future).
533   */
534
535  $SocialRewarding["references"]["googleKey"]              = "";
536
537
538  /**
539   * "Activates or deactivates automatic results filtering, which hides very
540   * similar results and results that all come from the same web host. Filtering
541   * tends to improve the end user experience on Google, but for your
542   * application you may prefer to turn it off. When enabled, filtering takes
543   * the following actions: Near-Duplicate Content Filter = If multiple search
544   * results contain identical titles and snippets, then only one of the
545   * documents is returned. Host Crowding = If multiple results come from the
546   * same web host, then only the first two are returned"
547   * (http://www.google.com/apis/reference.html).
548   */
549
550  $SocialRewarding["references"]["googleFilter"]           = false;
551
552
553  /**
554   * "Restricts the search to a subset of the Google web index, such as a country
555   * like 'Ukraine' or a topic like 'Linux'"
556   * (http://www.google.com/apis/reference.html). For instance, restrict to
557   * Austria use "countryAT".
558   */
559
560  $SocialRewarding["references"]["googleRestrictCountry"] = "";
561
562
563  /**
564   * "Restricts the search to documents within one or more languages"
565   * (http://www.google.com/apis/reference.html). For example, restrict language
566   * to German use "lang_de".
567   */
568
569  $SocialRewarding["references"]["googleRestrictLang"]     = "";
570
571
572  /**
573   * "A Boolean value which enables filtering of adult content in the search
574   * results. Many Google users prefer not to have adult sites included in their
575   * search results. Google's SafeSearch feature screens for sites that contain
576   * this type of information and eliminates them from search results. While no
577   * filter is 100% accurate, Google's filter uses advanced proprietary technology
578   * that checks keywords and phrases, URLs, and Open Directory categories"
579   * (http://www.google.com/apis/reference.html).
580   */
581
582  $SocialRewarding["references"]["googleSafeSearch"]       = false;
583
584
585  /**
586   * If an error occurs in the Google query, how often should be tried to execute
587   * the query?
588   */
589
590  $SocialRewarding["references"]["googleSearchAttempts"]  = 2;
591
```

```
592
593 /**
594  * Standard markup message for "siteLinkFactor". If a user does not set her/his
595  * own markup message, this one is used. Markups can be inserted using
596  * "<SocialRewardingReferences> Text </SocialRewardingReferences>".
597  */
598
599 $SocialRewarding["references"]["markupLinkStdText"]     = "There is a total of ";
600 $SocialRewarding["references"]["markupLinkStdText"].    = "$1 links pointing to ";
601 $SocialRewarding["references"]["markupLinkStdText"].    = "the references ($2) ";
602 $SocialRewarding["references"]["markupLinkStdText"].    = "in this article.";
603
604
605 /**
606  * Standard markup message for "siteSizeFactor". If a user does not set her/his
607  * own markup message, this one is used.
608  */
609
610 $SocialRewarding["references"]["markupSizeStdText"]     = "The total size of the ";
611 $SocialRewarding["references"]["markupSizeStdText"].    = "sites used as ";
612 $SocialRewarding["references"]["markupSizeStdText"].    = "references in this ";
613 $SocialRewarding["references"]["markupSizeStdText"].    = "article is $3.";
614
615
616 /**
617  * Standard markup message for "siteSelfLinkFactor". If a user does not set
618  * her/his own markup message, this one is used.
619  */
620
621 $SocialRewarding["references"]["markupSelfLinkStdText"] = "$4 links are pointing ";
622 $SocialRewarding["references"]["markupSelfLinkStdText"].= "to this site (until ";
623 $SocialRewarding["references"]["markupSelfLinkStdText"].= "$5).";
624
625
626 /**
627  * Set automatic "Amount of References" markup active?
628  */
629
630 $SocialRewarding["references"]["autoMarkup"]            = true;
631
632
633
634 /*******************************************************
635  * RATING OF ARTICLES CONFIGURATION
636  *******************************************************/
637
638 /**
639  * Activate social rewarding mechanism "Rating of Articles"?
640  */
641
642 $SocialRewarding["rating"]["active"]                    = true;
643
644
645 /**
646  * Rating scale (points a user can vote for an article).
647  */
648
649 $SocialRewarding["rating"]["scale"]                     = "0,1,2,3,4,5";
650
651
652 /**
653  * Allow users to leave a comment (beside the rating). Comments are
654  * displayed on the talk (discussion) page of the article. Can on each page
655  * manually be turned off from an author. Markups can be inserted using
656  * "<SocialRewardingRatingOfArticles></SocialRewardingRatingOfArticles>".
657  * If you want to deactivate comments, set attribute comment=false, e.g.
```

```
658   * "<SocialRewardingRatingOfArticles comment=false>".
659   */
660
661  $SocialRewarding["rating"]["comment"]                    = true;
662
663
664  /**
665   * Standard size for comment text field. Can be modified on each page
666   * from an author, e.g. set attribute "size=20".
667   */
668
669  $SocialRewarding["rating"]["commentStdSize"]             = 32;
670
671
672  /**
673   * Standard maximum character length for comments. Can be modified on
674   * each page from an author, e.g. set attribute "maxlength=100".
675   */
676
677  $SocialRewarding["rating"]["commentStdMaxLength"]        = 255;
678
679
680  /**
681   * Should new comments be inserted on top of talk pages?
682   */
683
684  $SocialRewarding["rating"]["commentNewOnTop"]            = true;
685
686
687  /**
688   * Should voted points of a user be displayed on talk pages?
689   */
690
691  $SocialRewarding["rating"]["commentDisplayPoints"]       = true;
692
693
694  /**
695   * Standard caption of rating button. Can be modified on each page
696   * from an author, e.g. set attribute "buttoncaption='Go'".
697   */
698
699  $SocialRewarding["rating"]["stdButtonCaption"]           = " Vote > ";
700
701
702  /**
703   * Should a JavaScript window be displayed after voting? Can be modified
704   * on each page from an author, e.g. set attribute "popup=true".
705   */
706
707  $SocialRewarding["rating"]["popup"]                      = true;
708
709
710  /**
711   * Standard message for JavaScript window. Can be modified
712   * on each page from an author, e.g. set attribute
713   * "popupmsg='Your vote was counted, thank you.'".
714   */
715
716  $SocialRewarding["rating"]["stdPopupMsg"]                = "Thank you for your ";
717  $SocialRewarding["rating"]["stdPopupMsg"].               = "vote.";
718
719
720  /**
721   * Should a vote from an author of an article be counted?
722   */
723
```

```
724 $SocialRewarding["rating"]["countAuthor"]                = true;
725
726
727 /**
728  * Should only logged in users have the possibility to vote?
729  */
730
731 $SocialRewarding["rating"]["onlyUsers"]                  = false;
732
733
734 /**
735  * Should users have the possibility to vote several times?
736  */
737
738 $SocialRewarding["rating"]["multipleVotes"]              = true;
739
740
741 /**
742  * Timeout before next vote (in seconds), 0 = no timeout. Important only
743  * if $SocialRewarding["rating"]["multipleVotes"] is set true.
744  */
745
746 $SocialRewarding["rating"]["voteTimeout"]                = 0;
747
748
749 /**
750  * Standard markup text for "Rating of Articles". If a user does not set
751  * her/his own markup message, this one is used. Markups can be
752  * inserted using
753  * "<SocialRewardingRatingPoints> Text </SocialRewardingRatingPoints>".
754  */
755
756 $SocialRewarding["rating"]["markupStdPointsText"]        = "This article was ";
757 $SocialRewarding["rating"]["markupStdPointsText"].       = "rated $1 time(s) ";
758 $SocialRewarding["rating"]["markupStdPointsText"].       = "with a score of $2 ";
759 $SocialRewarding["rating"]["markupStdPointsText"].       = "point(s) (until $3).";
760
761
762 /**
763  * Set automatic "Rating of Articles" (rating form) markup active?
764  */
765
766 $SocialRewarding["rating"]["autoMarkup"]                 = true;
767
768
769 /**
770  * Set automatic "Rating of Articles" (rating points) markup active?
771  */
772
773 $SocialRewarding["rating"]["autoMarkupPoints"]           = true;
774
775
776
777 /*******************************************************
778  * MOST VIEWED ARTICLES CONFIGURATION
779  *******************************************************/
780
781 /**
782  * Activate social rewarding mechanism "Most Viewed Articles"?
783  */
784
785 $SocialRewarding["viewed"]["active"]                     = true;
786
787
788 /**
789  * Standard markup text for "Most Viewed Articles". If a user does not set
```

```
790   * her/his own markup message, this one is used. Markups can be
791   * inserted using
792   * "<SocialRewardingMostViewedArticles show=true> Text
793   * </SocialRewardingMostViewedArticles>".
794   */
795
796  $SocialRewarding["viewed"]["stdMessage"]                = "This article has been ";
797  $SocialRewarding["viewed"]["stdMessage"].               = "accessed $1 time(s) ";
798  $SocialRewarding["viewed"]["stdMessage"].               = "(until $2).";
799
800
801  /**
802   * Counting method: "hits" or "visits". "hits" means that every request
803   * for a page is counted, "visits" that within a certain period of time
804   * (defined in $SocialRewarding["viewed"]["sessionTimeout"]) only one
805   * visit is counted regardless how often a user requests a page.
806   */
807
808  $SocialRewarding["viewed"]["countMethod"]               = "hits";
809
810
811  /**
812   * Count hits or visits from authors of an article?
813   */
814
815  $SocialRewarding["viewed"]["countAuthor"]               = true;
816
817
818  /**
819   * Session timeout in seconds before next visit is counted. Only important
820   * if $SocialRewarding["viewed"]["countMethod"] = "visits".
821   */
822
823  $SocialRewarding["viewed"]["sessionTimeout"]            = 1800;
824
825
826  /**
827   * Do also compute hits or visits on user-pages?
828   */
829
830  $SocialRewarding["viewed"]["addUserPages"]              = true;
831
832
833  /**
834   * Scale for calculating points of an article in percent, e.g. if an
835   * article has over 100 and under 200 percent visitors relating to the average
836   * visiting rate calculated for all articles, it gets three points.
837   */
838
839  $SocialRewarding["viewed"]["articleScale"]              = "0,25,50,100,200,300";
840
841
842  /**
843   * Basis for calculating social rewarding mechanism "Most Viewed Articles":
844   * "all", "articles", or "user_pages".
845   */
846
847  $SocialRewarding["viewed"]["calcBasis"]                 = "all";
848
849
850  /**
851   * Corrects calculation basis, that means excluding system pages
852   * (=pages with namespace like
853   * $SocialRewarding["reward"]["calcBasisCorrectionNS"]).
854   */
855
```

```
856 $SocialRewarding["viewed"]["calcBasisCorrection"]       = true;
857
858
859 /**
860  * Set automatic "Most Viewed Articles" markup active?
861  */
862
863 $SocialRewarding["viewed"]["autoMarkup"]                = true;
864
865
866
867 /*******************************************************
868  * RECOMMENDER SYSTEM CONFIGURATION
869  *******************************************************/
870
871 /**
872  * Activate social rewarding mechanism "Recommender System"?
873  */
874
875 $SocialRewarding["recommend"]["active"]                 = true;
876
877
878 /**
879  * Counting method: "hits" or "visits".
880  */
881
882 $SocialRewarding["recommend"]["countMethod"]            = "hits";
883
884
885 /**
886  * Count hits or visits from authors of an article?
887  */
888
889 $SocialRewarding["recommend"]["countAuthor"]            = true;
890
891
892 /**
893  * Session timeout in seconds before next visit is counted. Only important
894  * if $SocialRewarding["recommend"]["countMethod"] = "visits".
895  */
896
897 $SocialRewarding["recommend"]["sessionTimeout"]         = 1800;
898
899
900 /**
901  * Do also compute hits or visits on user-pages?
902  */
903
904 $SocialRewarding["recommend"]["addUserPages"]           = true;
905
906
907 /**
908  * Calculate recommendations on how much top-articles of an author?
909  */
910
911 $SocialRewarding["recommend"]["reduceData"]             = 5;
912
913
914 /**
915  * Counting every article as one ("equal") or sum up visits from an
916  * article ("repr"esentative).
917  */
918
919 $SocialRewarding["recommend"]["countingMethod"]         = "repr";
920
921
```

```
922 /**
923  * Activate weighting of articles so to set articles which has been
924  * more often visited by a user more important.
925  */
926
927 $SocialRewarding["recommend"]["weighting"]              = true;
928
929
930 /**
931  * "equal" means that the graduation of weighted articles is always
932  * the same regardless of visits. "repr"esentative weights articles
933  * on the basis of their visits.
934  */
935
936 $SocialRewarding["recommend"]["weightingMethod"]        = "repr";
937
938
939 /**
940  * How much recommendations should be displayed?
941  */
942
943 $SocialRewarding["recommend"]["reduceRecommendation"]   = 5;
944
945
946 /**
947  * Exclude user's visited articles?
948  */
949
950 $SocialRewarding["recommend"]["excludeVisitedArticles"] = true;
951
952
953 /**
954  * Exclude articles where user is author?
955  */
956
957 $SocialRewarding["recommend"]["excludeUsersArticles"]   = true;
958
959
960 /**
961  * Standard markup text for "Recommender System". If a user does not set
962  * her/his own markup message, this one is used. Markups can be
963  * inserted using
964  * "<SocialRewardingRecommend> Text </SocialRewardingRecommend>".
965  */
966
967 $SocialRewarding["recommend"]["markupStdMessage"]       = "Other interesting ";
968 $SocialRewarding["recommend"]["markupStdMessage"].      = "authors:";
969
970
971 /**
972  * Standard "Recommender System" markup method ("revision", "article",
973  * "author", or "interestedAuthor"). Can be modified on each page
974  * from an author, e.g. set attribute "method=article".
975  */
976
977 $SocialRewarding["recommend"]["markupStdMethod"]        = "author";
978
979
980 /**
981  * Display rank/position as standard? Can be modified on each page
982  * from an author, e.g. set attribute "rank=true".
983  */
984
985 $SocialRewarding["recommend"]["markupStdRank"]          = false;
986
987
```

```
988  /**
989   * Display points as standard? Can be modified on each page from an
990   * author, e.g. set attribute "points=true".
991   */
992
993  $SocialRewarding["recommend"]["markupStdPoints"]        = false;
994
995
996  /**
997   * Set automatic "Recommender System" markup active?
998   */
999
1000 $SocialRewarding["recommend"]["autoMarkup"]             = true;
1001
1002
1003
1004 /*********************************************************
1005  * DATABASE TABLES CONFIGURATION
1006  *********************************************************/
1007
1008 /**
1009  * If you want to rename database tables used in the SocialRewarding
1010  * extension, do it here. If you have set database tables' prefixes
1011  * in your MediaWiki, they are added automatically.
1012  */
1013
1014
1015 /**
1016  * Table to store cached objects in database (only needed if
1017  * $SocialRewarding["reward"]["cache"] = true and
1018  * $SocialRewarding["reward"]["cacheMethod"] = "db").
1019  */
1020
1021 $SocialRewarding["DB"]["cache"]                         = "sr__cache";
1022
1023
1024 /**
1025  * Table to store data for social rewarding mechanism "Amount of
1026  * References".
1027  */
1028
1029 $SocialRewarding["DB"]["references"]                    = "sr__references";
1030
1031
1032 /**
1033  * Table to store data for multiple votes and timeout of votes ("Rating
1034  * of Articles").
1035  */
1036
1037 $SocialRewarding["DB"]["ratedRevision"]                 = "sr__ratedrevision";
1038
1039
1040 /**
1041  * Table to store data for social rewarding mechanism "Rating of
1042  * Articles".
1043  */
1044
1045 $SocialRewarding["DB"]["rating"]                        = "sr__rating";
1046
1047
1048 /**
1049  * Table to store data for social rewarding mechanism "Most Viewed
1050  * Articles".
1051  */
1052
1053 $SocialRewarding["DB"]["viewedArticles"]                = "sr__viewedarticles";
```

```
1054
1055
1056 /**
1057  * Table to store data for multiple visits and timeout of visits ("Most
1058  * Viewed Articles").
1059  */
1060
1061 $SocialRewarding["DB"]["visitRevision"]                    = "sr__visitrevision";
1062
1063
1064 /**
1065  * Table to store data for social rewarding mechanism "Recommender
1066  * System".
1067  */
1068
1069 $SocialRewarding["DB"]["recommend"]                        = "sr__recommend";
1070
1071
1072 ?>
```

File 2: SocialRewardingConfigDetail.php

D Explanation of Files and Directories

This chapter contains an explanation of files and directories included in the SocialRewarding package. The list appears as displayed in the file system: alphabetically ordered and directories before files.

`SocialRewardingImg/`	Directory where all images are placed which are used by the SocialRewarding package.
`sparkline-php-0.2/`	Directory containing the sparkline package which is called from file `SocialRewardingSparkLine.php` to generate sparklines.
`SocialRewarding.php`	This is the main file for loading the whole SocialRewarding extension.
`SocialRewardingConfigDetail.php`	Detailed config file of the SocialRewarding package. If one feels the need to configure something, she/he can do it here.
`SocialRewardingConfigShort.php`	Short config file of the SocialRewarding package. One can configure the SocialRewarding extension either by using this file or `SocialRewardingConfigDetail.php` depending on one's settings in file `SocialRewarding.php`.
`SocialRewardingCOPYING`	Documentation of the GNU GENERAL PUBLIC LICENSE which is used to distribute the SocialRewarding package.
`SocialRewardingDocumentation.pdf`	Detailed documentation of the whole SocialRewarding extension.
`SocialRewardingGoogleSearch.php`	Class for query Google searches.
`SocialRewardingHooks.php`	Activate hooks if defined social rewarding methods are enabled (in config file).

SocialRewardingHooksFunctions.php	Define all hook functions which are loaded in file SocialRewardingHooks.php.
SocialRewardingINSTALL	Dummy file for installing the SocialRewarding extension.
SocialRewardingMarkups.php	Activate markups if social rewarding methods are enabled (in config file).
SocialRewardingMarkupsFunctions.php	Define all markup functions which execution is specified in SocialRewardingMarkups.php.
SocialRewardingOtherFunctions.php	Container for all useful functions which are not defined elsewhere.
SocialRewardingREADME	Quick documentation of the package, installation requirements, and the installation process.
SocialRewardingReward.php	Main class for computing points for all three social rewarding methods and displaying stars and sparklines.
SocialRewardingRewardManage.php	Class for handling cache administration.
SocialRewardingRewardRating.php	Class for computing points for a revision for social rewarding method Rating of Articles.
SocialRewardingRewardRecommend.php	Class containing all functions for social rewarding method Recommender System.
SocialRewardingRewardReferences.php	Class for computing points for a revision for social rewarding method Amount of References.
SocialRewardingRewardViewed.php	Class for computing points for a revision for social rewarding method Most Viewed Articles.
SocialRewardingSparkLine.php	File for displaying sparklines. Calls needed methods of sparkline package (which is located in directory sparkline-php-0.2) to generate an image which is returned.

SocialRewardingSpecialPages.php	Activate display of special pages if social rewarding methods are enabled (in config file).
SocialRewardingTables.sql	DDL statements for creating MySQL database tables.
SpecialSocialRewardingAuthorsHistory.php	Special page for displaying a history of ranking of authors. Only available if caching is activated and caching method is set to “db” (in config file).
SpecialSocialRewardingAuthorsRanking.php	Special page for displaying a ranking of authors.
SpecialSocialRewardingInstall.php	Special page for SocialRewarding package installation.
SpecialSocialRewardingListUsers.php	This page was adapted from the MediaWiki special page SpecialListusers.php. It lists all users and displays stars beside their usernames according to the amount of points computed by the social rewarding methods.
SpecialSocialRewardingMostViewedArticles.php	Special page for displaying a list of most viewed articles.
SpecialSocialRewardingMostViewedRevisions.php	Special page for displaying a list of most viewed revisions.
SpecialSocialRewardingRatingArticles.php	Special page for displaying a list of rated articles.
SpecialSocialRewardingRatingRevisions.php	Special page for displaying a list of rated revisions.
SpecialSocialRewardingRecommenderSystem.php	Special page for displaying a list of recommended revisions, articles, authors, and authors with same interests.
SpecialSocialRewardingReferencesArticles.php	Special page for displaying a list of articles with amount of references.
SpecialSocialRewardingReferencesRevisions.php	Special page for displaying a list of revisions with amount of references.

Table 9: Explanation of files and directories.

List of Abbreviations

API	Application Programming Interface
C2C	Consumer To Consumer
CSCW	Computer-Supported Cooperative Work
CVS	Concurrent Versioning System
DB	Database
DDL	Data Definition Language
EDI	Electronic Data Interchange
ER	Entity-Relationship
FAQ	Frequently Asked Questions
FK	Foreign Key
GNU	GNU's Not UNIX
GPL	GNU General Public License
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
ID	Identity Document
IRC	Internet Relay Chat
IT	Information Technology
MIT	Massachusetts Institute of Technology
MRI	Magnetic Resonance Imaging
MySQL	My Structured Query Language
PC	Personal Computer
PHP	PHP Hypertext Preprocessor
PK	Primary Key
SMS	Short Message Service
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
TCP/IP	Transmission Control Protocol/Internet Protocol
UML	Unified Modelling Language
URL	Uniform Resource Locator
US	United States
WWW	World Wide Web
XML	eXtended Markup Language

List of Figures

List of Tables

List of Files

Bibliography

[Abo07] About, Inc. *The Depression Dictionary – R*. `http://depression.about.com/library/glossary/blglossaryindexr.htm`, retrieved on 31 May 2007.

[Ada00] Adar, Eytan; Huberman, Bernardo A. *Free Riding on Gnutella*. Xerox Palo Alto Research Center, 2000.

[Ban06] Bandura, Albert. *Albert Bandura Vita*. Emory University, Atlanta, 2006.

[Bar99] Barabási, Albert-László; Albert, Réka. *Emergence of Scaling in Random Networks*. Science, 286: pages 509–512, 1999.

[Bar05] Barto, Andrew G.; Simsek, Özgür. *Intrinsic Motivation for Reinforcement Learning Systems*. Department of Computer Science, University of Massachusetts, Amherst, 2005.

[Boe07] Boettcher, Sue. *What Types of Virtual Communities can I Build and what Tools are Available?* `http://www.fullcirc.com/community/communitytypes.htm`, retrieved on 31 May 2007.

[But99] Butler, Brian S. *When is a Group not a Group: An Empirical Examination of Metaphors for Online Social Structure*. PhD thesis, Carnegie Mellon University, 1999.

[Byr97] Byrne, Donn. *An Overview (and Underview) of Research and Theory within the Attraction Paradigm*. Journal of Social & Personal Relationships, 14 (3): pages 417–431, 1997.

[Che06] Chesney, Thomas. *An Empirical Examination of Wikipedia's Credibility*. First Monday, 11, 2006.

[Cor96] Cornes, Richard; Sandler, Todd. *The Theory of Externalities, Public Goods, and Club Goods*. Cambridge University Press, 2nd edition, 1996.

[Cos05] Cosley, Dan; Frankowski, Dan; Kiesler, Sara et al. *How Oversight Improves Member-Maintained Communities*. CommunityLab, 2005.

[Dec87] Deci, Edward L.; Ryan, Richard M. *The Support of Autonomy and the Control of Behavior*. Journal of Personality and Social Psychology, 53 (4): pages 1024–1037, 1987.

[Dec00] Deci, Edward L.; Ryan, Richard M. *The "What" and "Why" of Goal Pursuits: Human Needs and the Self-Determination of Behavior*. Psychological Inquiry, 11: pages 227–268, 2000.

[Dix03] Dix, Alan; Finlay, Janet; Abowd, Gregory et al. *Human-Computer Interaction*. Prentice Hall, 3rd edition, 2003.

[eBa07] eBay Inc. *eBay Help: Feedback*. `http://pages.ebay.com/help/feedback/index.html`, retrieved on 31 May 2007.

[Ebe05] Ebersbach, Anja; Glaser, Markus; Heigl, Richard. *Wiki: Web Collaboration*. Springer, Berlin Heidelberg New York, 2005.

[Fel04] Feldman, Michal; Papadimitriou, Christos; Chuang, John et al. *Free-Riding and Whitewashing in Peer-to-Peer Systems*. School of Information Management and Systems, UC Berkeley, 2004.

[Fis01] Fischer, Gerhard. *Communities of Interest: Learning through the Interaction of Multiple Knowledge Systems*. IRIS (24th annual Information Systems Research Seminar in Scandinavia), Ulvik, Hardanger Fjord, Norway, 2001.

[Fre07] Free Software Foundation, Inc. *GNU Free Documentation License*. `http://www.gnu.org/licenses/fdl.txt`, retrieved on 31 May 2007.

[Ger06] Geroimenko, Vladimir; Chen, Chaomei. *Visualizing the Semantic Web*. Springer, London, 2nd edition, 2006.

[Goo07] Google. *Google Technology*. `http://www.google.com/technology/`, retrieved on 31 May 2007.

[Har82] Harkins, Stephen G.; Petty, Richard E. *Effects of Task Difficulty and Task Uniqueness on Social Loafing*. Journal of Personality and Social Psychology, 43: pages 1214–1229, 1982.

[Har07] Hardin, Russell. *The Free Rider Problem*. `http://plato.stanford.edu/entries/free-rider`, retrieved on 31 May 2007.

[Hoi07a] Hoisl, Bernhard; Aigner, Wolfgang; Miksch, Silvia. *Social Rewarding in Wiki Systems – Motivating the Community*. In *Online Communities and Social Computing*, pages 362–371. Springer, Berlin Heidelberg New York, 2007.

[Hoi07b] Hoisl, Bernhard; Aigner, Wolfgang; Miksch, Silvia. *Soziale Belohnung in Wiki Systemen*. In *Wikis im Social Web – Wikiposium 2005/06*, pages 60–72. OCG – Austrian Computer Society, Vienna, 2007.

[Jac85] Jackson, Jeffrey M.; Harkins, Stephen G. *Equity in Effort: An Explanation of the Social Loafing Effect*. Journal of Personality and Social Psychology, 49: pages 1199–1206, 1985.

[Kar93] Karau, Steven J.; Williams, Kipling D. *Social Loafing: A Meta-Analytic Review and Theoretical Integration.* Journal of Personality and Social Psychology, 65 (4): pages 681–706, 1993.

[Kar01] Karau, Steven J.; Williams, Kipling D. *Understanding Individual Motivation in Groups: The Collective Effort Model.* In *Groups at Work: Theory and Research*, pages 113–141. Lawrence Erlbaum Associates, 2001.

[Ker83] Kerr, Norbert L. *Motivation Losses in Small Groups: A Social Dilemma Analysis.* Journal of Personality and Social Psychology, 45 (4): pages 819–828, 1983.

[Kim00] Kim, Amy Jo. *Community Building on the Web*. Peachpit Press, Berkeley, 2000.

[Kle99] Kleinberg, Jon. *The Small-World Phenomenon: An Algorithmic Perspective.* Cornell Computer Science Technical Report, pages 99-110, 1999.

[Lak03] Lakhani, Karim R.; Hippel, Eric von. *How Open Source Software Works: "Free" User-to-User Assistance.* Research Policy (Special Issue), 32 (6): pages 923–943, 2003.

[Led95] Ledyard, John. *Public goods: A Survey of Experimental Research.* In *The Handbook of Experimental Economics*, pages 111–194. Princeton University Press, 1995.

[Leu01] Leuf, Bo; Cunningham, Ward. *The Wiki Way: Quick Collaboration on the Web*. Addison-Wesley Professional, 2001.

[Lin04] Ling, Kimberly; Beenen, Gerard; Ludford, Pamela et al. *Using Social Psychology to Motivate Contributions to Online Communities.* Journal of Computer-Mediated Communication, 10 (4), 2004.

[Loc90] Locke, Edwin A.; Latham, Gary P. *A Theory of Goal Setting and Task Performance*. Englewood Cliffs, 1990.

[Lud04] Ludford, Pamela; Cosley, Dan; Frankowski, Dan et al. *Think Different: Increasing Online Community Participation Using Uniqueness and Group Dissimilarity*. University of Minnesota, Department of Computer Science and Engineering, 2004.

[Mal02] Malonis, Jane A.; Braun, Erin; Mueckenheim, Jacqueline K. et al. *Gale Encyclopedia of E-Commerce*, volume 2. Gale Group, 2002.

[Man96] Manville, Brook; Foote, Nathaniel. *Harvest Your Workers' Knowledge.* Datamation, 46: pages 78–80, 1996.

[Man04] Mansfield, Edwin; Yohe, Gary. *Microeconomics – Theory / Applications*. W. W. Norton & Company, 11th edition, 2004.

[Mas87] Maslow, Abraham H. *Motivation and Personality*. Harper & Row, Publishers, 3rd edition, 1987.

[McF06] McFederies, Paul. *Technically Speaking: It's A Wiki, Wiki World*, volume 43, page 88. IEEE Spectrum, 2006.

[McG85] McGregor, Douglas. *The Human Side of Enterprise*. McGraw-Hill Book Company, 1985.

[Med07a] MediaWiki. *Help:Special Pages*. `http://www.mediawiki.org/wiki/Help:Special_pages`, retrieved on 31 May 2007.

[Med07b] MediaWiki. *How does MediaWiki Work?* `http://www.mediawiki.org/wiki/How_does_MediaWiki_work%3F`, retrieved on 31 May 2007.

[Med07c] MediaWiki. *Manual:MediaWiki Hooks*. `http://www.mediawiki.org/wiki/Manual:MediaWiki_hooks`, retrieved on 31 May 2007.

[Med07d] MediaWiki. *MediaWiki Documentation*. `http://svn.wikimedia.org/doc`, retrieved on 31 May 2007.

[Med07e] MediaWiki. *Welcome to MediaWiki.org*. `http://www.mediawiki.org/wiki/MediaWiki`, retrieved on 31 May 2007.

[Men05] Mendling, Jan. *Computer-Mediated Cooperative Work*. Vienna University of Economics and Business Administration, 2005.

[Mil67] Milgram, Stanley. *The Small World Problem*. Psychology Today, 2: pages 60–67, 1967.

[Moc02] Mockus, Audris; Fielding, Roy T.; Herbsleb, James D. *Two Case Studies of Open Source Software Development: Apache and Mozilla*, volume 11, pages 309–346. ACM Transactions on Software Engineering and Methodology, 2002.

[Mun00] Munzner, Tamara. *Interactive Visualization of Large Graphs and Networks*. PhD thesis, Stanford University, 2000.

[Nem74] Nemeth, Charlan; Swedlund, Mark; Kanki, Barbara. *Patterning of the Minority's Responses and their Influence on the Majority*. European Journal of Social Psychology, 4: pages 53–64, 1974.

[Nic05a] Nicholson, Walter. *Microeconomic Theory – Basic Principles and Extensions*. South-Western, Thomson, 9th edition, 2005.

[Nic05b] Nickols, Fred. *The Autonomous Performer – Some Implications of Configured Work*. Distance Consulting, 2005.

[Non00] Nonnecke, Blair; Preece, Jennifer. *Lurker Demographics: Counting the Silent.* ACM Conference on Human Factors in Computing Systems, CHI Letters, 4 (1): pages 73–80, 2000.

[Odl05] Odlyzko, Andrew; Tilly, Benjamin. *A Refutation of Metcalfe's Law and a Better Estimate for the Value of Networks and Network Interconnections*. University of Minnesota, 2005.

[One05] Oneupweb. *The Guide to Search Engine Marketing*. So Do It, LLC / Oneupweb, 2005.

[Pen99] Pennock, David M.; Horvitz, Eric. *Analysis of the Axiomatic Foundations of Collaborative Filtering*. AAAI Workshop on Artificial Intelligence for Electronic Commerce, Orlando, Florida, 1999.

[Pot05] Potanin, Alex; Noble, James; Frean, Marcus et al. *Scale-Free Geometry in OO Programs.* Communications of the ACM, 48 (5): pages 99–103, 2005.

[Puc02] Pucher, Robert; Mense, Alexander; Wahl, Harald. *How to Motivate Students in Project Based Learning.* IEEE Africon, 1: pages 443–446, 2002.

[Rac95] Rachlin, Howard. *Burrhus Frederic Skinner.* National Academia Press, 67: pages 362–377, 1995.

[Rei02] Reiss, Steven. *Who am I? The 16 Basic Desires that Motivate our Actions and Define our Personalities*. Berkley Publishing Group, 2002.

[Res97] Resnick, Paul; Varian, Hal R. *Recommender Systems.* Communications of the ACM, 40 (3): pages 56–58, 1997.

[Res00] Resnick, Paul; Zeckhauser, Richard; Friedman, Eric et al. *Reputation Systems: Facilitating Trust in Internet Interactions.* Communications of the ACM, 43: pages 45–48, 2000.

[San07] Sanger, Larry. *Why Wikipedia Must Jettison Its Anti-Elitism.* `http://www.kuro5hin.org/story/2004/12/30/142458/25`, retrieved on 31 May 2007.

[Sar04] Sarma, Anita; Hoek, Andrè van der. *A Need Hierarchy for Teams*. ISR Technical Report, University of California, Irvine, 2004.

[Sch71] Schramm, Wilbur. *The Nature of Communication between Humans.* In *The Process and Effects of Mass Communication*, pages 3–53. University of Illinois Press, Urbana Chicago London, 1971.

[Sch03] Schoberth, Thomas; Preece, Jenny; Heinzl, Armin. *Online Communities: A Longitude Analysis of Communication Activities.* Working Paper in Information Systems, 2003.

[Sta05] Stasko, John. *Social Visualization.* Georgia Institute of Technology, 2005.

[Ste87] Stegner, Wallace; Nelson, Lyle M.; Maccoby, Nathan. *Memorial Resolution – Wilbur Lang Schramm.* Stanford Historical Society, 1987.

[Sto06] Stock, Oliviero; Guerini, Marco; Strapparava, Carlo et al. *Mid-term Report on Communication & Persuasion Exemplar Progress.* IST – Information Society Technologies, 2006.

[Tan04] Tanasa, Doru; Trousse, Brigitte. *Advanced Data Preprocessing for Intersites Web Usage Mining.* IEEE Intelligent Systems, 19 (2): pages 59–65, 2004.

[Tap06] Tapscott, Don; Williams, Anthony D. *Wikinomics: How Mass Collaboration Changes Everything.* Portfolio, 2006.

[Tuf06] Tufte, Edward R. *Beautiful Evidence.* Graphics Press, 2006.

[Und07] Underwood, Mick. *Learning: Social Learning Theory.* `http://www.cultsock.ndirect.co.uk/MUHome/cshtml/psy/soclearn.html`, retrieved on 31 May 2007.

[Var06] Varian, Hal R. *Intermediate Microeconomics.* W. W. Norton & Company, 7th edition, 2006.

[Vit06] Vitasek, Kate. *Supply Chain and Logistics Terms and Glossary.* Supply Chain Visions, Washington, 2006.

[Wen98] Wenger, Etienne. *Communities of Practice: Learning, Meaning, and Identity.* Cambridge University Press, 1998.

[Whi07] White, Nancy. *Community Member Roles and Types* . `http://www.fullcirc.com/community/memberroles.htm`, retrieved on 31 May 2007.

[Wij05] Wijk, Jarke J. van. *The Value of Visualization.* IEEE Visualization, pages 79–86, 2005.

[Wik07a] Wikimedia. *Complete List of Wikimedia Projects.* `http://meta.wikimedia.org/wiki/Complete_list_of_Wikimedia_projects`, retrieved on 31 May 2007.

[Wik07b] Wikipedia. *Category: Wikipedia Statistics*. http://en.wikipedia.org/wiki/Category:Wikipedia_statistics, retrieved on 31 May 2007.

[Wik07c] Wikipedia. *Communities of Purpose*. http://en.wikipedia.org/wiki/Communities_of_Purpose, retrieved on 31 May 2007.

[Wik07d] Wikipedia. *Featured Article Criteria*. http://en.wikipedia.org/wiki/Wikipedia:Featured_article_criteria, retrieved on 31 May 2007.

[Wik07e] Wikipedia. *Image: Network Effect*. http://en.wikipedia.org/wiki/Image:Network_effect.png, retrieved on 31 May 2007.

[Wik07f] Wikipedia. *Image: Scale-Free Network Sample*. http://en.wikipedia.org/wiki/Image:Scale-free_network_sample.png, retrieved on 31 May 2007.

[Wik07g] Wikipedia. *Image: Social-Network*. http://en.wikipedia.org/wiki/Image:Social-network.png, retrieved on 31 May 2007.

[Wik07h] Wikipedia. *Metcalfe's Law*. http://en.wikipedia.org/wiki/Metcalfe's_law, retrieved on 31 May 2007.

[Wik07i] Wikipedia. *Social Network*. http://en.wikipedia.org/wiki/Social_network, retrieved on 31 May 2007.

[Wik07j] Wikipedia. *Statistics*. http://en.wikipedia.org/wiki/Wikipedia:Statistics, retrieved on 31 May 2007.

[Wik07k] Wikipedia. *Stub*. http://en.wikipedia.org/wiki/Wikipedia:Stub, retrieved on 31 May 2007.

[Wik07l] Wikipedia. *Wikipedia, the free Encyclopedia*. http://en.wikipedia.org/wiki/Wikipedia, retrieved on 31 May 2007.

[Wik07m] Wikipedia. *Wikipedia:How to Edit a Page*. http://en.wikipedia.org/wiki/Wikipedia:How_to_edit_a_page, retrieved on 31 May 2007.

[Wil98] Williams, Katherine; O'Reilly, Charles A. *Demography and Diversity in Organizations: A Review of 40 Years of Research*. Research in Organizational Behavior, 20: pages 77–140, 1998.

[Zip49] Zipf, George K. *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley, Cambridge, 1949.