Towards a Quality Service Layer for Web 2.0

Markus Schaal¹, David Davenport², and Ali Hamdi Cevik²

¹ METU-NCC, Kalkanli, Güzelyurt, KKTC
schaal@acm.org

² Bilkent University, Ankara, Turkey
david@cs.bilkent.edu.tr, a_cevik@ug.bilkent.edu.tr

Abstract. Despite the help of search engines and Web directories, identifying high quality content becomes increasingly difficult as the Internet gets ever more crowded with information.

Prior approaches for filtering and searching content with respect to user-specific preferences do exist: Recommendation engines employ collaborative filtering to support subjective selection, (semi-)automatic page ranking algorithms utilize the hypertext link structure of the World Wide Web to assess page importance, and trust-based systems employ social network analysis to determine the most suitable Web pages. The use of *implicit and explicit user feedback*, however, is often either ignored or its exploitation is limited to isolated Web sites. We thus propose a *quality overlay framework* that enables the collection and *processing of user-feedback*, and the subsequent *presentation of quality-enabled content* for any Web-site.

We present the quality overlay framework, propose an architecture for its realization, and validate our approach by scenarios and a detailed design with sample implementation.

1 Introduction

With the emergence of Web 2.0 applications, where information is not only disseminated from trusted sources across the net, but also anonymously published, syndicated, evaluated, selected, edited and recombined, information quality assessment becomes crucial. Wikipedia, for example, has already begun to face this challenge as the number of authors has diminished compared to the amount of knowledge that needs to be maintained.

In response to this there has been a growth in applications such as image tagging, recommender and recommendation engines, that exploit the wisdom of the crowd to filter out the best, most relevant, information and so improve quality.

In this article, we propose a quality service layer on top of existing Web applications. The quality service layer is responsible for the collection of implicit and explicit user feedback, for the processing of quality data, and for the navigation in quality-enabled content, essentially independent of the underlying content application server.

The *quality service layer* is depicted in Fig. 1 as a mediator between legacy content applications and quality-enabled applications. It combines and supports any type of quality enabling while building on both implicit feedback (e.g. *link structure* as exploited by Google Page Rank) and explicit feedback (e.g. *transaction feedback* as exploited by E-Bay's reputation system). We foresee novel applications such as the *Active*

Fig. 1. The Quality Service Layer

Classroom, the *Informed Customer*, and *Advanced Search*, which will be explained in more detail in Section 4.1.

At the core of the quality layer, the following concepts are to be supported:

User & Content Qualities. Feedback may differ in quality depending on its source. Therefore, the concept of user-related quality dimensions in addition to the content-related quality dimensions supports the processing of quality.

Sophisticated Quality Feedback. Users may be faced with either simple or sophisticated feedback options.

Online Quality Processing. Quality should be processed online as a function of time, allowing for both up-to-date quality assessments and adaptation to changes.

We support the validity of this framework in two ways. Firstly, we designed a software system for the deployment of the quality service layer under different conditions, and implemented one architectural alternative as an add-on module for the Moodle course management system. This way, we were able to stabilize the framework on its top-level by an in-depth technical evaluation of its implementation. Secondly, we developed three scenarios to prove the usability of the proposed framework. Both approaches are described in detail in Section 4.

2 Related Work

PageRank [5] and similar approaches such as OPIC [1] evaluate the importance of a Web page based on the link structure among Web sites. The underlying idea is simple: important pages link to other important pages. In PageRank, the importance of a

page depends on both the number of incoming links to the page and the importance of the pages which give those links. Google describes the concept with a non-egalitarian voting mechanism, where a link from one page is interpreted as a vote for this page.

The alternative way to evaluate Web resources is the collection and processing of implicit and explicit user feedback. While there are a number of commercial Web sites that collect feedback from users, such attempts are piecemeal and so cannot be applied generally to other sites. We thus propose an overlay architecture for the collection and incorporation of user-feedback. We are not aware of any previous attempts to do this, although there is some relevant research work, e.g. Lykourantzou et al. [7] propose a corporate Wiki, where articles are peer-reviewed by other employees and the most relevant peer-feedback is identified by a neural network. Averbakh et al. [4] present an approach for the incorporation of user feedback into the selection process for Semantic Web Service Discovery. Wang et al. [13] did a survey to incorporate the community and information quality aspect into the analysis of Wikipedia use and adoption. Quingliang et al. [8] have a different focus, they propose a framework for opinion retrieval that builds on cross-analysis of opinions and their interaction, employing a probability update model. All of these recent research papers have one aspect in common: they rely on user-feedback for quality assessment.

For the processing of user feedback, the notions of trust and belief are crucial. The bridge between trust and information quality is being investigated for multi-agent systems (e.g. Sabater & Sierra [10]), and slowly applied to the Social Web (e.g. Golbeck & Hendler [6]), and innovative applications (e.g. Schaal [11]). Recommender systems and collaborative filtering (e.g. Adomavicius & Tuzhilin [2]) are other approaches for the aggregation and mining of collective reputation, but still tend to neglect the notion of trust among people.

3 Quality Service Layer

We propose a *quality service layer* in order to enable *quality evaluation* based on *implicit and explicit feedback* associated with *content* available on the Internet. The quality service layer supports *explicit and implicit feedback collection from the user*, comprehensive *quality assessment for many content items*, and the *visualization of quality-decorated content*. The so-called *quality service* builds on the notion of *quality as a property of content*, to be *assessed or judged by human users*.

The quality service layer is not limited to a particular set of Web resources and in principle supports quality-enrichment for any subset of content available on the internet.

3.1 Basic Concepts

A *quality target* is anything for which quality assessment is required. We consider *users* and *content items* as quality targets. A *content item* is usually identified by an *URI* (*Unified Resource Identifier*). Users give (*explicit*) *feedback* (so-called explicit measurements) about content items, with respect to a set of *quality criteria*. For analysis, we consider time, i.e. the *time point* of each feedback given is relevant here. During analysis, so-called *quality evaluations* are generated for each quality target, as a function of time.

Note, we require users to be quality targets because the semantics of feedback may change depending on user qualities. Consider e.g. the positive feedback of an expert versus the positive feedback of a novice. Clearly, the latter should be given less weight in the computation of the quality of the feedback target.

The quality service layer must support the following functionalities:

1. *Identification and Registration of quality targets*, previously known to the content service implicitly.
2. *Recording of feedback* for particular quality targets.
3. *Provision of quality evaluations* for particular quality targets.
4. *Support of user interfaces*, for seamless integration of content services and quality services in the application layer.

3.2 Architecture

The general architecture for the realization of the quality service layer on top of legacy content is shown in Fig. 2. Specific deployments may vary depending on the services provided by the *original content service* and also depending on other parameters of the actual implementation context.

Fig. 2. The General Architecture (Component Diagram)

Below, each component depicted in Fig. 2 is described in more detail.

Quality Service Overlay: The Quality Service Overlay provides comprehensive access to the combined functionality of quality and content (quality-enabled content).

Quality Navigation: This component facilitates the navigation within quality-enabled contents. It acts as a comprehensive facade for both content and quality presentation. It also collects implicit feedback.

Feedback Functionality: This component facilitates the actual collection of both explicit and implicit feedback.

Quality Target Identification and Registration: These components encapsulate a crucial base functionality for the realization of the quality service layer. The Quality Target Identification component identifies the quality targets from the original content (through the Content Access component), then they are registered to the quality service by the Quality Target Registration component.

Content Access: The Content Access component serves as an interface to the original content, i.e. to the individual Web pages the user sees.

Quality Service: The Quality Service component provides the novel functionality of having quality associated with content, and associated functionalities.

Feedback Collection: This component encapsulates actual collection of the feedback which has been collected in explicit or implicit manners.

Evaluation: The evaluation component facilitates the evaluation of the quality-enabled content, based on explicit user feedback and automatically collected implicit feedback.

Note, both Quality Navigation and Feedback Functionality provide user-interfacing functionalities to the application layer, i.e. they can be used by the user interface of the application layer.

3.3 Quality Criteria, Feedback and Quality Evaluations

Ideally, feedback about quality targets (both explicit user feedback and automatically collected implicit feedback) should be collected with respect to as many quality criteria as possible. Several information quality frameworks have elaborated on the definition and categorization of information quality criteria or information quality dimensions, cf. e.g. Wang and Strong [14], Stvilia et al. [12], and Price and Shanks [9].

For our Moodle case study, cf. Section 4.2, we carefully selected a small set of information quality criteria as shown in Table 1. We tried to choose independent criteria that span a wide range of information quality aspects, while at the same time limiting the total number of criteria and their complexity in order to facilitate an easier judgement for the average user.

Table 1. Information Quality Criteria for Prototype Evaluation

Scale Type	Criteria Name	Values
Nominal	Content Type	examples, tutorial, reference, questions, other
Ordinal	Content Suitability	beginner, intermediate, expert
Interval	Overall Quality Rating	useless, weak, ok, good, excellent
	Currency	out-of-date, partly current, mostly current, current
	Correctness	wrong, partly correct, mostly correct, completely correct
	Ease of Understanding	impossible, hard, reasonable, easy
	Coverage	minimal, ok, good, complete

In order to support aggregation of feedback values from multiple users into a single value, sensitivity towards different scaling types and different semantics is needed. Obviously, a quality target is not better or worse just because its content type is *tutorial*, and not *example*. On the other hand, assigning 0 for out-of-date, 0.3 for partly current, 0.7 for mostly current, and 1.0 for current, and averaging them would be perfectly acceptable. *Content Type*, and *Currency* represent nominal and interval scales, respectively. For a detailed discussion of scale types, see the Handbook of Experimental Psychology [3, p. 16].

4 Validation

We have chosen to show the usefulness and technical soundness of our proposal by (a) providing scenarios to underline the significance of our proposal for education (scenario *Active Classroom*), commerce (scenario *Informed Customer*), and general interest (scenario *Advanced Search*), and (b) refining our proposal through the process of implementing the technical components of a quality service for the course management system of our school.

4.1 Scenarios

Active Classroom. In our perception of an active classroom, concepts to be learned are perceived by the instructors and the students in an interactive process, that is guided by the instructor, but requires the active participation of the students. In contemporary teaching, the perception of concepts would be supported by a Web-centric authoring tool, that allows all participants to edit and modify the explicit representation of concepts. In this scenario, the quality service layer would support the assessment and visualization of both the explicated concepts and the student contributors alike.

In particular,

- Student *Qualities* such as Expertise should be defined in addition to Content *Qualities* such as Accuracy.
- The *Feedback* should be collected as explicit feedback from instructors and students alike.
- The *Processing* of the qualities is targeted towards learning, i.e. it should reflect the student performance and it should be geared towards the perception of content quality as a realization of student progress.

Informed Customer. In the past, advertisements were needed in order to bring products and services to the market that were either unknown, had a small number of potential clients, or hidden qualities. With the help of a quality-enabled internet, product reviews can be evaluated by users and the resulting quality information can help to promote products. In order for this to work, the quality service layer should assess the quality of product reviews and products independently of the interest groups that would like to promote a particular product.

In particular,

- Customer *Qualities* may be used to distinguish different types of customers.
- The *Feedback* should explicitly contain reasons for negative assessments, e.g. price vs. product quality vs. ecological concerns.
- The *Processing* of product and report quality is targeted towards particular customer types, i.e. there might be more than one assessment per target item depending on the type of the customer asking for it.

Advanced Search. Instead of searching for a keyword only, advanced search in a internet of billions of pages of content could take the total quality of a content into account. The quality service layer, in order to support this scenario, needs to have a default quality value for those contents that do not yet receive their quality assessment through sufficient implicit and explicit feedback.

In particular,

- Similar to the active classroom, the expertise of the user is an important *Quality* to judge his authority on certain pages, i.e. domain-specific expertise should be assessed while contents should be classified according to their domain.
- For search purposes, *Feedback* is mostly collected implicitly, but even minimalistic explicit feedback will prove to be extraordinarily useful.
- For search, feedback should be *processed* conservatively, so that most of the contents receive the majority of their quality measure from Google Page Rank initially, with slight changes now and then towards quality-enabled search.

4.2 Moodle Case Study

As an implementation of the technical aspects of the Quality Service Layer, we developed what we called an Information Quality (InfoQual) Module for the native Wiki¹ in the Moodle² installation used at our school as a Course Management System. This information quality module can be used to enrich any number of instances of the Wiki module with a quality service. As Moodle is used by most of the courses in our university, we have the option to experiment with novel quality services in many different contexts, including Computer Engineering, Philosophy, English, Political Science, etc.

Implementation. The following functionalities have been implemented for the InfoQual module:

- quality targets are extracted and subsequently stored (identification/ registration),
- the feedback is recorded (with timestamp),
- the quality evaluations are freely configurable through a function editor, and
- the quality overlay interface is realized by AJAX.

For enabling the user interface components related to the quality layer, triggers are injected into Moodle's Page View, which fire if the mouse hovers over a URI or if the mouse is clicked on a link. This injection is illustrated in Fig. 3. The user contacts the *moodleInjection* instead of the original content. The user interface realization of the Quality Service Layer is orchestrated by it.

First, the original content is accessed in the legacy way. Then, the quality service is accessed, possibly by providing additional information extracted from the original content. Finally, the quality-enriched view is returned to the user. A similar sequence is used to identify quality targets in the original content and register them with the quality service.

¹ A Wiki is a Web-based hypertext, that allows for easy and collaborative editing by many Web-users.

² Moodle (cf. <http://moodle.org>) is a free and open-source e-learning software platform.

Fig. 3. Extension/Modification of the Moodle Wiki Module

Initial Classroom Usage. To test our prototype implementation, we chose the first-year Java-based CS1/2 (Introduction to Algorithms & Programming) courses. Java’s introduction, 15 years ago, coincided with, and was instrumental in the development of, the World Wide Web, so it is not surprising to find that a substantial number of tutorials, examples, etc. can be found online. Of course, not all such resources are still relevant and those that are vary considerably in their usefulness for freshmen students. Since we wish to encourage students to work together to build a community resource, these courses seemed to offer an ideal testbed. Unfortunately, the prototype was only completed towards the end of the semester and still had a rather crude user-interface, making it somewhat difficult for students to use. Even so, the resulting InfoQual Wiki (named CS_Gems) had 93 pages used to organize 292 links to external resources (Web pages), and 453 students made 1638 evaluations (an average of 4.25 per resource). While time didn’t allow a proper test of evaluation, initial comments were positive, though it did require some incentives (in the form of grades) to get all but the really enthusiastic students to contribute!

5 Discussion

This paper proposed a quality service layer for the collection and presentation of information quality measures on the Internet. A prototype based on the Moodle Wiki module was implemented and used by students to collaboratively collect, organize and evaluate Java learning resources from around the Web. Although the basic functionality is similar to many existing systems, the aim of the presentation of the quality function as an architectural layer is to support the generalization and standardization of quality assessment and quality enrichment for the World Wide Web.

References

1. Abiteboul, S., Preda, M., Cobena, G.: Adaptive on-line page importance computation. In: WWW 2003: Proceedings of the 12th International Conference on World Wide Web, pp. 280–290. ACM Press, New York (2003), <http://portal.acm.org/citation.cfm?id=775192>

2. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* 17(6), 734–749 (2005)
3. Atkinson, R.C., Herrnstein, R.J., Lindzey, G., Luce, R.D.: *Stevens' Handbook of Experimental Psychology*, 2nd edn. Wiley, New York (1988),
<http://nla.gov.au/nla.cat-vn1061642>
4. Averbakh, A., Krause, D., Skoutas, D.: Exploiting User Feedback to Improve Semantic Web Service Discovery (8th International Semantic Web Conference, Chantilly, VA, OCT 25–29, 2009). In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) *ISWC 2009. LNCS*, vol. 5823, pp. 33–48. Springer, Heidelberg (2009)
5. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems* 30(1-7), 107–117 (1998)
6. Golbeck, J., Hendler, J.: Inferring binary trust relationships in web-based social networks. *ACM Trans. Inter. Tech.* 6(4), 497–529 (2006),
<http://portal.acm.org/citation.cfm?id=1183463.1183470>
7. Lykourantzou, I., Papadaki, K., Vergados, D.J., Polemi, D., Loumos, V.: CorpWiki: A self-regulating wiki to promote corporate collective intelligence through expert peer matching. *Information Sciences* 180(1, Sp. Iss. SI), 18–38 (2010)
8. Miao, Q., Li, Q., Dai, R.: A unified framework for opinion retrieval. In: *Web Intelligence*, pp. 739–742. IEEE, Los Alamitos (2008)
9. Price, R., Shanks, G.: A semiotic information quality framework: development and comparative analysis. *Journal of Information Technology* 20(2), 88–102 (2005)
10. Sabater, J., Sierra, C.: Review on computational trust and reputation models. *Artificial Intelligence Review* 24(1), 33–60 (2005),
<http://portal.acm.org/citation.cfm?id=1057849.1057866>
11. Schaal, M.: A Bayesian Approach for Small Information Trust Updates. In: *Proceedings of IeCCS 2006* (2006)
12. Stvilia, B., Gasser, L., Twidale, M.B., Smith, L.C.: A framework for information quality assessment. *JASIST* 58(12), 1720–1733 (2007)
13. Wang, K., Lin, C.L., Chen, C.D., Yang, S.C.: The Adoption Of Wikipedia: A Community-And Information Quality-Based View. In: Huang, W., Teo, H.H. (eds.) *12TH Pacific Asia Conference on Information Systems (PACIS 2008)*, Suzhou, Peoples R China, July 03-07, pp. 248–259 (2008)
14. Wang, R.Y., Strong, D.M.: Beyond accuracy: what data quality means to data consumers. *J. Manage. Inf. Syst.* 12(4), 5–33 (1996)