GRAPE: An Expert Review Assignment Component for Scientific Conference Management Systems

Nicola Di Mauro, Teresa M.A. Basile, and Stefano Ferilli

Dipartimento di Informatica, University of Bari, Italy
{ndm, basile, ferilli}@di.uniba.it

Abstract. This paper describes GRAPE, an expert component for a scientific Conference Management System (CMS), to automatically assign reviewers to papers, one of the most difficult processes of conference management. In the current practice, this is typically done by a manual and time-consuming procedure, with a risk of bad quality results due to the many aspects and parameters to be taken into account, and on their interrelationships and (often contrasting) requirements. The proposed rule-based system was evaluated on real conference datasets obtaining good results when compared to the handmade ones, both in terms of quality of the assignments, and of reduction in execution time.

1 Introduction

The organization of scientific conferences often requires the use of a web-based management system (such as BYU [6], CyberChair [2], ConfMan [1], Microsoft CMT [3], and OpenConf [4])[1] to make some tasks a little easier to carry out, such as the job of reviewing papers. Some features typically provided by these packages are: submission of abstracts and papers by Authors; submission of reviews by the *Program Committee Members* (PCM); download of papers by *Program Committee* (PC); handling of reviewers preferences and bidding; web-based assignment of papers to PCMs for review; review progress tracking; web-based PC meeting; notification of acceptance/rejection; sending e-mails for notifications. When using these systems, the hardest and most time-consuming task is the process of assigning reviewers to submitted papers. Usually, this task is manually carried out by the *Program Committee Chair* (PCC) of the conference, that, generally, selects 3 or 4 reviewers *per* paper. Due to the many constraints to be fulfilled, such a manual task is very tedious and difficult, and sometimes does not result in the best solution. It can be the case of 300 submitted papers to be assigned to 30-40 reviewers, where some reviewers cannot revise specific papers because of conflict of interests, or should not revise papers about some conference topics due to their little expertise in that field; additionally, through the

[1] A list of other software, often developed *ad hoc* for specific events, can be found at http://www.acm.org/sigs/sgb/summary.html

M. Ali and F. Esposito (Eds.): IEA/AIE 2005, LNAI 3533, pp. 789–798, 2005.

bidding process reviewers generally express their preference in reviewing specific papers, and should be ensured some level of satisfaction in this respect. The more papers to be reviewed and constraints to be fulfilled, the more vain the hope to obtain a good solution is. Unfortunately, currently available software provides little support for automatic review assignment. Sometimes, they just support reviewers in selecting papers they wish to review, giving the PCC the possibility to use these preferences.

This paper describes GRAPE (Global Review Assignment Processing Engine), an expert component developed to be embedded in scientific Conference Management Systems (CMS). GRAPE, a successful real-life application, fully implemented and operational, performs a management activity by automatically assigning reviewers to papers submitted to a conference, additionally assessing the quality of the results of this activity in terms of profitability and efficiency. This system will become part of a web-based CMS, currently at prototype stage, whose main goal is to provide an easy-to-manage software package that features the traditional conference management functionality (e.g., paper submission, reviewer assignment, discussion on conflicting reviews, selection of papers, mailing to all actors, etc.) and addresses the weaknesses of other systems (such as automatic support for reviewers assignment, conference session management, etc.).

This paper is organized as follows. Section 2 introduces the main features of the CMS prototype in which GRAPE is embedded. In Section 3 the Reviewers Assignment Problem and its context are introduced, and some known systems that tackle it are presented. Section 4 describes GRAPE, whose experimental evaluation is discussed in Section 5. Finally, Section 6 will conclude the paper, with remarks and future work proposals.

2 The Conference Management System

Generally speaking, a CMS, among other basic services, can be seen as a system collecting documents (submitted) in electronic form, in PostScript (PS) or Portable Document Format (PDF), in a digital repository. The main characteristic of our CMS lies in its ability to understand the semantics of the document components and content. Intelligent techniques are exploited for the extraction of significant text, to be used for later categorization and information retrieval purposes. A preliminary Layout Analysis step identifies the blocks that make up a document and detects relations among them, resulting in the so-called layout structure. The next document processing step concerns the association of the proper logical role to each such component, resulting in so-called logical structure. This can enable a multiplicity of applications, including hierarchical browsing, structural hyperlinking, logical component-based retrieval and style translation. Our layout analysis process embedded in the CMS, sketched in Figure 1, takes as input a PDF/PS document and produces the initial document's XML basic representation, that describes it as a set of pages made up of basic blocks. Such a representation is then exploited by an algorithm, that collects

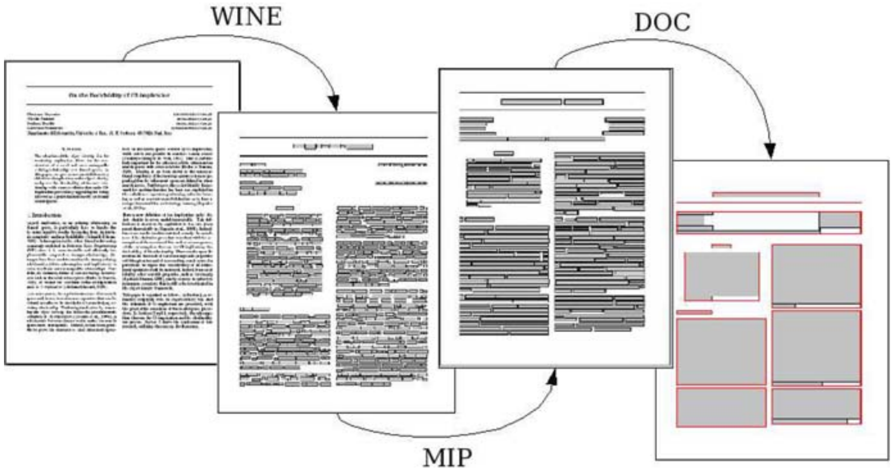

Fig. 1. Document Layout Analysis System

semantically related basic blocks into groups by identifying frames that surround them based on whitespace and background structure analysis.

When an Author connects to the Internet and opens the conference submission page, the received paper may undergo the following processing steps. The layout analysis algorithm may be applied, in order to single out its layout components, and then to classify it according to a description for the acceptable submission layout standards (e.g., full paper, poster, demo). A further step locates and labels the layout components of interest for that class (e.g., title, author, abstract and references in a full paper). The text that makes up each of such components may be read, stored and used to automatically file the submission record. The text contained in the title, abstract and bibliographic references, can be exploited to extract the paper topics, since we assume they concentrate the subject and research field the paper is concerned with.

3 The Conference Review Process

The review process on the contributions submitted by authors to a conference starts after the deadline for the paper submission phase. When the submission phase ends, suitable *PCM* are selected, which will act as *reviewers*, in order to evaluate the submitted papers. Hence, the PCC sends the collected submissions with review forms to individual reviewers. The review form consists of a set of questions to assess the quality of the paper, that the Reviewers must fill in and return it to the PCC. Each submission is typically examined and evaluated by 2 or 3 reviewers. Generally, the review process ends with the *Program Committee meeting*, where the papers are discussed on the basis of collected review forms, in order to their acceptance or rejection for presentation at the conference. After this meeting, anonymous extracts of the review forms (reviewer's comments) are typically sent back to all the authors, so that they can improve their paper, regardless of whether they were accepted or not. Finally, the authors of accepted papers may submit a new version of their paper, in the so-called *camera-ready* format, to the PCC, who will send them, together with the preface and the table of contents of the book, to the publisher in order to have the proceedings printed.

3.1 The Reviewers Selection

The process of identifying the right reviewers for each paper represents an hard task. In [7] O. Nierstrasz presented a small *pattern language*, that captures successful practice in several conference review processes. In this work we follow the patterns *ExpertsReviewPapers* and *ChampionsReviewPapers*, indicating that papers should be matched, and assigned for evaluation, to reviewers who are competent in the specific particular paper topics (*ExpertsReviewPapers*), and to reviewers who declared to be willing to review those papers in the bidding phase (*ChampionsReviewPapers*). As to the former pattern, the PCC can set up, before the submission phase of the conference starts, a list of research topics of interest for the conference. In order to get a match, generally, at first all reviewers are asked to specify which of the conference topics correspond to their main areas of expertise. Then, during the paper submission process, authors are asked to explicitly state which conference topics apply to their submissions. Such an information provides a first guideline for matching reviewers to papers. As to the latter pattern, as reported in [7], by distributing a list of titles, authors and abstracts to all reviewers, they may perform the so-called *bidding*, i.e. they may indicate which papers (i) they would like to review, (ii) they feel competent to review, and (iii) they do not want to review (either because they do not feel competent, or because they have a conflict of interest).

Finally, further information to match papers and Reviewers can be deduced from the papers. For example, related work by some reviewer explicitly mentioned in the paper may represent an indication of the appropriateness of that reviewer for that paper; conversely, if the reviewer is a co-author or a colleague of the paper authors, then a conflict of interest can be figured out.

Usually, the bidding preferences approach is preferred over the topics matching one. We give more value to the latter since the topics selected by a reviewer should refer to his background expertise, while specific preferences about papers could be due to matter of taste or some vague questions (e.g., the reviewer would like to revise a paper out of curiosity; the abstract that he has read is not very precise or misleading). We believe that if a paper preferred by a reviewer does not match his topics of expertise, this should be considered as an alarm bell.

3.2 Paper Distribution Systems

Most of the existent CMS, such as CMT [3] and CyberChair [2], provide tools for web-based paper submission and for review management. Both CMT and CyberChair have assignment functionalities. Specifically, CMT uses a greedy algorithm to assign a paper to the reviewers who gave the higher preference, but limiting the number of papers assigned to a reviewer by means of a threshold. When the system cannot find a reviewer for a paper, a matching of both the reviewers and paper topics is used. If this fails the result is unpredictable.

CyberChair [9], after the paper submission deadline, generates a paper distribution proposal for the PCC exploiting graph theory. This is done by combining the reviewer's expertise and willingness to review papers on certain topics with the preferences indicated by the reviewers when bidding for papers. The reviewer's expertise is obtained by asking the reviewers their expertise on the conference topics along with three levels: 1) expert on the topic, 2) not expert, but knowledgable in the subject area, 3) an informed outsider. CyberChair collects the bids, expertise levels and willingness to review papers on certain topics and the conflicts of interest of the reviewers, and it is tuned to assign as much as possible papers to reviewers based on their preferences. Each paper is assigned to the k reviewers with the least number of papers assigned so far, by using a list of the reviewers who indicated a *high* preference for the paper sorted according to the number of papers they have already been assigned so far. In case there are less than k reviewers, this process is repeated with the list of reviewers who indicated a *low* preference for the paper. In case there are not enough reviewers, the paper is assigned to the reviewers with the highest overall expertise.

An *ad-hoc* system is presented in [8], where the reviewers' assignment problem is compared to the more general problem of recommendation of items to users in web-based systems, and proposed a recommendation method based on collaborative filtering [8]. Such a method allows to compute a predicted rating for each pair (reviewer, paper), using a multi-step process which improves continuously the confidence level of ratings. In particular, each step consists of the following operations: (a) for each user, a sample of papers whose rating is expected to lead to the best confidence level improvement is selected, (b) each user is requested to rate the papers from its sample and (c) a collaborative filtering algorithm is performed to obtain a new set of predicted ratings based on the users ratings made so far. Step (c) results in a new level of confidence. The basic assumption is that each user provides a rating for each paper: Reviewers are required to rate the submitted papers based on title, abstract and authors information. These ratings are used by the algorithm to obtain the best possible assignment. This system relies on a variant of known techniques for optimal weighted matching in bipartite graphs [5], and delivers the best possible assignment. However, in practice, the number of papers is often large and it is difficult to ask for a comprehensive rating. Users rate only a small subset of the papers, and the rating table is sparse, with many unknown rating values. To overcome the problem in order to use the automatic assignment algorithm, they must then predict the missing rating values.

4 The GRAPE System

GRAPE (Global Review Assignment Processing Engine) is an expert system, written in CLIPS, for solving the reviewers assignment problem, that takes advantage from both the papers content (topics) and the reviewers preferences (biddings). It could be used exploiting the papers topics only, or both the paper topics and the reviewers' biddings. Its fundamental assumption is to prefer the topics matching approach over the reviewers' biddings one, based on the idea that they give assignments more reliability. Then, reviewers' preferences are used to tune paper assignments. Moreover, reviewers' preferences are useful because of the unpredictability of the distribution of reviewers and papers over the list of topics, which causes situations in which some papers have a lot of experts willing to review them, while some others simply do not have enough reviewers.

Let $P = \{p_1, \ldots, p_n\}$ denote the set of n papers submitted to the conference C, regarding t topics (*conference topics*, T_C), and $R = \{r_1, \ldots, r_m\}$ the set of m reviewers. The goal is to assign the papers to reviewers, such that the following *basic constraints* are fulfilled: 1) each paper is assigned to exactly k reviewers (usually, k is set to be equal to 3 or 4); 2) each reviewer should have roughly the same number of papers to review (the mean number of reviewers *per* reviewer is equal to nk/m); 3) papers should be reviewed by domain experts; and, 4) reviewers should revise articles based on their expertise and preferences. As regards constraint 2, GRAPE can take as input additional constraints MaxReviewsPerReviewer(r,h), indicating that the reviewer r can reviews at most h paper, that must be taken into account for calculating the mean number of reviews per reviewer.

We defined two measures to guide the system during the search of the best solutions: the *reviewer's gratification* and the *article's coverage*. The former represents the gratification degree g_{r_j} of a reviewer r_j calculated on his assigned papers. It is based on: a) the *confidence degree* between the reviewer r_j and the assigned articles: the confidence degree between a paper p_i, with topics T_{p_i} and the reviewer r_j, with expertise topics T_{r_j}, is equal to number of topics in common $T = T_{p_i} \cup T_{r_j}$; and, b) the number of assigned papers that the reviewer chose to revise (discussed in more details in Section 4.2). The article's coverage represents the coverage degree of an article after the assignments. It is based on: a) the *confidence degree* between the article and the assigned reviewers (the same as for Reviewer's gratification); and, b) the *expertise degree* of the assigned reviewers, represented by the number of topics. The expertise level of a reviewer r_j is equal to T_{r_j}/T_C. GRAPE tries to maximize both the reviewer gratification and the article coverage degree during the assignment process, in order to fulfill the third and fourth basic constraints. To reach this goal a fundamental prerequisite is that each reviewer must provide at least one topic of preference, otherwise the article coverage degree will be always null.

The two main inputs to the system are the set P of the submitted papers and the set R of the candidate reviewers. Each paper $p_i \in P$ is described by its title, author(s), affiliation(s) of the author(s) and topics T_{p_i}. On the other hand, each reviewer $r_j \in R$ is described by his name, affiliation and topics of interest T_{r_j}. Furthermore, the system can take as input a set of constraints CS indicating (i) the possibly specified maximum number of reviews *per* Reviewer (MaxReviewsPerReviewer(reviewer, h)), (ii) the papers that *must* be reviewed by a reviewer (MustReview(reviewer, paper)) indicated by the PCC. It can be also provided with a set of *conflicts CO* indicating which reviewers that cannot revise specific papers (CannotReview(reviewer, paper)) under suggestion of the PCC. Furthermore, the set of conflicts CO is enriched by GRAPE by deducting additional conflicts between papers and reviewers. Specifically, a conflict is assumed to exist between a paper p_i and a reviewer r_j if r_j is the (co-)author of p_i, or the affiliation of r_j is among the affiliation(s) reported in p_i.

Definition 1. *We say that* a reviewer r_j can revise a paper p_i with degree

$$\begin{cases} h \geq 1 & \textit{if the confidence degree between } r_j \textit{ and } p_i \textit{ is equal to } h \\ 0 \leq h < 1 & \textit{if the expertise degree of } r_j \textit{ is equal to } h \end{cases}$$

Definition 2. *Given a paper p_i, the number of* candidate reviewers *of p_i is the number of reviewers that can revise the paper with degree $k \geq 1$*

4.1 The Assignment Process

The assignment process is carried out into two phases. In the former, the system progressively assigns reviewers to papers with the lowest number of candidate reviewers (first those with only 1 candidate reviewer, then those with 2 candidate reviewers, and so on). This assures, for example, to assign a reviewer to papers with only one candidate reviewer. At the same time, the system *prefers* assigning papers to reviewers with few assignments. In this way, it avoids to have reviewers with zero or few assigned papers. Hence, this phase can be viewed as a search for reviews assignments by keeping low the average number of reviewers *per* reviewer and maximizing the coverage degree of the papers. In the latter phase, the remaining assignments are chosen by considering first the confidence levels and then the expertise level of the reviewers. In particular, given a paper p_i which has not been assigned k reviewers yet, GRAPE tries to assign it a reviewer r_j with a high confidence level between r_j and p_i. In case it is not possible, it assigns a reviewer with a high level of expertise.

4.2 The Bidding Process

The assignments resulting from the base process are presented to each reviewer, that receives the list A of the h assigned papers, followed by the list A' of the remaining ones (both, A and A' are sorted using the article's coverage degree). The papers are presented to the reviewer as virtually bid: the first $h/2$ papers of the list A are tagged *high*, and the following h papers are tagged *medium* (all the others are tagged *low*). Now, the reviewer can actually bid the papers by changing their tag: he can bid at most $h/2$ papers as *high* (*he would like to review*) and h as *medium* (*he feels competent to review*). Furthermore, he can bid $h/2$ papers as *no* (*he does not want to review*). All the others are assumed to be bid as *low* (*he does not feel competent*). Only papers actually bid by reviewers generate a preference constraint, of the form Bid(paper, level).

When all the reviewers have bid their papers, GRAPE searches for a new solution that takes into account these biddings. In particular, it tries to change previous assignments in order to maximize both article's coverage and reviewer's gratification. By taking the article's coverage high, GRAPE tries to assign the same number of papers bid with the same class to each reviewer. Then, the solution is presented to the reviewers as the final one.

It is important to say that, if the PCC does not like the solution, he can change some assignments to force the system to give another possible assignment configuration fulfilling these new preference constraints. In particular, he may: (i) assign a reviewer to a different paper; (ii) assign a paper to a different reviewer; (iii) remove a paper assignment; or, (iv) remove a reviewer assignment.

The main advantage of GRAPE relies in the fact that it is a rule-based system. Hence, it is very easy to add new rules in order to change/improve its behavior, and it is possible to describe background knowledge, such as further constraints or conflicts, in a natural way. For example, one can insert a rule that expresses the preference to assign a reviewer to the articles in which he is cited.

5 Evaluation

The system was evaluated on real-world datasets built by using data from a previous European conference and from this International conference. In order to have an insight on the quality of the results, in the following we present some interesting characteristics of the assignments suggested by GRAPE.

5.1 European Conference Experiment

This experiment consisted in a set of 383 papers to be distributed among 72 Reviewers, with $k = 3$ reviews *per* paper. The system was able to correctly assign 3 reviewers to each paper in 152 seconds. Obtaining a manual solution took about 10 hours of manual work from the 4 Program Chairs of that conference.

Each reviewer was assigned 14.93 papers on average (min 8, max 16) by topic (when there was confidence degree greater than 1 between the reviewer and the paper), and only 1.03 papers on average (min 0, max 8) by expertise degree (which is a very encouraging result). Table 1 reports the complete distribution of reviewers' assignments. The first row shows the number of assignments by type (Topics-Expertise). Noticeably, GRAPE made many good assignments: in particular, it assigned to 55 reviewers all 16 papers by topics (first row). The other rows refer to the topics of expertise of reviewers: the last two rows indicate that the system assigned an high number of papers by expertise to reviewers that had few topics.

The reviewers with highest gratification degree were r_{22}, r_{57}, and r_{20}. Indeed, they are the three reviewers that chose a lot of topics (r_{22} selected 11 topics, r_{57} selected 14, and r_{20} selected 17). On the other hand, the reviewers with the lowest gratification degree were r_{10} that selected few (and very rare among the papers) topics, and r_{56} that selected only two topics. As regards the papers, the best assigned papers with a high coverage degree were the p_{239} (concerning topics 1, 4, 15, 39 and 42), p_{231} (on topics 1, 2, 15, 30, 34 and 5), p_{303} (topics 1, 9, 11, 32 and 36), and p_{346} (topics 4, 15, 39 and 42). Table 2 reports the

Table 1. Reviewers' Assignments Distribution

Assignment	16-0	15-1	14-2	13-3	11-5	10- 5	8-8	8-6
#	55	3	2	3	3	1	4	1
Mean	5,73	2,67	2,5	2,67	1,67	1	1,5	1
Min	2	2	2	2	1	1	1	1
Max	18	3	3	4	2	1	2	1

Table 2. Reviewers per topic

Topic	1	2	4	9	11	15	30	32	34	36	39	42
Reviewers	17	14	10	5	5	12	11	5	3	7	20	17

Table 3. IEA/AIE 2005 Topics Distribution

ID	Topic	#r	#a	ID	Topic	#r	#a
1	Adaptive Control	3	11	18	Intelligent Interfaces	14	31
2	Applications to Design	3	19	19	Intelligent Systems in Educ.	11	12
3	Applications to Manufacturing	2	12	20	Internet Applications	12	24
4	Autonomous Agents	18	28	21	KBS Methodology	7	13
5	BioInformatics	9	8	22	Knowledge Management	16	30
6	Case-based Reasoning	9	4	23	Knowledge Processing	11	25
7	Computer Vision	3	20	24	Machine Learning	17	43
8	Constraint Satisfaction	6	9	25	Model-based Reasoning	6	11
9	Data Mining & Knowledge Disc.	24	44	26	Natural Language Process.	5	15
10	Decision Support	9	58	27	Neural Networks	11	29
11	Distributed Problem Solving	6	6	28	Planning and Scheduling	7	27
12	Expert Systems	15	28	29	Reasoning Under Uncertain.	4	20
13	Fuzzy Logic	4	13	30	Spatial Reasoning	7	9
14	Genetic Algorithms	5	29	31	Speech Recognition	2	8
15	Genetic Programming	2	6	32	System Integration	3	14
16	Heuristic Search	3	16	33	Systems for Real Life App.	10	41
17	Human-Robot Interaction	3	14	34	Temporal Reasoning	11	10

number of reviewers experienced in some topics of the conference. As one can see, there are lots of reviewers experienced with the topics appearing in papers with a high coverage degree. Papers with a low coverage degree were p_{15} (3 rare topics covered), p_{42} (2 rare topics) and p_{373} (0 topics).

5.2 IEA/AIE 2005 Experiment

In this experiment the dataset was built by using data from this conference[2], consisting of a set of 266 papers to be distributed among 60 Reviewers. The conference covered 34 topics as reported in Table 3, where #r represents the number

[2] IEA/AIE 2005 - The 18th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems

of reviewers experienced with the topic, and #a represents the papers regarding the topic. $k = 2$ reviews *per* paper were required. In solving the problem, the system was able to correctly assign 2 reviewers to each paper in 79.89 seconds.

GRAPE was able to assign papers to reviewers by considering the topics only (it never assigned a paper by expertise). In particular, it assigned 10 papers to 38 reviewers, 9 to 4 reviewers, 8 to 6 reviewers, 7 to 1 reviewer, 6 to 9 reviewers, 5 to 1 reviewer, and 2 to 1 reviewer, by considering some MaxReviewsPerReviewer constraints for some reviewers that explicitly requested ro revise few papers. The reviewers with the highest gratification degree, with 10 assigned papers, were r_{24} (that selected 7 topics), r_{32} (that selected 8 topics) and r_{41} (that selected 6 topics). As regards the papers, those assigned with highest coverage degree were p_{24}, p_{31}, p_{47}, p_{67}, p_{70}, p_{78}, p_{81}, p_{177}, p_{181}, p_{198}, p_{242} and p_{260}.

6 Conclusions

We presented the GRAPE expert system, specifically designed to solve the problem of reviewer assignments for scientific conference management. The proposed rule-based system was evaluated on real-world conference datasets obtaining good results when compared to the handmade ones, both in terms of quality and user-satisfaction of the assignments, and for reduction in execution time with respect to that taken by humans to perform the same process.

GRAPE is embedded in a web-based CMS in which we plan to insert some tools able to automatically extract the paper's topics from its title, abstract, and references, and the reviewer's topics by analyzing his previously written paper and web pages. Furthermore, we are planning to insert in our web-based CMS, a sessions manager system similar to GRAPE able to automatically propose sessions for the conference and the presentations for each session.

References

1. The confman software. `http://www.zakongroup.com/technology/openconf.shtml`.
2. The cyberchair software. `http://www.cyberchair.org`.
3. The microsoft conference management toolkit. `http://msrcmt.research.microsoft.com/cmt/`.
4. The openconf conference management system `http://www.zakongroup.com/technology/openconf.shtml`.
5. H.W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97, 1955.
6. Stephen W. Liddle. The byu paper review system. `http://blondie.cs.byu.edu/PaperReview/`.
7. O. Nierstrasz. Identify the champion. In N. Harrison, B. Foote, and H. Rohnert, editors, *Pattern Languages of Programm Design*, volume 4, pages 539–556. 2000.
8. Philippe Rigaux. An iterative rating method: Application to web-based conference management. In *ACM Intl. Conf. on Applied Computing (ACM-SAC'04)*, 2004.
9. Richard van de Stadt. Cyberchair: A web-based groupware application to facilitate the paper reviewing process. Available at www.cyberchair.org, 2001.