The Gugubarra Project: Building and Evaluating User Profiles for Visitors of Web Sites

Natascha Hoebel[1], Sascha Kaufmann[1], Karsten Tolle[1], and Roberto V. Zicari[1]

Abstract—**In this paper we will report the work currently being doing one the Gugubarra project. The project aims at building tools for better management of communities of Web visitors.**

***Index Terms*—User Profiles, Web Communities, User modeling, Clustering methods**

I. Introduction

THE Gugubarra project (Gugubarra is the Aboriginal name for the Kookaburra bird) began in 2004 within the database group (DBIS) at the Computer Science Institute of the Johann Wolfgang Goethe University, with the aim to build tools enhancing management capabilities of communities comprising of registered Web visitors.

In this paper we will report the results of the project so far and outline some open research issues.

The starting point of our project is the assumption that a community of users is registered on a Web site and that for each user a profile is built. A User profile is based on the actions and navigations the user performs on the Web site.

In Gugubarra, we offer various settings that can be used to create and manage user profiles. By using these settings the Web site owner can focus on the aspects he wants to analyze.

The approach we have in Gugubarra is as follows:

i) For each registered Web visitor we create a profile. These user profiles reflect the "inferred" interests of the users related to a set of pre-defined *topics* defined by the owner of the Web site. The profiles go beyond collecting the obvious information the user is willing to give at the time of registration. In Gugubarra, a user profile contains two parts: the *obvious profile*, given directly by the user and a *non obvious profile* (*NOP*), inferred by the user's behavior during his visits on the site. The obvious profile part is known from Ardissono [1] as Identification and Personal data.

ii) A user profile is (re)-calculated dynamically any time an explicit feedback is given by the user and/or a set of events occurred which are related to the user's behavior and to certain "locations" of the Web site.

iii) We cluster Web visitors by clustering similar profiles of interest [8]. Cluster of Web visitors can then be used to analyze patterns of interests in the Web community and to forecast further behavior. Clusters might also provide useful information to support the decision of what kind of new E-services to introduce for the Web community and when to introduce them.

A first research prototype system, called *Gugubarra 1.0* has been implemented in 2004 [9], which allows building and manipulating non-obvious user profiles. It was showcased at the CeBIT Trade Fairs in 2005 and 2006 [3]. Gugubarra 1.0 works as a test application on real data provided by the Web community *ViewZone.org*.

A new prototype system, *Gugubarra 2.0*, is currently being designed [7] which includes a more sophisticated approach to the definition of non-obvious user profiles and allows clustering users by interests [8]. In this paper we will focus mainly on the new features introduced in *Gugubarra 2.0* and refer to [9] for the features implemented in *Gugubarra 1.0*.

The relevant data used in the Gugubarra algorithm(s) are extracted from the various log files generated by the Web server. Based on this data, the user's navigation through the site is captured together with the corresponding topics, and the algorithm generates the non-obvious user profiles. Informally we can say that the NOP shows the path of the user through the site and therefore the footprints he has left there.

II. Building Non Obvious Profiles (NOPs)

We consider a Web site (or a domain) as a collection of Web pages that are linked together within the site. Each page has a specific content.

Manuscript received September 1, 2006.

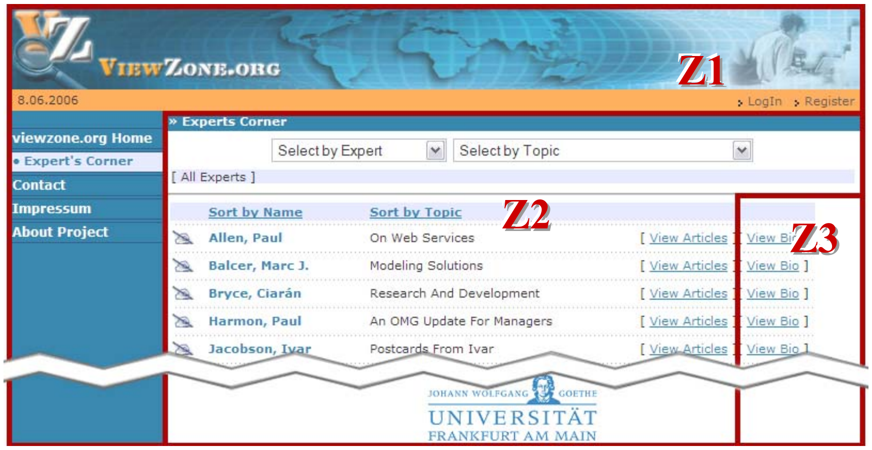
R. V. Zicari, Professor, Database and Information Systems (DBIS), Johann Wolfgang Goethe University, 60486 Frankfurt, Germany, Computer Science Institute (email: zicari@cs.uni-frankfurt.de).

N. Hoebel, Research Assistant, Database and Information Systems (DBIS), Johann Wolfgang Goethe University, 60486 Frankfurt, Germany, Computer Science Institute (email: hoebel@dbis.cs.uni-frankfurt.de).

S. Kaufmann, Research Assistant, Database and Information Systems (DBIS), Johann Wolfgang Goethe University, 60486 Frankfurt, Germany, Computer Science Institute (email: kaufmann@dbis.cs.uni-frankfurt.de).

K. Tolle, PhD, Research Assistant, Database and Information Systems (DBIS), Johann Wolfgang Goethe University, 60486 Frankfurt, Germany, Computer Science Institute (email: tolle@dbis.cs.uni-frankfurt.de).

1-4244-0596-3
/06/$20.00 ©2006 IEEE



Sort by Name	Sort by Topic		
Allen, Paul	On Web Services	[View Articles]	View Bio
Balcer, Marc J.	Modeling Solutions	[View Articles]	View Bio]
Bryce, Ciarán	Research And Development	[View Articles]	View Bio]
Harmon, Paul	An OMG Update For Managers	[View Articles]	View Bio]
Jacobson, Ivar	Postcards From Ivar	[View Articles]	View Bio]

Fig. 1. Page P_1”Expert’s Corner” with three Zones.

In Gugubarra 2.0 the following main concepts are used to create a user’s NOP: ***Zones, Topics, Weights, Actions*** and ***Feedback***.

There are several different parameters in Gugubarra 2.0 that combined determine a specific *strategy* to deploy and manage NOPs. We will present in this paper only one of the several strategies that have been designed in Gugubarra 2.0 [7].

In the rest of the paper we use *ViewZone.org* as an example of a Web community, to illustrate the main concepts of Gugubarra 2.0. ViewZone.org is a non-profit test community for IT professionals interested in various topics in Information Technology. ViewZone has a section called “Experts Corner”, where experts publish articles on several topics related to IT. The download of the articles is free of charge for registered users only.

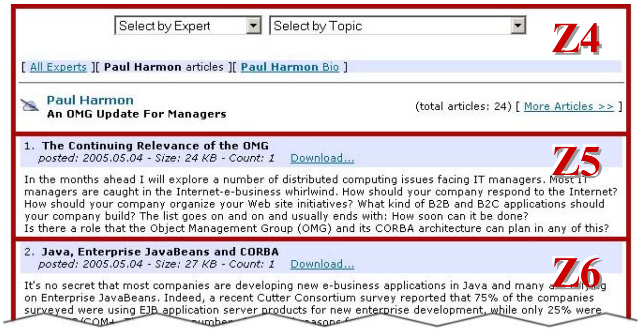
In this paper we will consider two fictitious users, Alice and Bob, who registered to ViewZone.org and will show how their profiles are created and changed through their actions on this Website.

A. Zones and States

A **zone** defines a location in the Web site. It can be a set of pages, a set of parts within a page, a set of parts of several pages, or any combination thereof.

A zone has a **state** associated (ON or OFF). The state ON indicates that the zone is being used to calculate the NOP of the user. A state of a zone can change, e.g. for example a zone with state OFF becomes ON when the user does an action within the zone. For simplicity in our example we will consider all zones to have a state ON.

Let’s consider three pages of ViewZone, P_1, P_2 and P_3. As illustrated in Fig. 1, 2, and 3, we have defined seven zones for the pages.



Paul Harmon
An OMG Update For Managers
(total articles: 24) [More Articles >>]

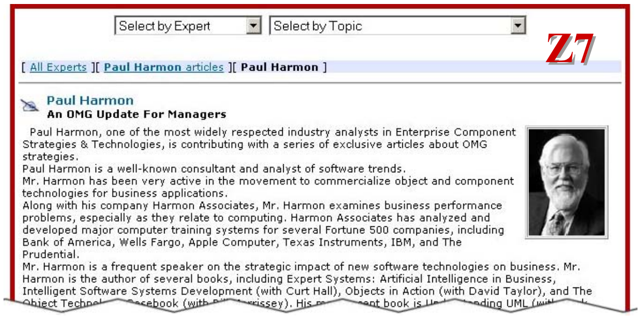
1. The Continuing Relevance of the OMG
posted: 2005.05.04 - Size: 24 KB - Count: 1

In the months ahead I will explore a number of distributed computing issues facing IT managers. Most IT managers are caught in the Internet-e-business whirlwind. How should your company respond to the Internet? How should your company organize your Web site initiatives? What kind of B2B and B2C applications should your company build? The list goes on and on and usually ends with: How soon can it be done?
Is there a role that the Object Management Group (OMG) and its CORBA architecture can plan in any of this?

2. Java, Enterprise JavaBeans and CORBA
posted: 2005.05.04 - Size: 27 KB - Count: 1

It's no secret that most companies are developing new e-business applications in Java and many a[illegible]g on Enterprise JavaBeans. Indeed, a recent Cutter Consortium survey reported that 75% of the companies surveyed were using EJB application server products for new enterprise development, while only 25% were

Fig. 2. Page P_2 ” Paul Harmon articles” with three zones shown.



Paul Harmon
An OMG Update For Managers

Paul Harmon, one of the most widely respected industry analysts in Enterprise Component Strategies & Technologies, is contributing with a series of exclusive articles about OMG strategies.
Paul Harmon is a well-known consultant and analyst of software trends.
Mr. Harmon has been very active in the movement to commercialize object and component technologies for business applications.
Along with his company Harmon Associates, Mr. Harmon examines business performance problems, especially as they relate to computing. Harmon Associates has analyzed and developed major computer training systems for several Fortune 500 companies, including Bank of America, Wells Fargo, Apple Computer, Texas Instruments, IBM, and The Prudential.
Mr. Harmon is a frequent speaker on the strategic impact of new software technologies on business. Mr. Harmon is the author of several books, including Expert Systems: Artificial Intelligence in Business, Intelligent Software Systems Development (with Curt Hall), Objects in Action (with David Taylor), and The

Fig. 3. Page P_3 ” Paul Harmon biography” with one zone shown.

B. Topics and Weights

Topics are related to the content of the Web site and are defined by the owner of the Web site for each *zone*.

Topics are associated to zones and do have a weight associated normally determined by the zone’s content.

A **weight** indicates the relative importance of the topic in respect to a scale from 0 (not relevant) to 1 (very relevant) in the zone.

In the example we consider the following topics for the ViewZone.org Web site:

- The “titles of the series of articles”,
- The “authors’ names”,
- The name of the Portal “ViewZone.org”,
- “OMG” and
- “JAVA”.

Let’s consider in the example, page P_1 (Expert Corner).
We have 3 zones for this page and the following associations:

- For zone Z_1 we associate as topic the Name of the Portal,
- For zone Z_2 the topics are the titles of the articles,
- For zone Z_3 the topics are the names of the Experts.

We define the following weighted topic lists defined for zones Z_1, Z_2 and Z_3 of page P_1 (Fig. 1):

- *Tp(Z_1)= {(’ViewZone.org’, 0.2)}*
- *Tp(Z_2)={(’On Web Services’, 0.5), (’Modeling Solutions’, 0.5), (’Research and Development’, 0.5), (’An OMG Update For Managers’, 0.5), ...}*
- *Tp(Z_3)={(’Allen, Paul’, 0.5), (’Balzer, Marc J.’, 0.5), (’Bryce, Ciarán’, 0.5), (’Harmon, Paul’, 0.5), ... }.*

For page P_2 (Fig. 2) we have a similar list of topics for zones Z_4, Z_5 and Z_6:

- *Tp(Z_4)= {(’Paul Harmon’, 0.8) (’An OMG Update For Managers’, 0.6)}*
- *Tp(Z_5)= {(’OMG’, 0.7), (’Paul Harmon’, 0.7), (’The Continuing Relevance of the OMG’, 0.5)}*
- *Tp(Z_6)= {(’OMG’, 0.5), (’JAVA’, 0.7), (’Paul Harmon’, 0.7), (’Java, Enterprise JavaBeans and CORBA’, 0.5)}.*

And for page P_3:

- *Tp(Z_7)= {(’Paul Harmon’, 0.8)}.*

C. Actions

Actions are also defined by the owner of the Web site, and are global, that is they are applicable to any zone defined for the Web site. Each action has a weight associated which indicates the importance given to such action by the owner of the Web site, ranging from *0* as minimum up to *n* as max.

In the example we have defined the following actions:

- A_1 = PageRequest, weight aw_1 = 1
- A_2 = Download PDF, weight aw_2 = 8.

Action A_1 is the default action to download a page. A_2 is the action that corresponds to downloading any PDF-file on the Website.

D. Calculating the NOP

In the example, we suppose the user Alice has already registered on ViewZone.org and logged in. We assume the initial NOP for Alice is empty (except for the obvious profile see [9]), because at registration time Alice didn’t specify any specific interest in the topics of the Web site.

To explain how we calculate the NOP, we will show in the example, how the NOP changes during one session, as if it would be calculated after each action of Alice. Normally this calculation would only take place based on rules defined by the Web site owner, e.g. during night when the traffic is low. Only in certain situations it would make sense to directly calculate the NOP, e.g. in case Alice wants to see her current NOP or gives a feedback.

In general the calculation of the level of interest x_i on topic *i* (see formula (1) below) is determined by two parts:

i) *Action Profile (ActP)* : the action a user does in a zone, which is computed in the *Action Profile (ActP)*

and

ii) *Duration Profile (DurP):* the time, a user spends on a page, which is computed in the *Duration Profile (DurP).*

These two factors can be parameterized by the Web site owner in accordance to his needs. This means the owner can increase or reduce the impact of the time a user spends on a page reciprocal to the actions he did.

It is worth mentioning that there are some problems related to the time a user spends, because you will not know if the user really has read the page or might have fallen asleep. Of course this can partially be absorbed by using timeouts. However, by allowing the Web site owner to adjust the parameters to the measured behavior of the community, we expect better results and a higher level of trust the Web site owner gives to the results.

In the overall formula (1) below, the parameters *a* and *b* are used to customize the ratio between *ActP(i)* and *DurP(i)*. In Gugubarra 2.0 this will be implemented in the user interface by a slider.

$$\begin{aligned} x_i &= a * ActP(i) + b * DurP(i), \\ where\ (a+b) &= 1 \end{aligned} \quad (1)$$

Let’s assume Alice’s first action at time t_0 is a PageRequest (A_1) of the page P_1, Experts Corner.

Alice’s NOP remains empty until Alice does perform her first action in a zone. This means that initially for every topic her interest is Zero as a starting point. The time duration is counted after the first page request (P_1).

Suppose after 2 minutes on page P_1 Alice performs an action A_1, in the zone Z_2; she clicks on the link "View Articles" related to author "Paul Harmon". This corresponds to the *PageRequest* (P_2). Her NOP is then computed (at time t_1) with the formula (1).

Let's have a closer look at the two parts of the formula (1), as defined by (2) and (3) below.

To compute *ActP(i)*, we determine zone *q*, where the action occurred and obtain the associated topic weight $v(Tp_i, Z_q)$. We then multiply this value by the sum for all weights for all occurred actions in this zone. Finally we calculate the sum of all zones, where an action occurred and the associated topic list which contains topic *i*, divided by the sum off all occurred action weights.

DurP(i) is computed in a similar way. We consider each visited page P_j, that contains the topic Tp_i and multiply the time the visitor spent on this page by its topic weight $v(Tp_i, P_j)$. We finally sum these values and divide it by the total time, the user spent on the Website.

$$ActP(i) = \frac{\sum_q \left(\sum_t aw_t * v(Tp_i, Z_q) \right)}{\sum_s aw_s} \quad \textbf{(2)}$$

$$DurP(i) = \frac{\sum_j \left(duration(P_j) * v(Tp_i, P_j) \right)}{\sum_k duration(P_k)} \quad \textbf{(3)}$$

Note that in Formula (3) we used the value $v(Tp_i, P_j)$, topics for a page to compute the duration part *DurP(i)*.

In *Gugubarra 2.0* however the owner associates topics and their weights to zones not to pages, see section B (*zone topic lists*). To calculate the NOP therefore we generate a *page topic list* composed by the zone topic lists for each page. For this calculation only zones with state=ON are taken into account. In addition the Web site owner can decide individually for every page how this topic list is generated by using one of the following rules:

- MAX-Rule: For each topic *i*, scan all zones on a page *j* and take the maximum topic weight.
  $x_i = \max(v(Tp_i, Z_q)) \forall Z_q \in P_j$
- MIN-Rule: For each topic *i*, scan all zones on a page *j* and take the minimum topic weight.
  $x_i = \min(v(Tp_i, Z_q)) \forall Z_q \in P_j$
- AVG-Rule: For each topic *i*, scan all zones on a page *j* and take the average topic weight.
  $x_i = avg(v(Tp_i, Z_q)) \forall Z_q \in P_j$

Coming back to the example, Alice first computed NOP at time t_1 is as follows:

NOP('Alice', t₁) = {('ViewZone.org', 0.1)
('On Web Services', 0.5), ('Modeling Solutions', 0.5), ('Research and Development', 0.5), ('An OMG Update For Managers', 0.5), ...
('Allen, Paul', 0.25), ('Balzer, Marc J.', 0.25), ('Bryce, Ciarán', 0.25), ('Harmon, Paul', 0.25), ...
}

Suppose now Alice stays 5 minutes on page P_2. After 5 minutes on page P_2, Alice downloads (action A_2) an article of Paul Harmon about 'Java, Enterprise JavaBeans and CORBA' in zone Z_6.

Note that every PDF is handled like one zone and has its own topic list. For the article 'Java, Enterprise JavaBeans and CORBA' we define zone Z_8 with the following weighted topics:

- *Tp(Z₈)= {('OMG', 0.4), ('JAVA', 0.8), ('Paul Harmon', 1.0), ('* Java, Enterprise JavaBeans and CORBA*', 1.0)}*

After the article download, Alice's NOP is (using the MAX-Rule):

NOP('Alice', t₂) = {('ViewZone.org', 0.03)
*('On Web Services', 0.01), ('Modeling Solutions', 0.01), ('Research and Development', 0.01), ('An OMG Update For Managers', **0.31**), ...*
*('Allen, Paul', 0.07), ('Balzer, Marc J.', 0.07), ('Bryce, Ciarán', 0.07), ('Harmon, Paul', **0.67**), ...*
*('OMG', **0.47**), ('The Continuing Relevance of the OMG', **0.18**), ('JAVA', **0.46**), ('Java, Enterprise JavaBeans and CORBA', **0.4**)*
}

Suppose now that after 10 minutes Alice is requesting Paul Harmon's biography (action A_1) in zone Z_4. This corresponds to the *PageRequest(P₃)*. After this, her NOP is as follows:

NOP('Alice', t₃) = {('ViewZone.org', 0.01)
*('On Web Services', 0.05), ('Modeling Solutions', 0.05), ('Research and Development', 0.05), ('An OMG Update For Managers', **0.17**), ...*
*('Allen, Paul', 0.03), ('Balzer, Marc J.', 0.03), ('Bryce, Ciarán', 0.03), ('Harmon, Paul', **0.76**), ...*
*('OMG', **0.42**), ('The Continuing Relevance of the OMG', **0.07**), ('JAVA', **0.62**), ('Java, Enterprise JavaBeans and CORBA', **0.57**)*
}

If now Alice is staying on page P_3 for some minutes and after that she leaves the Website without performing an action, then we do not take the time she has spent on P_3 into account for the NOP calculation.

The profile of a user is also re-calculated any time he gives an explicit feedback.

E. *User Feedback*

To measure the accuracy of a NOP we

- Show the NOP to the user.
- Use a feedback mechanism, where the user is asked directly to enter his preferences, e.g., by presenting the values of his NOP and asking him to verify or correct them.

From the feedback given by the user we build up another profile called the feedback profile (FP).

Both Gugubarra 1.0 and 2.0 compare the NOP with the FP and by using different calculations produces a derived profile (DP as a base for further calculations (see [9]).

Let's assume in our example that during the registration in ViewZone the user is first asked to give an initial explicit feedback, by telling his interests in the topics. This explicit feedback is then used as the starting NOP.

Let's consider our second user "Bob", who (in contrast to Alice) has specified his initial level of interest as defined in the registration form (0 minimal to 1 highest):

'An OMG Update For Managers':	***0.5***
'Business And Technology On The Internet':	*0.0*
'E-Government':	*0.0*
'Middleware For Mobile Computing':	***1.0***
...	...

After Bob has registered and has entered the above values (initial explicit feedback) these numbers are used as initial NOP. Therefore Bob's NOP contains the following topics/weights, indicating his declared interests:

NOP('Bob', t0) = {
('ViewZone.org', 0.0), ('Allen, Paul', 0.0), ('Balzer, Marc J.', 0.0), ('Bryce, Ciarán', 0.0), ('Harmon, Paul', 0.0),...,
*('On Web Services', **1.0**), ('Modeling Solutions', 0.0), ('Research and Development', 0.0), ('An OMG Update For Managers', **0.5**), ...*
}

In Gugubarra a user can be requested to confirm or correct his NOP that has been generated so far, triggered by customized rules defined by the Web site owner.

In addition the user could provide directly a feedback any time he views his NOP and wants to correct or confirm it.

We use the Feedback mechanism to "learn and compare" the interests of the users. We do not question the user feedback.

However, we do not take the FP as it is as the new starting NOP. We in fact calculate four different measures as illustrated in Fig. 4.

First we calculate the differences (ND, D, FD) of the different profiles we have as shown in Fig. 4. These differences are then used to calculate a Derived Profile based on rules, trying to filter situations the Web site owner expects wrong answers inside the FP. The Derived Profile will be used as the new starting point for the next NOP generation. For more details see [9].

$$\begin{tikzcd}
NOP(t_{i-1}) \arrow[r, leftrightarrow, "\text{NOP-difference (ND) at time } t_i"'] \arrow[d, leftrightarrow, "\text{difference (D) at time } t_{i-1}"'] & NOP(t_i) \arrow[d, leftrightarrow, "\text{difference (D) at time } t_i"] \\
FP(t_{i-1}) \arrow[r, leftrightarrow, "\text{FP-difference (FD) at time } t_i"'] & FP(t_i)
\end{tikzcd}$$

Fig. 4. Visualization of the variables D, FD and ND

Coming back to Alice, suppose she is visiting ViewZone.org regularly and after one month she checks her NOP and sees the following values:

NOP('Alice', t_4) = {('ViewZone.org', 0.01)

*('On Web Services', **0.4**), ('Modeling Solutions', 0.01), ('Research and Development', 0.01), ('An OMG Update For Managers', 0.27), ...*

*('Allen, Paul', **0.4**), ('Balzer, Marc J.', 0.01), ('Bryce, Ciarán', 0.01), ('Harmon, Paul', **0.76**), ...*
*('OMG', 0.5), ('The Continuing Relevance of the OMG', 0.02), ('JAVA', 0.5), ('Java, Enterprise JavaBeans and CORBA', **0.6**)*
}

Alice's NOP shows e.g. a little interest in the author 'Allen Paul' and the series of articles 'On Web Services', and a medium interest in the author 'Harmon, Paul' and the series of articles on 'Java, Enterprise JavaBeans and CORBA'.

If Alice was *never* interested in the author 'Allen Paul', she could give a feedback telling her "supposed" interests:

FP('Alice', t_5) = {('ViewZone.org', 0.01)

*('On Web Services', **0.6**), ('Modeling Solutions', 0.01), ('Research and Development', 0.01), ('An OMG Update For Managers', 0.27), ...*

*('Allen, Paul', **0.0**), ('Balzer, Marc J.', 0), ('Bryce, Ciarán', 0), ('Harmon, Paul', **1.0**), ...*
*('OMG', 0.5), ('The Continuing Relevance of the OMG', 0.02), ('JAVA', 0.5), ('Java, Enterprise JavaBeans and CORBA', **0.6**)*
}

The NOP are therefore re-calculated resulting into a new NOP. The re-calculation can be done by different rules as defined in [9].

III. Clustering Web Visitors

The use of NOPs opens up several interesting possibilities, for example to cluster together visitors of a Web site with "similar" interests and offer them then targeted/personalized e-services.

Clustering Web users by their behavior can also be useful for measuring "trends" in a Web community, and again it can be a valuable information for creating customized e-services.

Clustering Web users is also useful for e-CRM, where we build long-term-relationships and increase e-customer loyalty that is the degree to which a Web customer will stay with a specific vendor or a brand.

In most research done until now, Web users are clustered by their click streams or by their visited pages [4], [12], [13]. By using the *non-obvious profiles* approach we have the possibility to cluster Web users by looking at the content of the Web pages, and the users' interests in several topics related to the pages.

Once user profiles are created Gugubarra 2.0 is designed to allow clustering users by analyzing their current profiles. Different clustering techniques have been used to cluster profiles as e.g. in [2] the famous *k-means*. Gugubarra uses the ***Takahe Cluster Algorithm*** [8] (Takahe is a bird from New Zealand), which is a combination of hierarchical clustering with a centroid based method including a priority. It allows clustering Web users by similar interest in several topics.

The Takahe cluster-by-priority algorithm has a better computation time than the k-means algorithm because of the reduction in the number of topics. It works well also with high number of topics, since the introduction of a given threshold allows us to discard topics from the beginning that are considered not relevant for the clustering.

Our new cluster algorithm has the following properties:

- Combination of a divisive clustering algorithm with a centroid-based method.
- No overlapping in one level, but overlapping through the levels.
- Reducing the dimension of topics, by introducing a priority depended sequential examination of the topics including stop criteria (threshold topic, cluster number).
- Flexible through the possibility for the owner to manipulate different parameters (g, top of the dendrogram).
- Possibility to associate a meaning to each cluster, corresponding to a value of a pre-defined scale of interest.

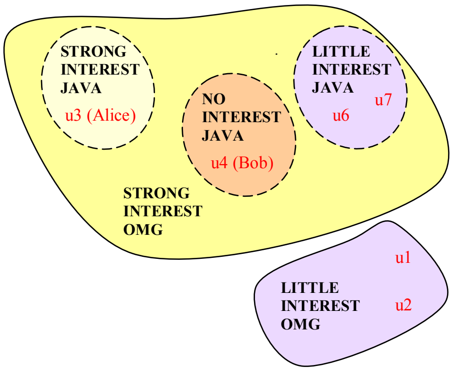


Fig. 5. Visual Extraction of topic JAVA on level two at a certain time t.

In the example on Fig. 5 we show seven users, which are clustered by their interests with the Takahe Cluster Algorithm.

IV. EThicS

No work on User Profiles can avoid issues on Ethics.

We would like to distinguish between two classes of Ethics problems:

i) What we call the *Business Code*, including issues such as Data Protection, Security,

and a broader issue we call

iv) *Social Code*: which relates to the issues on how addictive could be the new Internet technologies which encourage a higher level of "stickness" to a Web site.

We would like to raise the awareness in the research community of the danger that improper use of such technology can be very damaging. We believe this issue requires a broad discussion and attention in the research and industrial community.

In our opinion, the Google's Ten Golden Rules [5] are good examples of a behavior rules for Business code.

We would like to suggest introducing similar behavior rules for Social Code to help avoiding that

- Stickness equals sickness,
- Repeated becomes addictive.

V. Future Work and Related Work

One of the topic of research we plan to investigate in the future is to analyzing "trends" for a Web user community and when possible to make "forecasts" of the pattern of interest of the Web community. To study how clusters change over time, we are considering using 2D functions to visualize the motion of a Web user through the clusters over time.

We are also interested in using clusters to make a "forecast". This is related to areas of collaborative filtering (Evolutionary computing). A "forecast" should make an assumption on how the clusters and the interests will change from the present to the future.

In Gugubarra 2.0 we are also working on taking into account the various information available on Click streams.

User Profiles are used for example by recommender systems [10] or to create adaptive Web stores like [1]. These applications are not our main focus.

User Profiles always include different information and they are defined in different ways. In [1] they use likelihood e.g. to assume if a user has more low, medium or high technical interest. A User Profile can always be divided into several semantic (domain specific) parts. For example a part for a learner can include relations to other teammates and mentors like suggested in [6]. Our specific non obvious part is concentrated in topic interests.

Compared to systems such as Web Usage Mining [11] we include more granular information by the introduction of zones and actions. This way we do not only rely on the recorded click stream. Of course this implies that the Web site owner needs to invest the effort for defining zones and actions.

Another peculiarity of Gugubarra is the combination of the NOPs and the Feedback Profiles with custom rules for the generation of the NOPs. This is missing in systems such as the one of Acharyya and Ghosh [14] or of D'Ambrosio, Altendorf and Jorgensen [15].

Acknowledgment

Peter Werner, Naveed Mushtaq, Robin Aly and Evaldas Veršelis gave valuable input to the design and helped with the implementation of the Gugubarra 1.0, and the design of Gugubarra 2.0.

References

[1] L. Ardissono and A. Goy, *Tailoring the interaction with users in electronic shops.* In Proc. 7th Int. Conf. on User Modeling, Banff, Canada, 1999.

[2] Ed H. Chi and Adam Rosien and Jeffrey Heer, *LumberJack: Intelligent Discovery and Analysis of Web User Traffic Composition.* Proc. of ACMSIGKDD Workshop on Web Mining for Usage Patterns and User Profiles, ACM Press, Canada, 2002.

[3] DBIS J.W. Goethe University, *CeBIT Presentations for Gugubarra 1.0.,* http://www.dbis.informatik.uni-frankfurt.de/news/?mode=art&l=e&aid=2&tmid=3&smid=7

[4] Y. Fu, K. Sandhu, and M. Shih, *Fast Clustering of Web Users Based on Navigation Patterns.* In World Multiconference on Systemics, Cybernetics and Informatics (SCI/ISAS'99), pages 560--567, Orlando, FL, 1999.

[5] Google, *"Ten Golden Rules".* http://www.msnbc.msn.com/id/10296177/site/newsweek/

[6] IEEE LTSC, *PAPI Learner Standard*. Draft: http://edutool.com/papi/ and IEEE LTSC: http://ieeeltsc.org/.

[7] Natascha Hoebel, Sascha Kaufmann, Karsten Tolle and Roberto Zicari., *The Design of the Gugubarra 2.0: A Tool for Building and Managing Profiles of Web User.* DBIS-Report in Preparation, Frankfurt, 2006.

[8] Natascha Hoebel and Roberto Zicari, *On Clustering Visitors of a Web Site by Behavior and Interests.* Dbis report May 2006.

[9] Naveed Mushtaq, Karsten Tolle, Peter Werner and Roberto Zicari., *Building and Evaluating Non-Obvious User Profiles for Visitors of Web Sites.* IEEE Conference on E-Commerce Technology (CEC 04), San Diego, California, USA, 2004.

[10] G. Shani, D. Heckerman, and R. Brafman., *An MDP-based recommender system.* Journal of Machine Learning Research 6: 1265-1295, 2005.

[11] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, Pang-Ning Tan., *Web usage mining: discovery and applications of usage patterns from Web data.* ACM Press New York, USA 2000 ISSN:1931-0145.

[12] Qing Wang, Dwight J. Makaroff, H. Keith Edwards, *Characterizing Customer Groups for an E-Commerce Website.* Proceedings of the 5th ACM Conference on Electronic Commerce (EC '04). ACM Press, New York, 2004.

[13] Weinan Wang, Osmar R. Zaïane, *Clustering Web Sessions by Sequence Alignment.* Proceedings of the 13th International Workshop on Database and Expert Systems Applications, pages 394-398, September 2002.

[14] S. Acharyya, J. Ghosh, *Context-Sensitive Modeling of Web-Surfing Behaviour Using Concept Trees*,.WEBKDD 2003, 2003.

[15] B. D'Ambrosio, E. Altendorf, J. Jorgensen, *Probabilistic Relational Models of On-line User Behavior*. WEBKDD 2003, 2003.