T-Bookmarks: Providing TV-related Web Resources at Anywhere

Douglas Véras, Thiago Prota, Carlos Ferraz
Center of Informatics
Federal University of Pernambuco (UFPE)
Recife, Brazil
{dvs,tmp,cagf}@cin.ufpe.br

***Abstract*— In many parts of the world, the technologies of Digital TV and Connected TV are in a vast number of homes. Given this scenario, there are many sources of information in which the users may obtain access, especially with the union of "broadcast" and "broadband" technologies in a single TV set. Thus, this paper proposes a model (named T-Bookmarks) for convergent applications in Hybrid TV, responsible for retrieving Web resources of different categories (news, blogs, images, and others), related to programs watched by users. Also, our proposal takes into account the way resources are presented to the user, since she does not always want to visualize Web resources on the TV screen, as the experience of watching a TV program may be compromised. Therefore, that information is provided through a Web server so it may be accessed in any device connected to the Internet at any moment.**

Keywords-digital TV; connected TV; ubiquitous computing; distributed systems; recommender systems.

I. INTRODUCTION

In many parts of the world, Digital TV (DTV) is in a vast number of homes [11]. Regarding today's TV technology, two aspects may be considered: *a) transmission via radio broadcast characterized by the same information being sent to several TV receptors at the same time* and *b) the broadband technology that allows the distribution of information for the user on demand*. The latter one is known as "Connected TV", which is vastly present in the international market [5]. However, one technology does not exclude the other, and they may be complementary, according to Hybrid TV (Digital and "Connected") technology [4], increasing the interaction experience of the user and the quality of information she receives.

Nevertheless, the searching for information in these technologies is motivated by different causes. The traditional searches or recommendation for TV shows/sites on TV become linked, once the search for sites, for example, is generally oriented to tasks arising from the search for a service that attends a necessity, entertainment, curiosity, and others. Particularly, on a Connected TV, this search or recommendation for a certain content on the Web (a film, a song, Web pages, extra contents, etc.) may arise from the TV show being watched by the user who might have used a Web browser and found something of her interest, many times cluttering the continuous flow of the TV show. This fact may be noted, given the number of people that watch TV and surf the Web at the same time [7], sometimes even accessing the site of the TV show under exhibition. It reveals that TV and Web are complementary media.

Thus, there are some new challenges related to the recommendations made to the users of this new kind of hybrid TV, including: what information would be useful to the user when such information is in a broader context, not only related to a similar TV show, but also to any resource available in the Web? How are those recommendations exposed to the user, since many Web resources of different categories (videos, music, images, news, and others) may be suggested depending on the TV program the user watches? How must that information be arranged to make its access be easy and intuitive? In the last case, the displaying of results may be done in a Web page, so the user may access this information using his/her Personal Computer (PC), Smartphone, or even the Hybrid TV, through a Web browser. Therewith, access to available information on the Web, related to the TV shows watched by the user, would be done in a simpler and more direct way than if it was done searching for those resources in a conventional way through Web browsers, either on the "Connected" TV or PC.

Therefore, this paper presents a distributed system, acting as a model to convergent applications, based on Hybrid TVs, linking aspects from the DTV and Web fields. This system, called T-Bookmarks, provides recommendation of Web resources related to TV shows watched by the users. For that, T-Bookmarks architecture serves as a basis for the construction of applications responsible for recovering such Web resources, and organizing them in a Web server. Thus, the recommendation results may be accessed in any Internet device.

The remainder of this paper is structured as follows: In the next section, some related works are presented regarding to the access of additional information about TV shows, comparing them to proposed system in this paper. In Section III, the T-Bookmarks solution is presented, while in Section IV, a case study, as a proof of concept, of the system is presented under a real environment. In Section V, their implementation and profile algorithm are discussed. Section VI concludes the paper and gives hints on future work.

II. RELATED WORKS

A system that searches for Web contents-related to the TV shows and provides integration with other devices (smartphones, for example) is proposed by Patel et al. [8]. Although this system has provided many capabilities that make the search easier, such as: indication/correction of

 ISBN: 978-1-61208-289-9

terms inserted incorrectly and exhibition of contents related to the searched terms, including different categories of contents (images, videos, related news, and others), the user still need search for information or TV shows of his interest which requiring the insertion of text to determinate the searched item. This may result in a loss of the user's time when searching for contents of interest. On the other hand, a system that brings these contents in an automated manner (implicitly to the user) would make this process easier, with the option of searching the TV show that is currently in exhibition or based on the history of TV shows preferred/watched by the user. Thus, this system is proposed in this article (named T-Bookmarks).

Another work that aims the convergence between TV and Web is proposed by Dimitrova et al. [1]. In it, two recommendation systems of content related to TV shows were conceived. The first one is a recommendation system of news (MyInfo), while the second one recovers information about films (InfoSip). To access news (MyInfo), the users may select one of six categories: weather, traffic, sports, financial news, headline and local news. Besides, they may see the filmographies (InfoSip) based on the history of what is watched on the TV. By using a remote control, users may receive information related to the context of film as predefined questions such as: "Who is this actor?", "What is this song?" and "What city is this?", among others.

However, this information is not explored completely, since it is limited to certain TV shows types. Also, it does not allow the user to access resources available on the Internet, because it is restricted only to the TV device. Instead, T-Bookmarks system provides access to information (categorized Web resources) for any kind of TV show and from any device with Internet access.

Ma and Tanaka [6] present a method for dynamic integration between TV shows and Web-related contents, where a prototype system called WebTelop is described. Web pages related to the TV show's content are retrieved automatically in real time and they are used to add related extra content to TV show in the form of subtitles. During TV show transmission, a virtual agent is displayed on the screen to help the user to surf the related web pages. However, the work proposed in [6] disposes resources only based on the TV show being exhibited and to be accessed on the TV itself. Alternatively, T-Bookmarks system provides resources for the running TV show as well as for the TV shows most watched by the user. Furthermore, he may visualize the resources in a Web page, on the "Hybrid TV", tablet, smartphone, and other devices with access to the Internet. Besides that, the resources are persisted for access at any moment.

III. T-BOOKMARKS SOLUTION

Basically, the T-Bookmarks is a distributed system that aims to serve as base in the construction of applications responsible for capturing, through the usage history, which TV shows are "favorites" by the user, and after that, recovering Web resources of different categories (news, videos, images, and blogs), related to these "favorite" TV shows. Furthermore, the user have the option to recover Web resources related to the current TV show that he is watching, without requiring to generate a list of "favorite" TV shows. Figure 1 shows the general idea of the T-Bookmarks system.

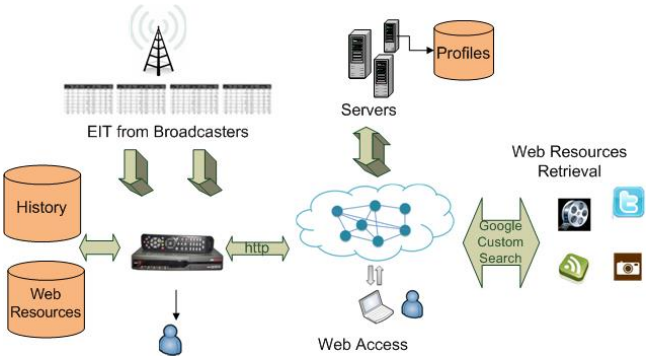

Figure 1. T-Bookmarks solution overview.

As can be seen in Figure 1, T-Bookmarks uses the EIT (Event Information Table), which are tables emitted by broadcasters together with the TV shows in order to obtain the exhibited TV show's data, such as: title, channel's name, TV show's time/date of begin/end, and others. More than the information of TV shows and programming guide, the receptor must be capable of storing the user's usage history, allowing the user profile's creation with his "favorite" TV shows being captured in an implicit manner.

From there, the system requires a search API (Web service) for Web resources according the user's profile (or search for the TV show in exhibition) since the reduced processing capability of TV devices as well as manipulation of a large mass of data and resources available on the Web. After the response of this search, based on the favorite TV shows or on the current TV show, the addresses (Web URLs) are saved in a categorized way according to their type (videos, news, images and blogs). These URLs are saved in the TV itself and, after that, they are sent to server, where the resources are provided to the user.

Finally, another important point to be highlighted is in the manner how these resources may be accessed by the user. Once these resources are in a server, they may be accessed by any platform with Internet access, including the "Connected TV" itself, at any moment.

A. *Use Scenario*

In this section, a T-Bookmarks use scenario is presented in order to illustrate the main functionalities provided as description of the system general idea. For this purpose, a short story is described from the point of view of the user, as follows.

"A viewer, turning on its TV, decides to watch a talk show, but in the meantime, he surfs the Internet (on PC) seeing some emails. During that TV program, he starts to search the Web for videos related to the talk show in exhibition, but he does not want to waste time searching for these videos, so he activates the T-Bookmarks by selecting the option to recommend resources related to the running TV show, and continues to read his emails. After reading some emails, he then decides to see what were the Web resources,

especially videos, related to the talk show. The application creates a list of Web resources related to the running TV show and makes it available on a Web server, thus it can be accessed anywhere (including his PC). By accessing the list, the user finds some interviews that had not watched yet and decides to access them on his computer. Among these TV shows, the viewer watches some ones for a short time, while he watches other ones completely, and so on. On the weekend, he decides to surf the Internet looking for varied content, nothing very objective, but related to the TV shows he more liked, which also was inferred implicitly by the T-Bookmarks. Then the user opens the application and requests resources related to his favorite TV shows. Hence, he accesses a resource list on a Web page by selecting blogs related to one of its favorite TV shows by visualizing them on its own TV via Web browser."

B. Architecture

T-Bookmarks is a distributed system based in the client-server model. The client model has an architecture divided in layers and makes requests to a Web service [3] in order to find Web resources (a list of Web addresses – URLs). In the following sub-topics, the T-Bookmarks Client and Server modules are presented, respectively.

1) Client Module

The Client module's architecture (see Figure 2) is divided in three layers: Model, View and Control (MVC). The inferior layers provide services for the adjacent superior layers through the facade present in each one. Below, the essential components of each layer are described.

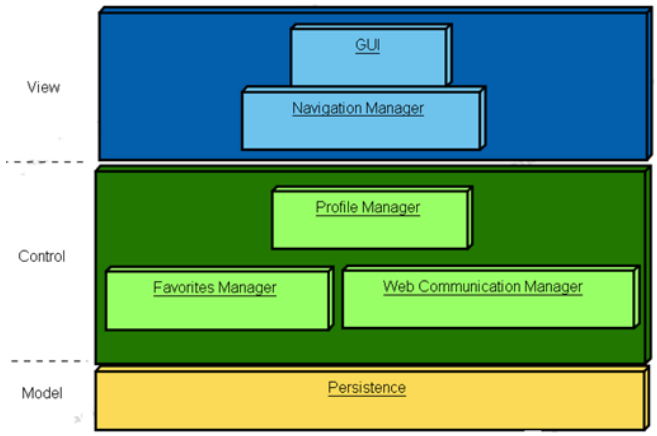

Figure 2. T-Bookmarks client module architecture.

The View layer is responsible by interface with the user. It is used to receive user's input data or requests and to present the result of the interaction with the system to the user. In the T-Bookmarks, there are two main components situated in this layer as described in the following:

- GUI – this component is responsible for exhibiting the recommendation options of Web resources ("Current" or "History" based) to the user as previously mentioned in the use scenario (Section III.A). Furthermore, it shows to the user, his unique identifier, that will be utilized to access the Web page containing the recommendation results;
- Navigation Manager – this component is responsible for providing navigation between recommendation options (exhibited by GUI) to the user through remote control. Besides it is responsible for passing the user's choices to the system. When the user selects an option (pressing the "ENTER" key from remote control), the Navigation Manager forwards this information to the Profile Manager component, in the Control layer, and then it sends a feedback message to the user through GUI, so he is aware that the desired option was selected.

The Control layer, situated between the View and Data (Model) layers, defines the core behavior of the system, interpreting the user's actions and determining what the system should do. Three main components are in this layer, as briefly described in the following:

- Profile Manager – this component interprets/receives the requests from Navigation Manager component. Furthermore, it is the Client main component, responsible for: *a) managing the user's profile through watched TV shows*, and accessing services in the Model layer; *b) requesting the Web resources to the Web Communication Manager component* according the selected option by the user ("Current" or History"); and *c) extracting the basic TV show's data such as*: title, channel name, duration time, among others, and transforming them in data structures represented by "business" objects of the application like: user profile, search criteria, and others;
- Favorite Manager – this component is required by the Profile Manager with objective of calculating the TV show "score" in order to classify it as a favorite TV show or not. The Favorite Manager component also creates the list of favorite TV shows, ordering the TV shows with highest score first. In this component, a profiling algorithm is implemented (proposed in [10]), based in the time that the TV show is watched; and
- Web Communication Manager – this component is responsible for accessing services that are out of the Client module (available on the Web). Its objectives are: *a) recover Web resources:* through an API (Google Custom Search API [3]) according each resource category - Videos, News, Images and Blogs. This Web API invokes functions of this component when a request is completed (callback). Thus, the Web Communication Manager accesses services from Model layer in order to persist the search results, distinguishing Web resources from "Current" or "History" recommendation; *b) communication with Server module:* sending the Web resource lists so that they can be accessed through the Web page provided by the Server module and retrieving the user identifier, generated in the Server from first time that the application is load. This identifier is sent to the GUI component in

 ISBN: 978-1-61208-289-9

order to the user can use it to access the Web resources.

The Data layer (Model) models the data behind the "business" layer. Thus, it is only responsible for storage, manipulation and generation of data. In the architecture of the T-Bookmarks, the Persistence component offers services for persistence of data from the user profile, with his watched and favorite TV shows as well as services for persistence of Web resources as result of user requests.

2) Server Module

The Server module architecture follows the MVC pattern (similarly as Client module architecture), as shown in Figure 3. This architecture is directed basically to receive Client module data and to save them in a database, as well as receiving requests from the users through Web browsers in order to show them the persisted data.

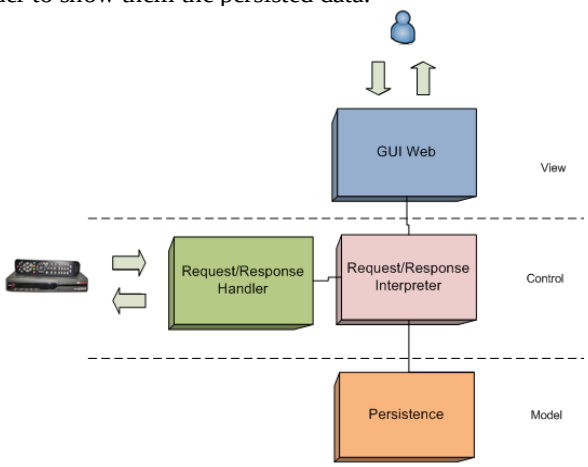

Figure 3. T-Bookmarks server module architecture.

As can be seen in Figure 3, the Server module has three layers composed by four main components according to the following descriptions:

- Request/Response Handler – this component is common to the majority of Web servers, being responsible to receive the requirements of services, returning the results through a response. In this case, servlets are utilized as entry port for requests from the Client module that communicates with the server for: *a) Obtaining identifier for posterior access of the recommended data*. This identifier is unique for each application (not user) and generated automatically by the server in the Requirement/Response Interpreter component; and *b) Sending the Web URL recovery through the API:* in order to make it available for access through Web page;
- Request/Response Interpreter – this is the main component of the server, responsible for receiving the requests from the Client module as well as the Web interface (GUI), interpreting them and taking the correspondent actions. After these actions are taken, responses are sent back to the Client module or user by accessing the Web page. According to the respective requests, the functionalities provided to the Client module are: *a) Generation of unique identifier (ID) for each application:* where once time loaded, the application requires the identifier and after server response, saves the ID locally, similar to a Web cookie; and *b) Interpreting data sent by the Client module:* according user's ID, kind of resource (images, videos, news or blogs) and Web recovery mode (based on the user's usage history or on the running TV show). Beyond interpreting actions from the Client module, this layer is also responsible for interpreting user actions through GUI Web. These actions are: requesting the exhibition of Web resources having as entry an identifier; selecting Web resources categories for exhibition in the browser itself. For this reason, the Requests/Response Interpreter accesses the data saved in Persistence component and returns the Web resources to GUI Web component;
- GUI Web – this component belongs to Presentation layer. It is responsible for showing to the user the Web resources sent from the Client module, according to the user ID. For each identifier, the GUI Web presents a list of resources solicited by the user in the Client module in a categorized way: by Web resources related to the running TV show, and, by Web resources related to the usage history, both cases are classified according to kind of resources: images, videos, news, and blogs. In addition, the user can type the ID, through a search box, in order to retrieve his desired Web resources. So, with a list of available resources he can select a resource which will be opened on same page or in other page, as the user wishes; and
- Persistence – this component, representing the Data Layer, provides storage and searching functions through files. Among these files are: files with the last ID of a registered application, serving for querying as well as storage (when it is necessary increase it). In addition, this layer offers to reading/writing in files with the user's Web resources, persisted separately according user's ID, their Web resource category (images, videos, news or twitters) and their recommendation option ("current" or "history").

IV. CASE STUDY

As a proof of concept, in this section, an experiment performed in the Brazilian System of Digital TV (SBTVD) [2] under a real environment is described. For this, a Smart TV Samsung [9] was used, with Internet access, where the T-Bookmarks client application was installed. Moreover, the server application was executed in a computer from the same network (intranet) of the TV.

As it was previously described, every client application on the TV has an identifier generated automatically through a request to the server when the application is loaded at first time on TV. If this number is generated correctly, no alert is displayed on the screen and the user can select one of the options using the remote control (navigation by arrow keys

 ISBN: 978-1-61208-289-9

"UP" and "DOWN", pressing "ENTER" on the desired option). When one option is selected, a feedback message is sent to the user becoming him aware of the selected option, while he is waiting the system response. We recall that the application is executed in "background", where the user profile is implicitly created based on the TV shows which he watches.

Figure 4 shows the response obtained when the user chooses the "Current Recommendation" option, where resources are recovered and sent to the server application. Hereafter, according to this response, the data can be accessed on the server's website through the "4" identifier (generated and showed on TV). Thus, this page may be accessed by a browser on PC, smartphone, the Samsung Smart TV, or any other device with Internet access.

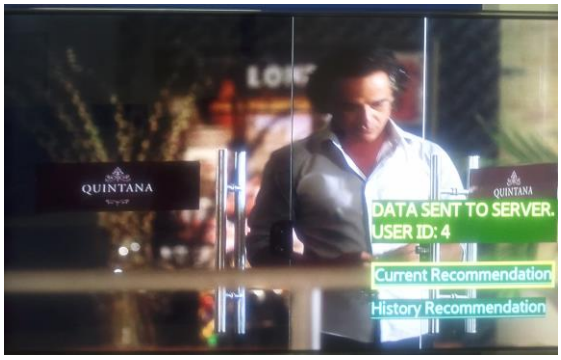

Figure 4. User selects the "Current Recommendation" option.

Likewise, the user can select the "History Recommendation" option. Then the T-Bookmarks retrieves Web resources related to his favorite TV shows and shows to the user the same response on the TV screen (confirmation of the data sent to the server, with his application ID).

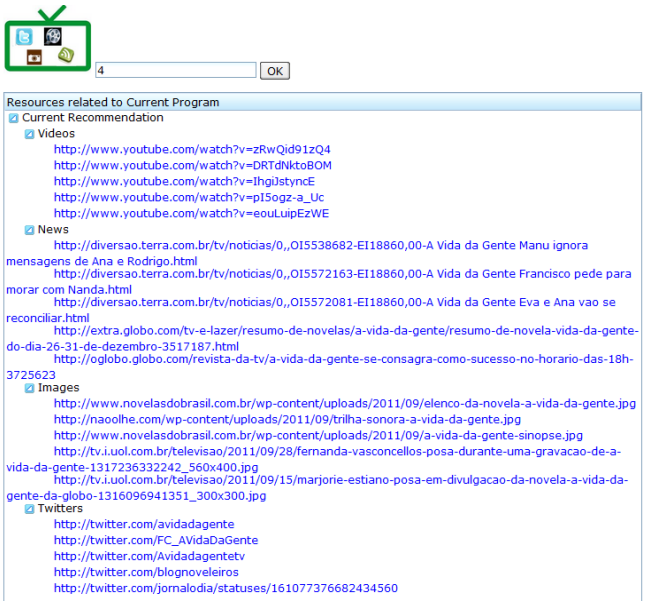

Figure 5. Recovery results of web resources on web page.

The result of these requests may be viewed through the Web page available on the server. In the first phase of this experiment, a user watched the "A Vida da Gente" TV show at the "Globo HD" channel and he selected the "Current Recommendation" option; thus, the T-Bookmarks has generated a list of resources available on the web page associated with the ID "4", as seen in Figure 5.

In the second phase of this case study, a user watched TV shows in the created environment during a period of one week (Sunday to Saturday). In this period, the T-Bookmarks has collected information about which TV shows has been watched and their respective scores according "watched" timing (in minutes) relative to TV show total duration, as shown in Table 1.

TABLE I. LISTING OF TV SHOWS WATCHED WITH THEIR RESPECTIVE SCORES.

TV Show	Channel	Duration	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Satuday	Total Score
Globo Esporte	Globo HD	30	0	0	0	15	0	5	0	8,365
Jornal Hoje	Globo HD	30	0	30	30	30	20	5	0	31,685
Video Show	Globo HD	35	0	0	0	0	20	5	0	8,454
Mulheres de Areia	Globo HD	100	0	90	90	90	90	55	0	37,679
Globo Notícia	Globo HD	3	0	3	3	3	0	3	0	28,2
Malhação	Globo HD	30	0	30	30	30	0	30	0	30
A Vida da gente	Globo HD	50	0	20	20	0	50	50	20	29,648
Fina Estampa	Globo HD	80	0	40	40	40	80	60	60	37,840
Jornal da Globo	Globo HD	30	0	0	0	0	30	30	0	15
Glee	Globo HD	50	0	0	0	0	50	0	0	7,833
Vidas em Jogo	Record HD	45	0	45	45	45	45	45	0	38,75
Rei Davi	Record HD	60	0	0	50	0	50	0	0	14,342
Dr. House – (Season 5)	Record HD	60	0	0	0	0	20	0	0	4,138
Fantástico	Globo HD	140	60	0	0	0	0	0	0	4,478
Altas Horas	Globo HD	120	0	0	0	0	0	0	30	2,968

Also, a list of favorite TV shows of this user was created according the first three TV shows "most favorite", in this order: "Vidas em Jogo" ("Record HD" channel) with score 38.75, "Fina Estampa" ("Globo HD" channel) with score 37.84 and "Mulheres de Areia" ("Globo HD" channel) with score 37.679. Thus, these three TV shows had related Web resources recovered and saved on the server, where the user could access them, similarly to what occurred in the recommendation from the "Current Recommendation" option (shown in Figure 5).

V. DISCUSSION

In this section, the main implementation aspects of the T-Bookmarks are discussed, as well as the use of the algorithm proposed in [10].

A. *Implementation*

The client-server architecture definition, for this novel design of recommender system cross-domain (TV and Web), is necessary due to computational limitations of TV sets and the mass of data to be fetched (any resource on the Web). Thus, T-Bookmarks uses a search service available in the Web cloud, removing the complexity of its implementation from client application.

As it was described previously in this paper, there are two options for retrieving Web resources provided to the

user: *a) related to the TV show in exhibition;* and *b) related to user's favorite TV show list (implicitly generated)*. For this, some data from EIT tables regarding to the TV shows are needed. However, few data are extracted from these tables compared to the amount of data available, so future implementations could utilize more data, such as the TV show "category", in order to bring more relevant results to the recommendation. Furthermore, the use of data from the broadcasters may not always be sufficient, even the possibility of errors by them, which directly affect the user profiling, implicitly made by the application. Thus, in order to alleviate this problem, consultation services for TV shows information on the Web may be used, for example.

Currently, T-Bookmarks searches for Web resources belonging to four categories: Video, News, Images and Blogs. However, new categories could be added to the client-server system, even using the same search API currently used, as well as through different search engines, which could also be included in the proposed architecture.

Another aspect that could be changed, according to the requirements, is the amount of recommended resources, since it is currently limited to five resources by category (Video, News, Images, and so). This was defined in order to provide the user only the most relevant Web resources by the Google Search API. Furthermore, the amount of favorite TV shows to be used in the search results is limited to only three (the first three programs in the favorite list). Multiplying these factors, the T- Bookmarks provides to the user sixty Web resources in the request for the list of favorites, while the search based on the running TV show retrieves twenty Web resources.

Furthermore, the recovery of Web resources was based on a single technique of recommender systems: based on user history, determining its favorites TV shows list implicitly. However, other techniques (ontologies, context-aware, collaborative filtering, among others) could be used to improve the relevance of the resources available to the user as well as other algorithms may be used to generate the user profile implicitly, different from the algorithm used currently [10].

Finally, the use of an identifier does not allow the installed application can be used by different users. For this, an authentication mechanism would be necessary, that it could hinder the T-Bookmarks use, because the user would need to register and login to each application use. Moreover, the application could be personalized in many ways such as: the choice of the quantity of resources searched; the choice of resource categories shown; among others.

B. Recommender Algorithm

According to the implementation of the algorithm proposed in [10] and the result of the case study presented in Section IV (see Table 1), the three TV shows with the highest score were used to search for Web resources when the user selected the recommendation based in favorite TV shows on the T-Bookmarks GUI.

The calculation of the TV show score takes into account its relative time watched and total duration. As we have seen, the "Globo Notícia" ("Globo HD" channel) has duration of three minutes while the "Malhação" ("Globo HD" channel) TV show has duration of thirty minutes. In both cases, the user watched the entire TV show on the same days and the same number of times each, but in the first case the score was 28.2, while in the second one the score was 30. This is a result of factors *Wf(d(c))* and *Se(d(c))* [10], which increase the score of programs with higher durations.

However, this algorithm does not take into account the amount of times a TV program is showed during the week, so these TV shows hardly enter into the top of the list of favorites, as it is the case of "Fantástico" and "Altas Horas" programs, which are showed only once a week; "Rei Davi", which is showed twice a week, among others. Furthermore, TV series like "Glee" and "Dr. House" showed only in one or two days a week should have more "weight" in their scores because they are TV shows with certain "inter-dependence" between each episode.

VI. Conclusion and Future Work

This paper has presented a distributed system, called T-Bookmarks, as a proof of concept, based on a client-server architecture, which captures which TV shows are the user's "favorites" (implicitly) and then retrieves Web resources from different categories related to these "favorite" TV shows. In T-Bookmarks, recommendation results are available in a Web page so the user can access them on any device with Internet access. Thus, users do not have to spend much time searching for Web resources related to the TV shows they are watching as this task is performed by the system. Finally, this way of viewing also allows the user to see the Web resources when he feels needed (any time), since the Web resources are saved in a Web server.

As future work, we plan to perform tests with multiple users to enhance the prototype according to the needs raised in the experimental tests. These tests aim to assess: *a) the prototype interface (GUI)*; *b) the algorithm for calculating the implicit user profile*; *c) the relevance of Web resources presented to users according to their watched TV shows.*

References

[1] N. Dimitrova et al., "Content augmentation aspects of personalized entertainment experience.", Procs. of the 3rd Workshop on Personalization in Future TV, 2003, pp. 42-51.

[2] Forum SBTVD, available at http://forumsbtvd.org.br, retrieved: July, 2013.

[3] Google Custom Search API, available at https://developers.google.com/custom-search/, retrieved: July, 2013.

[4] HbbTV, ETSI Technical Specification 102 796, "Hybrid Broadcast Broadband TV", available at http://www.hbbtv.org/pages/about_hbbtv/specification.php.

[5] R. Jacoby, "Why Connected TV Is Poised To Revolutionize Entertainment", available at http://mashable.com/2011/05/11/future-connected-tv/, retrieved: July, 2013.

[6] Q. Ma and K. Tanaka, "WebTelop: dynamic TV-content augmentation by using web pages.", Proc. of IEEE International Conference on Multimedia and Expo (ICME), 2003, Vol.2, pp. 173-176.

[7] N. Wire, available at http://blog.nielsen.com/nielsenwire/online_mobile/40-of-tablet-and-smartphone-owners-se-them-while-watching-tv/, retrieved: July, 2013.

 ISBN: 978-1-61208-289-9

[8] M. Patel et al., “Google TV search: dual-wielding search and discovery in a large-scale product.”, Procs. of the 1st Intl. Conf. on Designing Interactive User Experiences for TV and Video, Publisher: ACM, 2008, pp. 95-104.

[9] Smart TV Samsung, available at http://www.samsung.com.br/smarttv/, retrieved: July, 2013.

[10] D. Weis et al., “A user profile-based personalization system for digital multimedia content.”, Proceedings of the 3rd Intl. Conf. on Digital Interactive Media in Entertainment and Arts, 2008, pp. 281-288, publisher: ACM.

[11] World media televisions per capita (Factbook), available at http://www.nationmaster.com/graph/med_tel_percap-media-televisions-per-capita, retrieved: July, 2013.

 ISBN: 978-1-61208-289-9