Telebuddies on the move: social stitching to enhance the networked gaming experience

Kris Luyten Kristof Thys Steven Huypens Karin Coninx
Hasselt University – transNationale Universiteit Limburg
Expertise Center for Digital Media - IBBT
Wetenschapspark 2, Diepenbeek, Belgium
{kris.luyten, kristof.thys, steven.huypens, karin.coninx}@uhasselt.be

ABSTRACT

In this paper we report on our work to enable "laid-back" social interactions using television as a primary interaction medium and mobile devices that participate as a secondary medium. By integrating semantic web techniques with interactive television we were able to create smart applications that can run as extensions of television shows and stimulate groups of users to communicate. Participants are teamed up to play along with a television show. Teams are composed based on common ground, which can be shared interests as well as shared characteristics that can be found between the users. Participating is possible using a set-top box or a mobile device; this allows people that are co-located and share a television set to participate individually. The usage of a mobile device to participate does not require a television set however: users on the move can also play along in a team. Our system does not require a new television format, but is also able to reuse existing television shows and socialize them in order to be re-broadcasted with support for distributed teams that play along with the show.

Keywords

Interactive television, social games, semantic web, cooperative user interfaces

1. INTRODUCTION

With networked interactive television rapidly gaining ground worldwide because of more powerful set-top boxes, new challenges arise to exploit this immense network of users. Interactive Digital Television is becoming a widely spread medium that enriches everyday lives of millions of people. Networking is already an ubiquitous component of set-top boxes that are currently used and open up an opportunity to support interaction of television viewers with mobile users. People can communicate with each other using this infrastructure in a relaxed and comfortable way, regardless of the location of the user. For example, a recent strategic document of the Flemish minister of media[1], states in Flanders (Belgium) 97% of the inhabitants have access to the cable network, and the majority of inhabitants own a mobile phone. The development of interactive television is mentioned as one of the important new developments for Flanders in this strategic document. With the advances in mobile telephony (e.g. growing availability of 3G network services) and the introduction of interactive television in 2005, there is an infrastructure available to interconnect devices and people based upon a common denominator: the television show they are watching.

Although creating new television formats that explicitly use the new technologies and support viewer interaction is necessary to exploit all possibilities, we show many existing "traditional" formats are suitable for group interaction and can even be used in another context. Existing television formats are more accessible for the wide range of viewers with different backgrounds and standard television sets; in general it is much cheaper to use existing material. A television show that has already been broadcasted can be broadcasted again easily without any extra production costs. Even in case the producers want to add viewer interaction to an existing television show, it will be cheaper to rely on existing production methods and add interaction afterwards in comparison with using a new production method including interaction.

The Telebuddies system presented in this paper allows annotating existing television shows and turning them into interactive shows [5]. We extended the system so participation of mobile users is also supported. We will also delve deeper into the way participating teams are created in the Telebuddies system. Telebuddies exploits similarities and social relations between spectators to make the interactive television experience more exciting and fun. An important component is deciding on how to divide the participants in different groups or teams, where the size of each group depends on the type of game that is played. Typically a television show has a large audience, in this case the iDTV users and the mobile participants. To create an optimal gaming experience, relatively small groups of related people are created by an algorithm based on common ground. We rely on several techniques often used to build semantic web services [2, 13].

The remainder of this paper is structured as follows: We

[1] "Beleidsnota 2004–2009", minister Geert Bourgeois, `http://www.geertbourgeois.be/documenten/media.pdf`

start out in section 2 with a use case describing the system using a concrete example. Section 3 explains the core of Telebuddies that supports a combination of a traditional telvision show, iDTV and mobile gaming. Section 4 describes how semantic web techniques allow creating social experience in multiplayer games. Section 5 elaborates on the technical aspects of the Telebuddies system. Finally, section 6 discusses the future work and section 7 gives the conclusions.

2. CASE STUDY

In this section we describe how the system works using a specific example. "Test the Nation" or in Dutch "De Nationale Test", was broadcasted on Flemish television in September 2004. It is a television show exploiting groups of people with shared characteristics to compete against each other.

On the website[2] of the show we can find the following description:

> [...] Test the nation is the world's largest simultaneous I.Q. test. After watching the show every participant will know his or her I.Q. score. In the studio, groups of participants will include: Celebrities, Blondes, Garbage Men, Unemployed, Lawyers... At the end of the show, the different groups in the studio will be compared, as well as different states, republicans vs. democrats, etc. [...]

With emerging interactive television technologies, the quiz experience will not be limited to participants inside the studio. Spectators watching the show can play along. In contrast with existing interactive television games, the spectators play together with other spectators, just as the participants in the studio compete or collaborate with each other.

While the quiz television show is being broadcast, mobile and iDTV users can play along using the Telebuddies system which is synchronized creating an integrated experience. When the television show starts, everyone is invited to sign up and participate. When the broadcaster signals the game to start, groups of participants are created, based on their user profiles and using the techniques described later. Spectators are moved into a chat room where they can communicate with other members of the same group. The maximum and minimum number of members in one group depends on the television show. Members in the same group have profiles with shared characteristics. In this case a team could be composed out of members that fulfill three criteria; live in the same village, like to play computer games and are students.

When the participants in the studio are presented with a question, the same question is sent to all interactive users. Interactive television users receive the question being shown in overlay of the original TV-image (Figure 1), while mobile users are presented with the interfaces shown in figure 2(a) and figure 2(b) depending on their ability to follow the

[2] http://www.bbc.co.uk/testthenation/

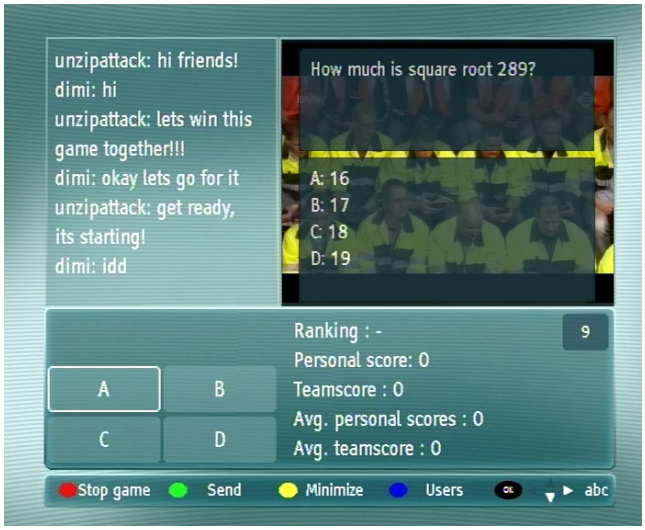


Figure 1: Question shown in overlay with the television screen. The viewer's input is restricted to answering the question.

television show. The user can answer the question, and between questions it is possible to communicate with other group members.

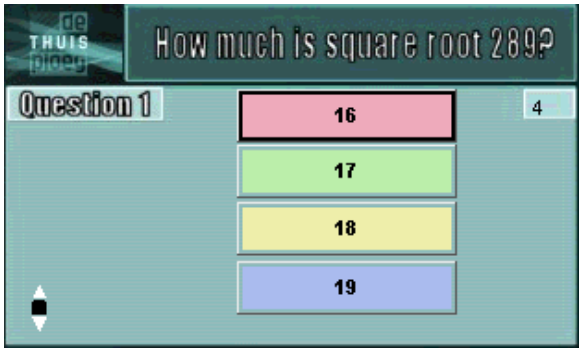
Afterwards, an evaluation of the quiz is shown, winners and losers are proclaimed and different viewer groups are compared against each other. In following shows, the same groups can be reused, extended or reduced. This gives the possibility to form micro-communities of viewers that often play together and compete against other micro-communities.

3. COMBINING IDTV AND MOBILITY

The approach presented here starts with a recorded television show and adds viewer interaction to this show, but live shows can be handled as well. For this purpose we create an *interaction script* that includes different types of events that require interaction with the viewer on different timestamps. An interaction script is a regular file that is loaded by our system when the television show is broadcasted; the script will be executed simultaneously. Depending on the type of television show, this interaction script can have a different vocabulary since questions can have different formats. We use a typical television quiz that can be turned into an interactive quiz as a proof-of-concept for our approach. This means our interaction scenario has to allow the viewer to participate in the quiz and basically answer questions within the appropriate time spans. When the participants in the studio have to answer a question, solve a riddle... the participant gets the same time to find an answer. In addition, since many quizzes incorporate teams that compete with each other the Telebuddies system also supports collaboration of different remote viewers and mobile users as team members in the quiz.

Each question in the interaction scenario is timestamped and has a predefined duration so our software can present the question to the viewer at the required time and the viewer

(a) On a small mobile device that acts as a remote, only the part the users needs to interact shown. This interface is used in combination with a television on which additional required information is shown.



(b) On a larger mobile device that is being used "on the move", people can participate without using a television. All required information to interact is displayed in the user interface.

Figure 2: Interface on mobile device

has a limited time period to respond. The interaction scenario that enriches an existing television format is the core of our system. Depending on the format, the interaction scenario could incorporate elements that are defined in an ontology associated with the format. The usage of ontologies to annotate formats provides us with a powerful tool to add new types of functionality to existing formats. A format is written down as a list of entries in XML syntax, where each entry has a timestamp and an action. An action in our example is also specified using XML and describes a question, time limit to answer the question, the possible answers and in case of a multiple choice question an identification of the correct answer. If required by the television format, the influence on score when the question was answered is also included in the description. An action is interpreted on the server and sent to the client devices. These actions are synchronized with the television show by the timestamps. It is important to present the question to all participants in a timely fashion. Since there are heterogeneous networks involved with a large number of participants, there is a need for a reliable and performant networking solution. Although optimizing network performance is not the scope of this paper, we will provide more detail in section 5 about the setup we use to minimize the network load.

Figure 1 shows the iDTV interface for the viewer when a question is retrieved from the interaction scenario and presented during the television show. The question is shown in overlay with the normal television image to create an immersive user experience. When a question is presented, the user input is constrained according to the type of answer that is expected and the focus of the user interface is automatically shifted to the part of the user interface presenting the input possibilities. For example, figure 1 shows at the bottom left that the user can only use "A", "B", "C" and "D" during the time the question is shown on screen. Alternatively the input can automatically be constrained to free form text, dates, numbers... according to the answer the user has to give. The available time to provide an answer is also limited and indicated in the user interface.

A similar approach is used to allow users with mobile devices[3] to participate in the game. There are two different possibilities: one can use the mobile device as a remote control while watching the show on television (figure 2(a)) or the mobile device can be used to participate at the same time without watching the show (figure 2(b)). The former is a necessary enhancement if the same television set is being watched by multiple persons (e.g. a family): since there is only one remote control only one viewer can participate using the set-top box, other viewers (family members) can now also participate using their mobile devices. The latter enables mobile participation without the need for a television signal. Section 5 shows that traditional Internet connections can be used, making the system available to a wide range of mobile devices and it makes it virtually location independent.

4. FOAF FOR COMMON GROUND

Most interactive television games are only suited for individual participation, although a lot of studio games are played in team. In a networked environment, the challenge is to divide all viewers in groups that have the required number of participants according to the game rules. Furthermore, a team should be composed with care: people cannot even begin to coordinate on content, without assuming a vast amount of shared information or common ground, e.g. mutual knowledge [4, 10]. With the magnitude of users that collaborate and play together in this networked system, an

[3]We consider the mobile devices unable to receive television signal, for now mobile television standards are not included in this approach

automatic smart way for setting up different teams of players is invaluable. In this section we present a way to compose a team based on matching characteristics of the participants.

Using semantic profiles of participants enables us to find shared characteristics that can be used to define the teams. In combination with the context of the television show this results in teams with people that share the most appropriate common ground for that specific television show: e.g. the show Family Feud[4] implies family connections between participants in the same team. Spectators can be subdivided in smaller groups by incrementally adding more required characteristics if required. If the maximum amount of shared characteristics is already used to divide the teams and if they are still too big according to the game requirements, a random subdivision can be used to obtain a smaller size.

We use known techniques originated from the semantic web initiative as proposed by Tim Berners-Lee [2], in order to both enrich the interactive experience as well as the social game experience for the user. A Telebuddies user is identified by a Friend-Of-A-Friend (FOAF, [3]) profile. This profile contains information about the social relations of the user next to traditional information that identifies the user (such as name, address, hobbies, hair color...). Listing 1 gives an example of a FOAF profile that is used by the Telebuddies system. We use the del.icio.us social bookmarking tags[5] as a folksonomy [7] that identifies the interests of the participants. This example also shows that the FOAF profile is based on a Resource Description Framework (RDF[6]) vocabulary for describing people and their social networks.

Listing 1: FOAF profile of a telebuddies user

```
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/"
        ...>
 <foaf:PersonalProfileDocument rdf:about="">
   <foaf:maker rdf:nodeID="me"/>
   <foaf:primaryTopic rdf:nodeID="me"/>
 </foaf:PersonalProfileDocument>
 <foaf:Person rdf:nodeID="me">
   <foaf:name>Kristel Clijsters</foaf:name>
   <foaf:title>Miss</foaf:title>
   <foaf:interest>
<rdf:Class
 rdf:about="http://del.icio.us/tag/Puppies"/>
<rdf:Class
 rdf:about="http://del.icio.us/tag/winniethepooh"/>
  ...
<rdf:Class
 rdf:about="http://del.icio.us/tag/clouseau"/>
<rdf:Class
 rdf:about="http://del.icio.us/tag/skydiving"/>
   </foaf:interest>
   <swn:yearOfBirth>1979</swn:yearOfBirth>
   <foaf:birthday>22-11</foaf:birthday>
   <vCard:ADR rdf:parseType="Resource">
     <vCard:Locality>
       Borgloon
     </vCard:Locality>
     <vCard:Pcode>3840</vCard:Pcode>
     <vCard:Street>
       De Tieckenstraat
     </vCard:Street>
   </vCard:ADR>
   <foaf:givenname>Kristel</foaf:givenname>
   <foaf:knows>
     <foaf:Person>
       <foaf:name>Kris Luyten</foaf:name>
       <rdfs:seeAlso
rdf:resource="http://research.edm.uhasselt.be/\
             kris/kris.foaf.rdf"/>
     </foaf:Person>
   </foaf:knows>
   <foaf:knows>...</foaf:knows>
  ...
</rdf:RDF>
```

The use of RDF makes these profiles machine processable. The RDF syntax allows us to query for relations between different FOAF files and build a graph of related FOAF profiles. The RDF Primer [6] states:

> "RDF is intended for situations in which this information needs to be processed by applications, rather than being only displayed to people. RDF provides a common framework for expressing this information so it can be exchanged between applications without loss of meaning. [...] The ability to exchange information between different applications means that the information may be made available to applications other than those for which it was originally created."

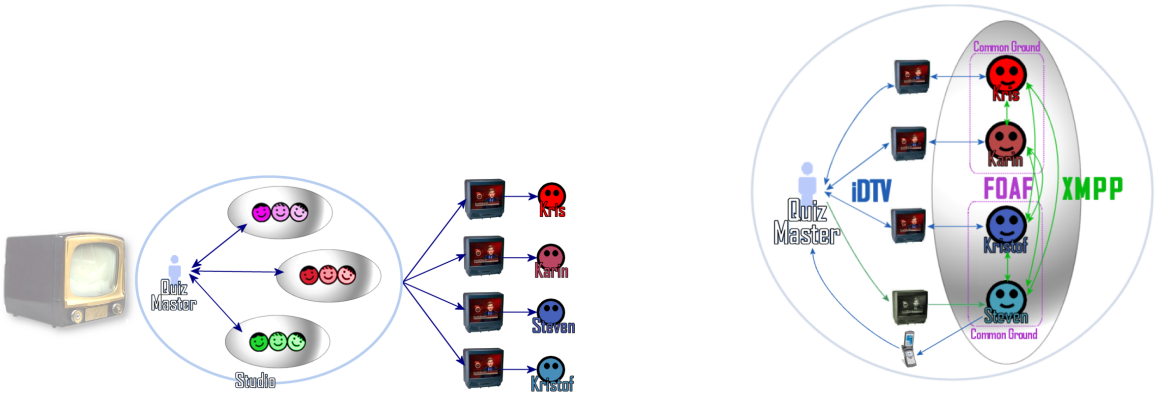
A complete overview of the RDF language falls outside the scope of this paper.

We rely on all viewers having a FOAF profile available somewhere on the Internet. If this is not the case, a simple form-based interface will allow them to create a FOAF profile and store it on the Telebuddies server. We can use these profiles to find a common ground between the viewers and create viewer subgroups or teams accordingly. Examples of a common ground are whether people exercise in the same sports club, are family of each other, live in the same street or have common friends. Since we can exploit arbitrary relationships between different FOAF profiles because of the RDF-based syntax, the type of relationships and properties that are used to determine the common ground between different viewers is constrained by the type of television show. For example, we mentioned the quiz Family Feud requires the participants in one group to have a family connection. Another show broadcasted in the Flemish part of Belgium is "De Nationale Test" where there are four teams of about thirty people in the studio and each team has some particular characteristics (like the same profession or hobby). Using the profile matching facility presented in this section, similar teams of spectators can be composed and they can play along with the show as if they were a team in the stu-

[4] http://www.familyfeud.tv/

[5] http://del.icio.us/ allows users to share and tag bookmarks online. The tag collection is accessible via http://del.icio.us/tag/

[6] http://www.w3.org/RDF/



(a) Traditional situation

(b) When the Telebuddies system is used

Figure 3: Before and with the Telebuddies System

dio. Section 2 describe how we used this show as our test case.

Our implementation uses the Jena[7] library [8] to query the FOAF profiles of the viewers. We build an RDF graph that contains the FOAF profiles as nodes and relationships with other FOAF profiles as edges. To query this graph we use RDQL [13], a query language for querying RDF graphs. RDQL allows us to select a number of FOAF profiles that fulfill a set of predefined criteria and are related with each other (e.g. have a family connection). Each television show has a configuration file that defines which characteristics are important to match for that particular show. Listing 2 shows a part of such a configuration file: it specifies two competing teams of exactly 5 persons, where one team needs to be composed out of males born between 1980 and 1987 and the other team needs to be composed out of people living in the same town. Figure 4 shows a set of FOAF profiles and indicates how they can be grouped together according to the characteristics one chooses to match with, which is of course dependent of the type of game show.

Listing 2: A television-show specific configuration file that defines the types of teams that should be created.

```
...
<min>5</min>
<max>5</max>
<competingteams>2</competingteams>
<criterium groupName="Mannekes">
  <ns var="z">
    http://purl.org/net/swn#yearOfBirth
  </ns>
  <critRule>
    ?z &gt;= 1980, ?z &lt;= 1987
  </critRule>
  <ns var="gender">
    http://xmlns.com/foaf/0.1/gender
  </ns>
  <critRule>?gender EQ "male"</critRule>
</criterium>
```

[7] http://jena.sourceforge.net

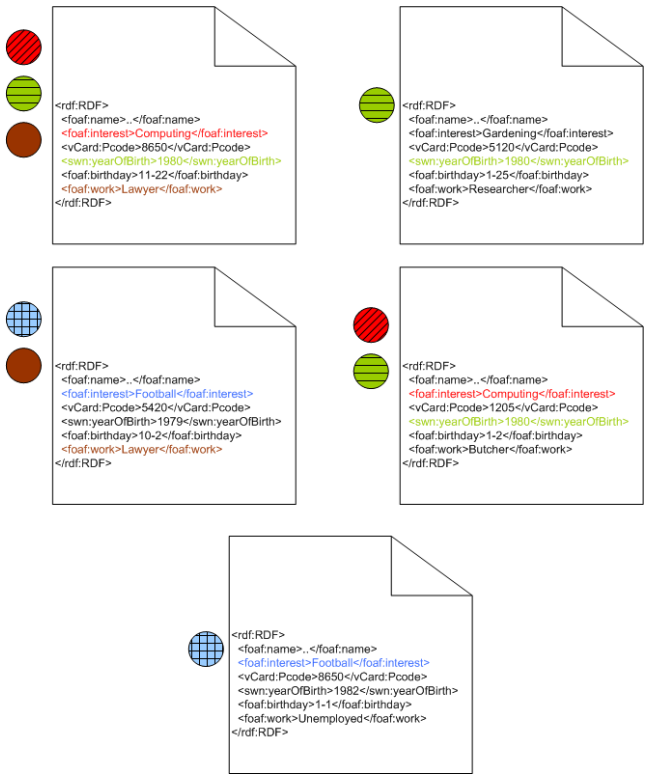


Figure 4: A set of profiles and candidate groups.

```
<criterium groupName="Hasselaren">
  <ns var="z">
    http://www.w3.org/2001/vcard-rdf/3.0#Pcode
  </ns>
  <critRule>?z ~~ /^3500/</critRule>
</criterium>
...
```

We are also aware that the use of such profiles requires a system to protect the privacy of a user. Currently, our implementation does not include security and privacy management except for a user login. The user is free to fill in the data she/he wants to expose.

5. SYSTEM SETUP

The system we describe is a client-server system with two types of clients: set-top boxes and mobile devices. The iDTV client application that runs on a set-top box is implemented on top of the MHP platform [9], a commonly used software platform for iDTV set-top boxes. Since the software platforms for iDTV set-top boxes are in constant evolution right now, it is hard to select the most appropriate platform for developing iDTV applications. MHP can be considered as the smallest subset of available Java-based programming libraries for the set-top box, so the application should be easy to port to other set-top boxes that offer a more extensive platform for deployment. We also chose MHP because it is a widespread and standardized software platform.

The mobile application is implemented on the J2ME platform using Mobile Information Device Profile version 2 (MIDP 2.0[8]). Nowadays, the majority of smartphones support this Java profile. In combination with the Connected Limited Device Configuration (CLDC[9]) complex mobile applications can be built that allow using IP-based network communication, a requirement for the system we describe in this paper.

Figure 5 gives an overview of the Telebuddies overall architecture. The Telebuddies system synchronizes with the broadcasts using timestamps to ensure timely delivery of questions. For both data communication between the Telebuddies system and the client devices and chat communication between participants we use the XMPP protocol [1], which can be considered as a "ubiquitous" protocol because it does not require anything from the client except being network-enabled and to be able to process XML-structured data. Notice no streaming media is distributed using XMPP: the game information (questions etc.) are delivered separately from the television signal since mobile devices that have no capability to present a television signal can also participate. A more reliable synchronization mechanism is required if the system would be deployed (e.g. synchronizing with the NTP timestamps in the MPEG-2 stream that contains the television content). Also we are aware that sending the interaction script to the client in advance implies a possible security issue. However we focused on the human-computer interaction and social computing side in this paper rather then investigating the network synchronization and security issues.

In addition to the traditional broadcasting service, figure 5 shows there are also two separate services provided by the Telebuddies system. One service is responsible for scenario execution (i.e. synchronization with the "application logic" of the specific television show) and another one to process the FOAF profiles and create participant groups. For processing purposes the FOAF profiles are collected and stored on a single server system. Afterwards the set of profiles is processed on a central system: the overhead of querying all the FOAF profiles over a network would be too high and processing time would be unreasonably long.

6. FUTURE WORK

Before the Telebuddies system can be widely deployed, some issues need to be resolved. First the scalability of the system has not been tested thoroughly yet. Because of the immense number of viewers some shows have (up to millions), the system needs to be extremely scalable. In particular the throughput of the system should be optimized: serving millions of interacting clients with time-dependent data over heterogeneous networks is still an open problem. With the current trend of massive multi-player online role playing games that also need to support a large amount of users and require a throughput that allows users to interact with each other, it is likely solutions for the scalability problem will come from that side [11, 12].

In our future work we intend to investigate the synchronization issues into more detail. The use of multicasting could solve most synchronization issues. With the newly available multicasting for mobile devices using 3G services, there might be less effort necessary to make sure all devices are synchronized. We are already cooperating with networking experts in the Flemish IBBT research project "Multimedia Content Distribution Platform"[10], where interaction and synchronization for iDTV and mobile clients is one of the main research topics.

Although we strongly believe the Telebuddies system lowers the threshold of participating in online multiplayer games, this statement still needs to be verified.

Another issue that needs to be resolved as part of our future work is to find new, safe and convenient ways to gather social metadata of different persons: the usefulness of the FOAF profiles for finding a common ground depends on this. A last issue related with the previous one is sustaining a suitable privacy policy to store, maintain and process the FOAF profiles.

7. CONCLUSIONS

In this paper, a system is presented that combines traditional television shows, iDTV, social networking and mobile devices. We believe this system lowers the threshold to participate in online multiplayer games by creating a social experience on television.

People playing together through television and mobile de-

[8]MIDP 2.0 specification is available at `http://jcp.org/aboutJava/communityprocess/final/jsr118/index.html`

[9]`http://java.sun.com/products/cldc/`

[10]`https://projects.ibbt.be/mcdp/`

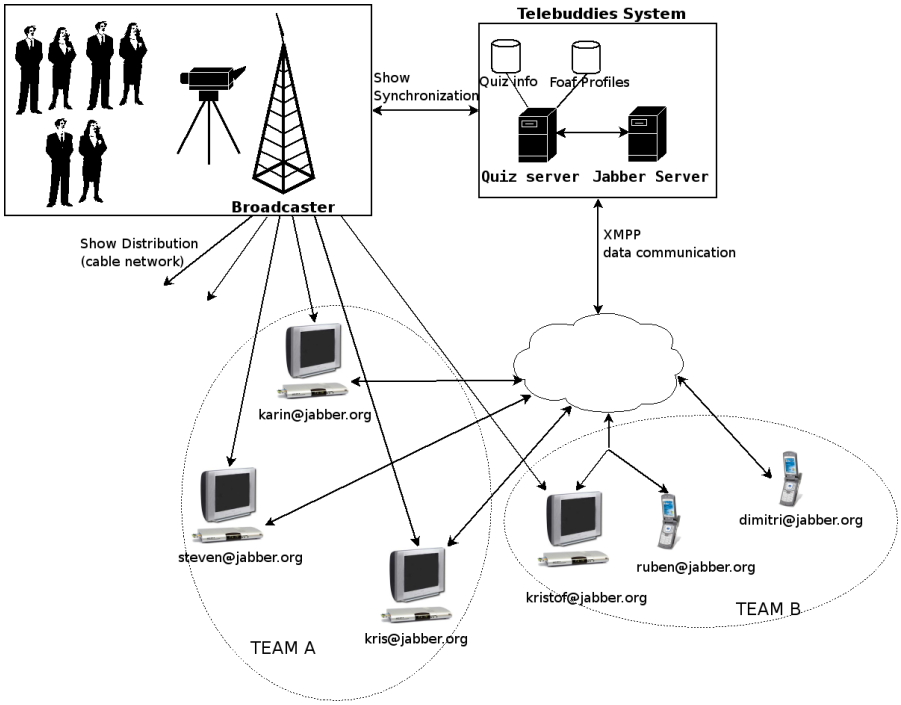


Figure 5: Telebuddies network architecture overview. The figure shows two competing groups. On of the groups has two mobile devices that participate: one device that is used "on the move" (dimitri@jabber.org), another one (ruben@jabber.org) is used in the same room with a television user (kristof@jabber.org).

vices are carefully selected by using their user profile and existing social relationships with other participants. This can greatly enhance the gaming experience and provide a more accessible way to use participative interactive television. Although our system has not yet been tested in real-life settings, the examples discussed in this paper are fully supported by our system and tested in lab settings. The functionality was tested with smaller groups of up to 6 persons playing together in a team or competing against each other in separate teams.

In conclusion we believe the system we presented turns an individual activity such as watching a television show into an interactive social experience. The threshold to participate is lower than in conventional collaborative approaches since everyone in a team will share the same characteristics. Semantic web technologies help us to accomplish this: with the Friend-of-a-Friend profiles, a network of user profiles that include a description of their social relationships is available. Additional material, including screenshots, are provided on the following website:
`http://research.edm.uhasselt.be/kris/research/projects/telebuddies/`

Acknowledgments

The authors would like to thank Tim Dupont, Dimitri Plaisier, Peter Quax, Daniël Teunkens, Ruben Thys and Koen Van den Bergh for their valuable contributions. Part of the research at EDM is funded by EFRO (European Fund for Regional Development) and the Flemish Government.

8. REFERENCES

[1] D.J. Adams. *Programming Jabber*. O'Reilly, 2002.

[2] Tim Berners-Lee. *Weaving the Web*. Harper San Francisco, 2002.

[3] Dan Brickley and Libby Miller. *FOAF Vocabulary Specification*. World Wide Web, 2005. `http://xmlns.com/foaf/0.1/`.

[4] Herbert H. Clark and Susan E. Brennan. *Perspectives on Socially Shared Cognition*, chapter Grounding in Communication, pages 127–149. American Psychological Association, 1993.

[5] Kris Luyten, Kristof Thys, Steven Huypens, and Karin Coninx. Telebuddies: social stitching with interactive television. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1049–1054, New York, NY, USA, 2006. ACM Press.

[6] Frank Manola and Eric Miller. *RDF Primer*. World Wide Web, `http://www.w3.org/TR/rdf-primer/`, 2004.

[7] Adam Mathes. Folksonomies: Cooperative Classification and Communication Through Metadata. *World Wide Web, http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html*, 2004.

[8] Brian McBride. Jena: Implementing the RDF Model and Syntax Specification. In *Semantic Web Workshop, WWW2001*, 2001.

[9] Steven Morris and Anthony Smith-Chaigneau. *Interactive TV Standards*. Elsevier, 2005.

[10] Jeremy P. Birnholtz, Thomas A. Finholt, Daniel B. Horn, and Sung Joo Bae. Grounding needs: achieving common ground via lightweight chat in large, distributed, ad-hoc groups. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 21–30, New York, USA, 2005. ACM Press.

[11] Peter Quax, Tom Jehaes, Pieter Jorissen, and Wim Lamotte. A multi-user framework supporting video-based avatars. In *Proceedings of the 2nd workshop on Network and system support for games*, pages 137–147. ACM Press, 2003.

[12] Peter Quax, Patrick Monsieurs, Tom Jehaes, and Wim Lamotte. Using autonomous avatars to simulate a large-scale multi-user networked virtual environment. In *Proceedings of the 2004 International Conference on Virtual-Reality Continuum and its Applications in Industry (VRCAI2004)*, pages 88–94. ACM Press, 2004.

[13] Andy Seaborne. *RDQL – A Query Language for RDF*. World Wide Web, 2004. `http://www.w3.org/Submission/RDQL/`.