REWARD A Real World Achievement and Record Database

Tonio Triebel, Sascha Schnaufer, Benjamin Guthier ,Hendrik Lemelson, Gregor Schiele, Wolfgang Effelsberg
University of Mannheim
Email: {triebel, schnaufer, guthier, schiele, lemelson, effelsberg}@informatik.uni-mannheim.de

Abstract—We propose a system for acquiring, maintaining, and communicating Real-World Achievements. The concept is based on virtual world systems that are in common use in current Massively Multiplayer Online Games (MMOG) like World of Warcraft or Warhammer Online. The system design is centered around a global database of achievement definitions and individual progress.

I. INTRODUCTION

The vision of Pervasive Computing has gained widespread attention in the research community. Much research has been done and many Pervasive Computing systems have been developed. What is still missing are convincing large scale applications, so-called 'killer applications', that persuade normal end users to use Pervasive Computing technology in their daily life. We propose to use a *global real world achievement system* as one such application.

Achievement systems originate from online entertainment systems like Microsoft's XBox Live or Blizzard's World of Warcraft. The intention is to trace a user's deeds in the virtual gaming world and to reward him with records in his profile. These records are called *achievements*. Whenever a user obtains an achievement, a notification message is sent to people in the same context as the user. Recipients of a notification could be friends or other users in the vicinity.

The introduction of these achievements had a significant impact on the users' gaming behavior. Attaining as many achievements as possible or attaining a particularly challenging achievement has become a new motivation. Some users are even willing to invest their playing time into relatively useless or boring tasks, simply to get the achievements associated with them. This impact is based on the fact that people are used to such rewards in the real world. Awarding people for their efforts is common practice in our society: pupils get graded, scientists obtain a doctor's degree, and in the business world people use titles to denote position (CEO, vice president, team leader or professional achievement plaques hang in people's offices).

We argue that the emergence of a pervasive communication infrastructure makes a real-world achievement system more and more feasible. We thus propose a global system for acquisition, maintenance and communication of achievements in the real world.

II. SYSTEM DESIGN

In order to implement a Real-World achievement system we developed a simple system architecture. See Figure 1 for a schematic drawing. The individual parts are described in detail in the following Subsections.

A user's effort to obtain an achievement is registered by the *Event Service*. This can be a software installed on either a user's mobile device or a stationary terminal, as described later. The Event Service transmits user events to the central *Awarding Server*, which is connected to the *Achievement Database*. The database contains information about all possible achievements as well as individual progress and achievement histories. This information is used by the Awarding Server to check if an achievement was obtained. If this is the case, the obtained achievement is recorded in the database. Then a list of users that are potentially interested in the particular achievement is created, and a notification flag is set in their profiles. An Ajax-driven web application is responsible for *Presentation and Notification* of acquired achievements. It polls the database for new achievements and displays them to logged-in users in near real-time or asynchronously at a later time.

A. Achievement Database

The central component of our system is the Achievement Database. It mainly stores two classes of information: user-specific data and achievement definitions. The user-specific data contains various profile information, a friend list, an achievement history and notification flags. Additionally, a number of personal *states* is saved. In most cases, these are simple integer numbers representing achievement-related counters, like "user has been to Paris once" or "user performed x charitable activities". The types of states that are recorded are listed in the global achievement definitions part of the database. All possible achievements are categorized and saved, together with the achievement issuer, states related to the achievement and a number of conditions that must be met in order to obtain it. As an example, a soft drink producing company could issue an achievement for having tasted each of their products at least once. Associated states could be counters of soft drink purchases, and the condition could be $counter \geq 1$.
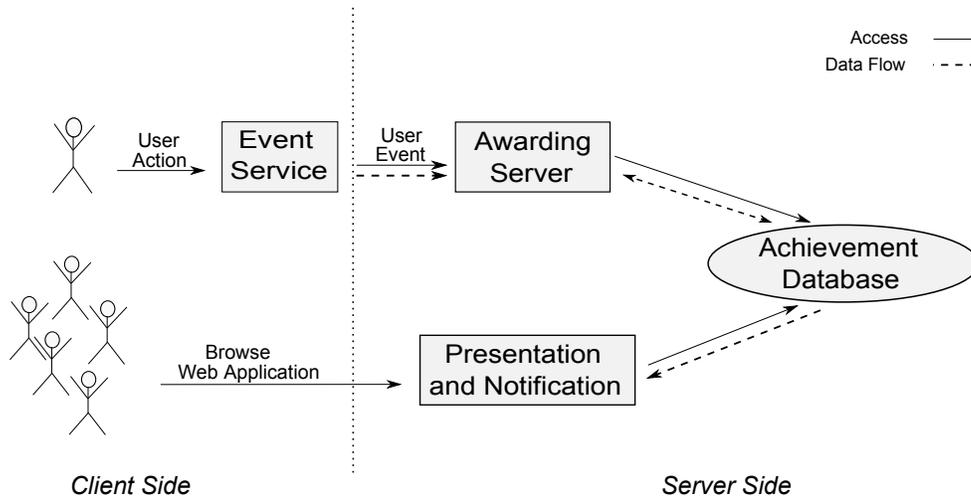
Figure 1. Overview of the design of our real-world achievement system. The solid lines denote initiated access, while the dashed lines represent the flow of information.

B. Event Service

The Event Service is a software running on either a user's handheld device or on a stationary terminal. Its role is to keep track of user actions and to generate user events and transmit them to the Awarding Server.

User Initiated Events are events that originate from software running on a user's handheld device. A user interested in progressing towards a particular achievement, needs to have it running while doing so. The software monitors the built-in sensors like camera, WLAN, GPS or acceleration sensor. Using their output, it can for example detect a user's position and send a "User is in Asia" event to the Awarding Server to progress the globetrotter achievement. Another example could be a 2D barcode containing a secure ID, put up at the entrance of Neuschwanstein Castle. A user scans the barcode using the mobile phone's camera. The Event System then informs the Awarding Server of the user's sightseeing progress.

Third Party Initiated Events are triggered by software running on a terminal owned by a trusted third party. It can be done automatically or manually by a human operator. The latter is the preferred method of event creation in situations where protection against cheating is an issue, or user initiated approches are unfeasible. In this case, the Event System only needs to be installed on selected terminals, rather than on each user's device. It is also possible to use a simple website to enter achievement progress manually. A use case for third-party initiated events could be tracking world record attempts or personal achievements during sporting events. In a more trivial scenario, students could authenticate themselves automatically at an RFID reader inside a lecture room to make progress towards achievements related to lecture attendance.

C. Awarding Server

The Awarding Server is directly connected to the database. It provides the application logic for the entire system. Whenever a user event occurs, the Awarding Server is informed. It performs a mapping between user events and related states according to the achievement definitions stored in the database. The updated personal states are written to the database, and associated achievement conditions are tested. As soon as all the conditions are met, the user is awarded with an achievement. This results in an entry in the user's achievement history. At the same time, a notification list consisting of persons who are potentially interested in the achievement is created. For this purpose, aspects like profiles, friend lists and achievement histories of other users are taken into account. A notification flag is then set in the recipients' database entries. We believe that notifying relevant others about obtained achievements is an important aspect of the motivation achievement systems bring forth.

D. Presentation and Notification

The component that implements the Presentation and Notification of acquired achievements can be located at the same site as the Awarding Server. It is a passive application querying the contents of the database and transforming them into a suitable display format. It can be accessed through a regular internet browser, either from home or in a mobile fashion. A user interested in receiving achievement notifications logs in to the web application. The scripts on the website poll the database for personal notification flags in regular time intervals. Once a new notification arrives, it is displayed to the user. Browsing user profiles and achievement histories can be done in a manner similar to existing social networks.

III. Demonstration

In order to demonstrate the applicability of our approach, we implemented a prototype of the Real-World Achievement System and defined a list of achievements. The list can be seen in the appendix. The implementation is based on the Django web framework and uses an Apache web server and an SQLite database. The database entries consist of user account data such as name and ID, the achievement definitions and the users' achievement progress. As a third-party Event Service we used an RFID-based interface to identify a person at any location where achievements can be obtained. The RFID readers are connected to computers running the Event Service software. The software sends the achievement event and the corresponding user ID to the Awarding Server. The server updates the database and checks the achievement conditions. If a condition is met, the sever updates the user's profile. In order to present the achievements, we implemented an Ajax-driven web application that polls the achievement data from the database and displays the obtained achievements immediately. (*http://dione.informatik.uni-mannheim.de/demo/achievements/*)

A. Demonstration Setup

The demonstration consists of two parts. The first part is a trial phase for the whole duration of the PerCom 2010. The participiants are given the chance to experience the Real-World Achievement System. Achievement points and titles related to the conference can be collected. The second part takes place during the demo session. In addition to the in-depth demonstration of the system components, the results of the trial phase are evaluated and presented in real time.

1) Trial Phase of the Achievement System: The trial phase spans the entire duration of the conference. Volunteers are able to create an account for the achievement system during registration. Once activated, the account can be used to gather achievements across the conference area. As interface to the system, RFID tags are handed out. The tags can be attached to the conference badges and are used to identify the users at achievement-relevant locations. For instance if someone enters a lecture room he can read out his RFID tag using an installed RFID reader. The Event Service then generates an event to increment the counter of the number of talks visited and sends it to the Awarding Server. If any achievement condition is met, the Awarding Server updates the database. The result can be seen immediately on the achievement web application.

2) Demonstration of the Achievement System: The second part of the demonstration takes place in the demo session. The functionalities of the system components described earlier are presented in detail. Participants are able to explore the system and obtain exemplary achievements. New achievement types can be entered into the database. In addition to the third party initiated events used during the trial phase, different examples for user initiated events are given. For this purpose handheld devices with an installed Event Service are available.

In order to get an impression of the system's impact, a real-time evaluation of the current achievement progress of all users is performed. Statistics like user information, individual achievement progress and total number of obtained achievements are presented.

Appendix

The following is a subset of the *list of achievements* of the demonstration:

- PerCom Author Achievement: Authors and co-authors of PerCom publications will get a achievement during registration.
- Talks Visited Achievement: Visitor achievements are given to users having visited a certain number of talks (e.g. 5/10/20/all talks visited)
- Early Bird Achievement: Every user who uses the system before a certain time (e.g. 9am) gets the achievement.
- Earliest Bird Achievement: Every day, the first user of the system gets this achievement.
- Presentation Hopper Achievement: If a users participates in two talks of parallel sessions, he will get an achievement.
- Audience Achievement: Presenters of talks with a sufficiently large audience get an achievement.
- Group Achievements: Users with the same affiliation can obtain group achievements like "talks visited by group" or "early bird group".
- Last Man Standing Achievement: The last user who uses the system at the conference banquette will get an achievement.