SecondLife® as an Evaluation Platform for Multiagent
Systems Featuring Social Interactions

(Demo Paper)

Matthias Rehm, Peter Rosina
Multimedia Concepts and Applications
Faculty of Applied Informatics
University of Augsburg
Eichleitnerstr. 30
86159 Augsburg, Germany
{rehm|rosina}@informatik.uni-augsburg.de

ABSTRACT
In this paper we propose to use SecondLife® as an evaluation
environment for multiagent systems that model social group
dynamics and social interactions. To this end we developed
a control interface for NPCs in SecondLife® that allows for
running supervised or unsupervised long-term evaluations.
Thus, this platform offers the possibility to move evaluation
studies from the laboratory into a “natural” setting for the
participants. First tests show the potential for this kind of
evaluation.

Categories and Subject Descriptors
I.2.1 [Artificial Intelligence]: Applications and Expert
Systems; I.2.11 [Artificial Intelligence]: Distributed Arti-
ficial Intelligence; I.6.7 [Simulation and Modeling]: Sim-
ulation Support Systems

General Terms
Experimentation, Human factors

Keywords
SecondLife, evaluation, multiagent system

1. INTRODUCTION
SecondLife® (SL) represents the first massive 3D multi-
player platform that is not primarily concerned with gam-
ing but aims at establishing a general virtual meeting place.
Thus, every conceivable type of interaction is in principle
possible, be it buying or selling virtual or real goods, or be
it playing out as a real DJ in a virtual club. Central fea-
ture of SL is the use of virtual agents as interaction devices
which can either represent a real user (avatar) or can be
non-player characters (bots). Consequently, SL represents
a multiagent system where users in the form of avatars and
autonomous virtual agents can engage in social interactions.
This offers for the first time the opportunity to evaluate
multiagent system techniques in unconstrained tests with
an unlimited number of participants. Evaluation studies are
no longer confined to laboratory settings, but can be done
in the field i.e. in the “natural” environment of the users. To
exploit this possibility, we created a control architecture for
autonomous agents in SL and exemplified the general use of
this architecture with a simulation toolbox for social group
dynamics, which constitutes our first test case for multiagent
evaluation approaches in SL.

The control architecture had to integrate the following
components:

• Low-level behavior control: For animating an agent,
sending and receiving speech events, and for navigating
through the environment, SL provides an open source
client which was modified to handle the special needs
of coordinated verbal and nonverbal behavior.

• High-level behavior control: To abstract from the te-
dious work of controlling every parameter for the agent
in SL, an abstract control module was realized (Bot-
Control). The BotControl represents the interface be-
tween SL and the third-party application. It provides
the necessary control methods for agents in SL which
can be incorporated in arbitrary applications handling
the low-level behavior routines of the agents as well
as the event handling for SL events occurring in social
interactions like spatial behavior.

To test a multiagent social simulation in SL we relied on
previous work on the social group dynamics of agents ([5]).
Two additional components create this simulation:

• Behavior toolbox: The toolbox handles the agents’ be-
havior according to a set of theories from the social sci-
ences. Each theory can be plugged or unplugged to test
its applicability in the specific scenario and select the
optimal model. The toolbox has already proven suc-
cessful in simulating appropriate communicative be-
behavior.

• Chatterbot functionality (AIML): To realize believable
linguistic behavior, chatterbot functionality was inte-
grated into our system. To this end, a widely used
AIML based chatterbot program was extended to deal
with interaction categories from the behavior toolbox
as a pattern structuring mechanism.
A first pilot study was run to exemplify the potential of SecondLife® as an evaluation platform and is summarized below (Section 3).

2. CONTROLLING AN AGENT

Figure 1 (left) gives an overview of the different components, which were integrated into our system.

2.1 Low-level behavior control (SL-Client)

The communication with the SL-server is realized by a modified version of an open source library called libsecondlife. This modification was necessary to allow for more functionality in controlling the character on the one hand, and on the other hand in providing more information about the environment. The resulting SL-client serves as an interface, which handles events between the higher level control component and the SL-server.

2.2 High-level behavior control (BotControl)

BotControl represents the high-level behavior control that allows for more abstract specification of agent behavior. It serves as the control center for an agent allowing to monitor its interactions, to set interaction parameters, to connect control components, and to manually override the behavior components in order to control the agent by the user. An example of an interaction parameter is the spatial behavior of the agent. Following Hall’s ideas on proxemics [2], we define different spatial areas that trigger different behavior routines in the agent (see Figure 1 right). To this end, a scan radius can be defined that provides a threshold above which an agent does not react to other agents. Moreover, it is possible to define an area which triggers a follow event if the agent is currently in an interaction with a user.

2.3 Integrating AIML

Interaction in SecondLife® consists mainly of chat events between users. Consequently, it was necessary to endow our system with chat capabilities. As a means to this end we rely on a well-established AI technique, i.e. the Artificial Intelligence Markup Language (AIML), which was developed for the chatterbot A.L.I.C.E (e.g. [6]).

2.4 Integrating Behavior Models for Agents

To allow for testing different behavior models for multiagent interactions, BotControl provides an interface to transmit data about the agent’s perceptions to a given control module. To exemplify this feature a previously developed behavior toolbox was integrated [5]. It allows for either rapidly prototyping different models of social group dynamics like Congruity Theory [4] or Social Impact [3] in a given multiagent application or for using it as the central behavior control component. From the available theories, Interaction Process Analysis (IPA) [1] was chosen as a test case. IPA generates an interaction category for the agent at each turn. This category then has to be mapped to an appropriate observable behavior of the agent. In SecondLife®, the primary interaction modality is verbal, thus, AIML was augmented by IPA-based interaction categories. In principle, the agent could also react by pre-defined animations that accompany its utterances. But as this is rarely seen in the user, it was thus not taken into account for the pilot study.

3. PRELIMINARY RESULTS

A pilot study was run to show the feasibility of the approach and to test if users are interested at all in interacting with the agent. Two test runs were carried out in different areas of SecondLife®, both sparsely crowded to prevent the agent from annoying the users by continuously trying to interact with them. Test runs lasted for around 15 hours a day for seven days. The agent interacted with 39 users in this time, the average time for the interactions was 6 minutes 34 seconds. It is unclear how many users discovered that they were talking to an agent. 17 of the 39 users explicitly marked this fact. One-third of those continued their interaction and made some positive remarks and requested more information about the project. Another third finished the interaction shortly after the discovery (under two minutes) and the last third became hostile against the agent. It is unclear if the other 22 users did not discover that they were talking to an agent or if they did not care. A closer analysis of the log files for the verbal interactions might give some insights concerning this question. The pilot also showed some technical limitations. To sum up, the main goals of this approach have been reached. It was possible to attract many users to interact with the agent showing the potential of running unsupervised large scale evaluation studies in SecondLife®.

4. ACKNOWLEDGMENTS

The work described in this paper is partly funded by the German Research Foundation (DFG) under research grant RE 2619/2-1 (CUBE-G).

5. REFERENCES