Innovations through Bots on Second Life

The word "Bot" in context of Second Life encompasses a whole spectrum of possible software applications that serve to extend and complement Second Life. Familiar examples include dance bots, who receive commands to dance, AI bots, who are connected to Artificial Intelligence algorithms for tasks such as normal conversation, and theatre bots, as propagated by the Second Life division of the Virtual Shakespeare Initiative.

After a brief contextual introduction, this paper will elaborate on details of theatre bots, particularly InfoBots. This paper assumes that the audience has a general non-technical familiarity with Second Life.

What is a bot?

The standard definition of a "bot" is a non-player-character that depends on either external applications or "LSL" to run. "Non-player-character" is a word derived from the term "non-playable-character (NPC)" used often to describe the "background characters" in traditional games. The first should not be confused with the latter, as bots can be playable by another character or avatar. The main distinction is that a bot is self-sufficient without a one-to-one sentient person logged in to control the avatar.

Traditional Bots

As of March 2008, the usage of bots have already become prevalent on Second Life. Dance bots that either load an animation in their inventory or on a pose-ball often help populate virtual venues. Some bots attempt to emulate avatars. A bot can have its chat connected to external (or even inworld LSL) applications that make the chat appear intelligible in the form of a flavor or a mix of flavors of AI (Artificial Intelligence). The presence of a non-phantom bot affects the traffic of a region, and also the lag; however, not all bots are non-phantom. Although the phenomenon of traffic bots has caught the bulk of the tabloid media of the virtual world, traffic bots compose of only a small subset of a spectrum of possible bots.

Theatre Bots

The Second Life division of the Virtual Shakespeare Initiative, more commonly known as the "Second Life Shakespeare Company" or SLSC, plans to use several bot technologies that would help virtual theatre productions, and even extend the current notion of what virtual theatre in a global multiuser world can be.

The bots used for the SLSC can be broken down into two main groups: InfoBots and AvaBots. InfoBots are self-sufficient based on either received information or stored information. AvaBots, which may possess auto-reflexes, require input from either a technician or actor/director.

Both bots are generally phantom to take up fewer resources of the simulator, and would not affect the traffic of the region. Automatic rebaking and de-ruthing would ensure proper appearance in spite of lag.

- **simulBots** allow for the simultaneous real-time broadcast of 3d avatar movements across the grid. They are especially relevant for events anticipating a large audience, which a single sim, or even a 4-sim venue, cannot possibly support. They are the 3d analogue of simulCasts, which are 2D live broadcast of an event on a screen. Each simulBot represents an actor present in the main venue, and receives the corresponding motion and actions of the same actor. simulBots can potentially be used to create an "infinite theatre," a theatre not bound by size or location, when a large number of simulBots are used across the grid.
  - **Version 0.02:** We would be using a hybrid of external software and the existing "SL Voice" technology implemented on Second Life and bots for our prototype exhibit in an upcoming SLSC miniproduction. The bot movements would be transferred through an external network to the simulBots, but the voice chat from the main sim would be broadcast through "group" voice chat to the simulSims. The simulBots use a proprietary format similar to the RecBot format below, and would have a vestigial line that spawns the "talking" animations of a simulBot to seamlessly indicate which character is speaking. We plan to eventually be able to stream voice from each actor in the main venue directly to each simulBot.
- **RecBots**: Every single digital event can be recorded, similarly, every single digital movement or action can be recorded for later playback. RecBots playback a scene in live 3d from stored information captured from idealized keyframes of a live performance. In addition, the playback can be controlled, whether paused, skipped, or rewound, and individual components of a scene can be modified.
  - Format: RecBots read from an XML file with a specialized schema. Each line corresponds to either a time interval or a line in a playscript. The current format is RecBot Markup Language (RBML) v0.02, which has the following attributes:
    - ID: Line number or time interval identifier (note that this is non-unique, as several differnet avatars can move during the same interval or line number)
    - AvatarName: Name of the avatar the simulBot should emulate
    - AvatarKey: UUID of the avatar above
    - CoordSys:
      - Global: relative to global coordinates of a region (simulator)
      - Relative: relative to the zeros of a stage corner
      - UUID: move towards an object or avatar with UUID
    - Position: (x,y,z) or UUID in a coordinate system with respect to above
    - Load: the UUID (or structured array) of an asset to load
    - Stream: the starting identifier of the sound stream to accompany the line, 0 for none.
    - Tween: the method of interpolation for a simulBot to "seamlessly" move from line to line.
  - Modification of assets: The skin, shape, and clothing of each recBot can be customized, as can the assets used on the stage.

- o Playback Action:
  - ▪ Usual playback features of a traditional VCR such as rewind, fast-forward, skip (a scene), and pause.
  - ▪ 3d Playback Modification
    - A character being cut out or a new character introduced (see Manual Interface)
    - A line being cut out
    - Assets removed or reassigned. (Example: default line has characters sitting on a bench. Remove the bench, replacing it with a horse, for example.)
  - o Re-Record: the variations created through the "playback actions" can then be re-recorded to create new scenes.
  - o Manual Interface: This allows the user to directly key in and modify each attribute of the data format described above.
- UsherBot: an usher bot provides information and also serves as a 3d virtual tour guide.

AvaBots

- CrowdBots help establish the teeming atmosphere of a scene by providing automated avatars that have certain reflexes that appear to "live" in the background of a scene.
- IndivBot: an avatar at the main venue that represent the live actor or performer.

Second Life and also many third party applications developed for it are based on open source technologies. Bot Serialization is a technique to dynamically replace a bot that has crashed, as to make the disappearance seamless.

Conclusion

In summary, bots can be used for many constructive and innovative tasks to complement and extend Second Life, and even to create new things to use the Second Life platform for.