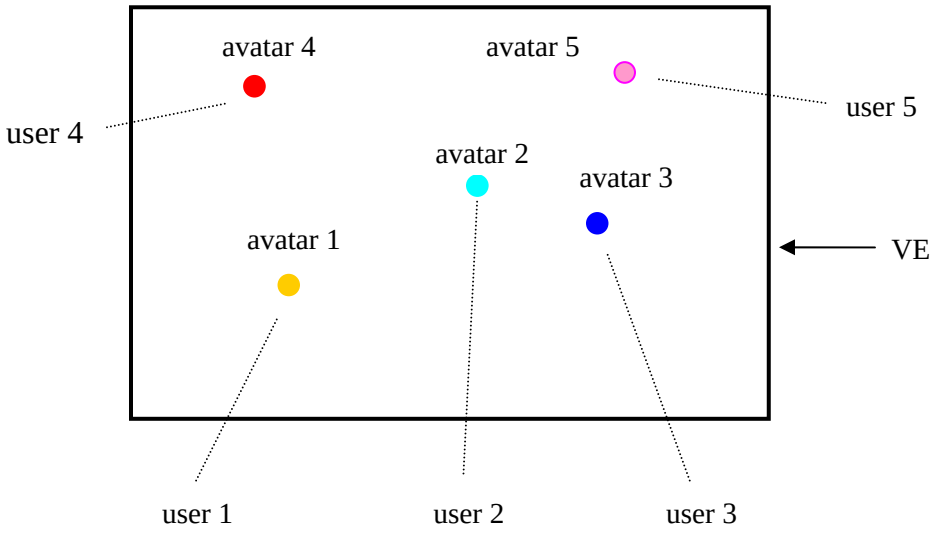
CS 457 – Winter 2009
Assignment 1
Due: January 30, 2009 at 5 pm

This assignment is concerned with performance evaluation of a multi-user online game system.

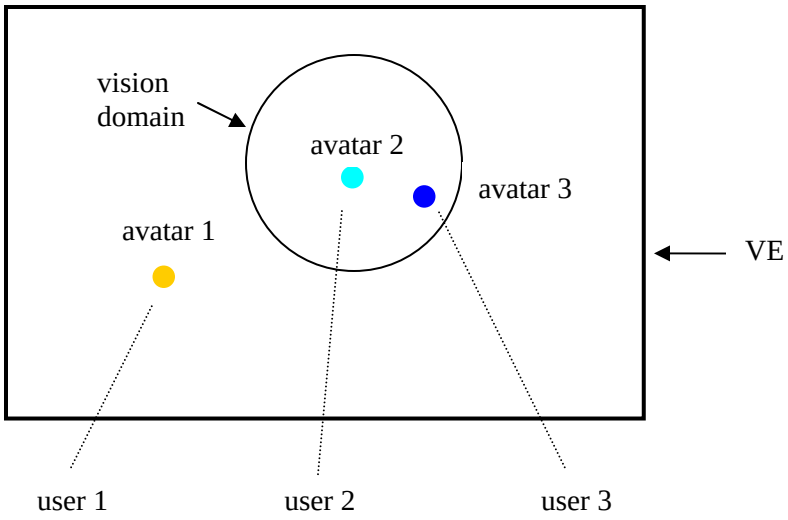
I. System Description

The system description in this section is at a level of detail required for the purpose of this assignment.

Multi-user online game is a shared virtual environment (VE) where multiple users at their workstations interact with each other over a network. Each user is represented by an avatar in the VE. See Figure below.



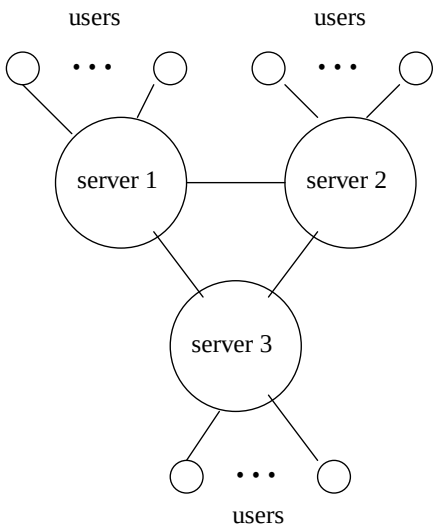
A user moves around, performs various actions, or interacts with the other users within the VE. These actions may result in changes in the location of the user in the VE. After a user has made a move, its new location should be seen by the other users as soon as possible to ensure realistic interaction. Distributing location updates to all other users can result in an excessive amount of resources required, especially when the number of users is large. To avoid this problem, a "vision domain" is defined for each user to describe the area in the VE where interactions between this user and the other users will likely take place. Then, a location update of a user is sent only to those users who are within its vision domain (referred to as the "affected users"). This will substantially reduce the number of updates to be distributed and thereby reduce the amount of resources required. As an example, a VE and the vision domain of user 2 are shown in the figure below. In this example, a location update submitted by user 2 is distributed to user 3, but not user 1.



A popular architecture for multi-user online game is client-server. In this architecture, all information regarding the VE is maintained by a central server. All users, at their workstations, interact with this server. When a user makes a move, the resulting location update is transmitted, in the form of an "update packet", to the server. At the server, when this packet is processed, the corresponding avatar is moved to its new location in the VE. The server also transmits the new location information, in the form of an update packet, to all affected users.

II. Proposed System Architecture

The client-server architecture is not scalable because the server becomes the bottleneck when the number of users is large. To address the issue of scalability, a two-level hierarchical architecture is proposed. At the lower level, multiple servers are deployed. An example with three servers is shown below.



Each user is assigned to a server; this is the user's "local server". Any other servers are referred to as "remote server" for this user. Each server maintains its own copy of the VE and interacts with its assigned users (or "local users") using the client-server model. At the higher level, the various servers communicate among themselves with the objective of sending location updates to all affected users as soon as possible and keeping their copies of the VE as consistent as possible.

A brief description of the system operation is as follows. At the user side, there is a DVE client program running on the workstation. This program accepts input commands from the user which may trigger a change in the location of his avatar. Any such change is transmitted, in the form of an update packet, to the user's local server for processing. When this update packet is processed, the local server applies changes to its copy of the VE and distributes the location update to the client programs of all affected users. The affected users may be located at the local server, or at one or more remote servers. In the former case, the local server transmits the update directly to the affected users. In the latter case, an update packet at the higher level of the hierarchy, referred to as a "syn packet", is transmitted to each of the remote servers. These remote servers, upon processing the syn packet, update their own copies of the VE, and distribute the update to their local users who are affected by the update.

For the two-level hierarchical architecture, servers can be added when there is a need to support more users. However, if a user makes a move and an affected user is at a remote server, then a syn packet must first be sent to the remote server, resulting in a longer delay.

III. Assignment

Suppose the goal of the study is to obtain results that would address the following issues for the proposed two-level hierarchical architecture:

1. How good is the proposed architecture in terms of scalability?
2. How consistent are the copies of the VE's at the various servers?

Provide answers to Steps 2, 3, and 4 of the Systematic Approach to Performance Evaluation discussed in class, i.e.,

Step 2: Services and outcomes
Step 3: Performance metrics
Step 4: System parameters and workload parameters

You must give a clear definition of each system parameter, each workload parameter, and each performance metric; and discuss the reasons for including them on your lists. For step 3, you should consider both individual and global performance metrics.

Note that you only need to consider consistency of the VE's at the various servers. Consistency of VE's seen by the users is outside the scope of this assignment. The following definition of consistency should be used when you prepare your answers.

With respect to a server, say server *a*,

- A user is in the consistent state if server *a* is the local server of this user.
- A user is also in the consistent state if this user is logged on to a remote server *b* and its location in server *a*'s VE is the same as that in server *b*'s VE; otherwise this user is in the inconsistent state.

Note that with respect to a given server, the state of a user (consistent or inconsistent) can change as a result of user movement. This is in addition to state change caused by delay in transmitting location updates to affected users. Algorithm to maintain consistency is outside the scope of this assignment.

IV. Simplifications

- In preparing your answers, you only need to consider changes in avatar location because of user movements; there is no need to consider detailed behaviours of the users in terms of movements within the VE.
- There is no need to consider details of the system architectures of the servers and the user workstations.
- There is no need to consider load balancing and server failure.
- For the network, use the assumptions that
  - the delay in transmitting an update packet between a user *u* and a server *a* is given by d_{au} (a constant depending on *a* and *u*) for all *a*, *u*.
  - the delay in transmitting a syn packet between servers *a* and *b* is given by D_{ab} (a constant depending on *a* and *b*) for all *a*, *b*.

NOTE:

- Late assignments will not be accepted.
- This assignment will count 7.5% towards the final grade.
- Your mark will be affected by the quality of your answers compared to the quality of answers submitted by the other students in class.