

## МОДЕЛЬ ВЗАИМОДЕЙСТВИЯ ВИРТУАЛЬНЫХ ОБЪЕКТОВ ДЛЯ ИМИТАЦИИ РАБОТЫ В СЕТИ INTERNET

*Аннотация.* Обсуждается модель взаимодействия виртуальных объектов, которая позволяет эффективно строить процесс обучения работе в сети Internet.

*Ключевые слова:* виртуальность. Интернет.

*Дисциплины:* программирование.

В настоящее время во все сферы нашей жизни происходит внедрение глобальных информационных технологий. Одним из развивающихся направлений этой области является использование служб Internet в повседневной работе. К сожалению, большинство пользователей этой сети являются неквалифицированными, что приводит к бесполезному трафику, несоразмерной оплате за канал Internet, а, в ряде случаев, и к нарушению этики общения. Это можно объяснить тем, что человек, черпая свои знания из литературы, пытается применить их на практике, что без опытного инструктора сделать довольно сложно.

Существующие программы, обучающие работе в сети Internet, представляют собой простые слайды, которые описывают основные термины, понятия этой сети, а также правила работы с некоторыми программами типа Web-браузер.

Необходимой составляющей любого обучения является практическое закрепление полученных знаний. В нашей стране большинство учебных заведений не могут себе позволить подключение к Internet лишь в учебных целях, так как это потребует больших затрат на оплату канала связи. Поэтому единственным выходом из данной ситуации может служить использование программ-эмуляторов, т.е. таких программ, которые имитируют работу сети Internet без непосредственного подключения к ней. Программа-эмулятор не должна просто имитировать работу сети, она должна быть инструктором для обучаемого.

В этой статье предлагается модель и архитектура взаимодействия виртуальных объектов, предназначенных для имитации работы в сети Internet.

Любую систему можно описать ее состоянием. Состояние системы - набор значений ее параметров в заданное время. С другой стороны, система постоянно подвержена изменениям, как внутренним со стороны пользователя (ввод команды), так и внешним (ошибка передачи, ответ на письмо). Такое изменение иногда называют воздействием. Следовательно, функция изменений состояния системы выглядит так:

$$S_{i+1} = f(P_i, U_i, E_i)$$

где :  $P_i$  - текущие параметры системы;  $U_i$  - воздействие пользователя;  $E_i$  - внешние воздействия

$P, U, E$  представляют собой наборы допустимых и недопустимых воздействий. В случае, если было хотя бы одно недопустимое воздействие, система переходит в ошибочное состояние. Ошибочное состояние представляет собой такое состояние, при котором нормальная работа системы невозможна, и требуются другие воздействия, чтобы она перешла в нормальное состояние. Следовательно, наборы допустимых и недопустимых воздействий изменяются в зависимости от состояния системы. Таким образом, имитационная модель работы в сети Internet выглядит как зависимость состояния системы от допустимых или недопустимых воздействий.

Будем называть любое воздействие событием. Тогда любое действие пользователя или изменение другим образом состояния системы будет событием. С другой стороны, имитационная модель предполагает имитацию воздействия пользователя. Поэтому необходимо проектировать виртуальные объекты таким образом, чтобы они, с одной стороны, позволяли пользователю воздействовать через них на систему, а с другой стороны, позволяли самой системе имитировать деятельность пользователя.

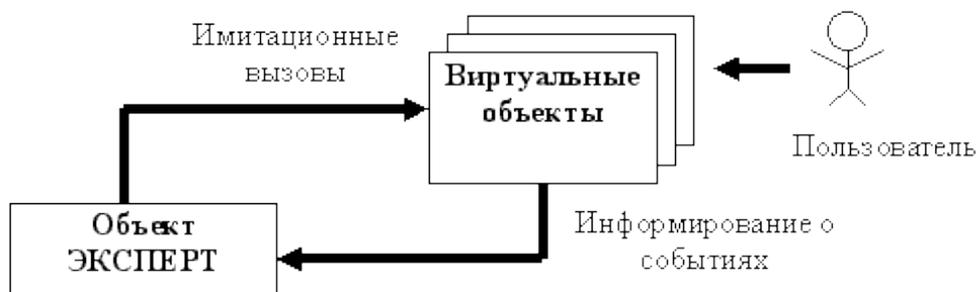


Рис.1. Модель взаимодействия виртуальных объектов

Посредством виртуального объекта (рис.1) реализуется имитация какой-либо службы Internet. При этом целесообразно использовать принципы многопоточного программирования, когда каждый виртуальный объект представляет собой отдельный процесс в вычислительной машине. Любое воздействие на виртуальный объект порождает событие, которое информирует объект-эксперт об изменении состояния системы.

Объект-эксперт реализует процесс обучения в данной системе. В этом объекте хранятся все множества состояний  $S$ , а также множества допустимых и недопустимых воздействий  $P, U, E$ . Кроме этого, объект-эксперт хранит правила изменения состояний, т.е. допустимые воздействия, которые переводят систему в другое допустимое состояние. В функции этого объекта входит отображение информации в виде текстовых и графических экранов, описывающих пользователю состояние системы. При необходимости пользователь может инициировать имитационное воздействие на виртуальный объект со стороны объекта-эксперта, для того чтобы посмотреть каким образом то или иное его воздействие переводит систему в иное допустимое или недопустимое состояние. Сам объект-эксперт реализуется как самостоятельный процесс, отслеживающий изменение состояния системы посредством событий, поступающих от виртуальных объектов.

Рассмотрим использование представленной модели на примере эмулятора службы передачи файлов (FTP).

Виртуальный объект, эмулятор FTP, представляет собой абстрактный удаленный компьютер, с которым пользователь производит соединение и абстрактный терминал, с которого пользователь вводит команды. Каждый из этих объектов реализуется самостоятельным потоком команд, работающих параллельно.

Для обеспечения имитации были выделены множества  $P, U, E$ . В частности, в начальный момент времени допустимым действием пользователя является выдача команды на соединение с удаленным компьютером. Из множества  $E$  нет допустимых состояний, так как воздействий со стороны системы быть не может потому, что имитируемый сеанс связи еще не установлен.

После прохождения команды установки соединения множества  $P, U, E$  изменяются. При этом становится допустимым внешнее воздействие, такое как имитация случайного обрыва канала связи. От пользователя ожидают ввод команд, отображающих каталог или позволяющих "получить" какой-либо файл. Здесь недопустимыми командами являются команды с неверными параметрами (неверный путь или имя файла).

Когда пользователь завершил имитируемый сеанс связи соответствующей командой, система возвращается к первоначальному состоянию, при котором допустимо только одно воздействие из множества  $U$  - команда установления удаленного соединения.

Объект-эксперт реализован в виде самостоятельного потока, который уведомляется об изменении состояния системы путем вызова виртуальным объектом соответствующего метода объекта-эксперта. Сам объект-эксперт по команде пользователя (нажатие кнопки) вызывает имитационный метод виртуального объекта для демонстрации пользователю ожидаемых от него действий. При этом состояние системы меняется точно также, как если бы эту команду ввел сам пользователь. В объекте-эксперте хранится таблица переходов допустимых и недопустимых состояний, на основании которых анализируются воздействия  $U, E$  на их допустимость. При вызове метода уведомляющего о событии объект-эксперт изменяет текущий элемент таблицы состояний в соответствии с правилами, которые она определяет.

Показанная модель взаимодействия виртуальных объектов позволяет эффективно строить процесс обучения работе в сети Internet. Кроме этого, при использовании объектно-ориентированного подхода, а также приемов параллельного программирования, такая модель достаточно проста в реализации.