

РАЗРАБОТКА ОБЪЕКТНОЙ МОДЕЛИ РАБОТЫ WEB-УЗЛА

Терещенко В.В., группа АСУ-00б

Руководитель проф. Лаздынь С.В.

Internet - сеть, которая объединяет миллионы компьютеров повсюду в мире. Эти машины могут быть самых разнообразных типов с различными операционными системами. Объединение подобного многообразия произошло благодаря семейству (или стеку) коммуникационных протоколов TCP/IP (Transmission Control Protocol / Internet Protocol). Протокол - набор инструкций, описывающих как системы взаимодействуют друг с другом, как должна пересылаться информация по компьютерной сети и т.п. Поскольку все системы в Internet поддерживают TCP/IP, то данные могут гарантированно пересылаться от одного компьютера к другому без каких-либо проблем.

Интернет можно определить как множество локальных сетей, называемых web-узлами и дающих доступ в Интернет конечному пользователю, объединенных между собой. С точки зрения физических соединений, они могут иметь самую разную природу. Начиная от модемных соединений по телефонной линии (до 50 кбит/с) и заканчивая связью через спутники.

Для оценки качества работы web-узлов используется интегральный критерий - время реакции web-узла. Такой выбор очевиден. Если web-узел работает медленно или неустойчиво, это может негативно на количестве клиентов, пользующихся его услугами. По результатам исследования компании Zona Research (USA), 30% посетителей не ждут окончания загрузки web-страницы, если это время превышает 8 секунд, а 70% посетителей "уходят раздраженными", если это время превышает 12 секунд.

Кроме времени реакции web-узла используется еще множество тестов и параметров, характеризующих работу как web-узла в целом, так и отдельных его составляющих. Условно их можно разбить на следующие группы: общая

производительность web-узла, производительность Интернет-каналов, производительность серверов, производительность сетевого оборудования.

Основными методами моделирования систем, аналогичных выбранной по принципу функционирования, являются следующие методы моделирования:

- аналитическое моделирование;
- имитационное моделирование;
- экспериментальные методы;
- аналитическое моделирование на основе теории систем массового обслуживания.

Проанализировав все вышеперечисленные методы можно прийти к выводу, что наиболее оптимальным является аналитическое моделирование на основе теории систем массового обслуживания, но и оно не лишено своих недостатков. Из всех методов, применяемых теорией массового обслуживания, наиболее приемлемым является принцип Δt .

Он заключается в том, что весь промежуток функционирования объекта разбивается на участки Δt , в течении которых все параметры системы или объекта считаются постоянными и для них составляется модель, производятся вычисления, определяются параметры.

Достоинство этого принципа в возможности использования для непрерывных процессов. А недостаток в громоздкости решения задачи и получение большого количества избыточной информации, если подстраиваться под самый быстрый процесс или недополучение информации, если подстраиваться под самый медленный процесс.

В некоторых случаях, например, когда мы моделируем появление заявок на компьютерах клиентов, целесообразно использовать принцип особых состояний.

Под особыми состояниями подразумевают такие моменты времени, когда какой-либо из параметров изменяется скачком. Например, поступление заявки в систему, момент выхода заявки из системы. Не особые состояния - состояния

или промежутки времени, в которые параметры системы не меняются совсем или меняются плавно. Например, ожидание, обработка заявок и т. д.

Сам принцип заключается в том, что рассматривается последовательность особых состояний и составляется модель системы в эти моменты времени. Обычно рассматривают особые состояния одного и того же типа до тех пор, пока не наступит особое состояние другого типа. Достоинство заключается в получении любых характеристик систем. Недостатком является сложность получения алгоритма.

Основным недостатком этих методов является необходимость их адаптации под каждую конкретную систему, которая может занимать много времени. Наиболее приемлемым методом решения данной задачи является применение к уже перечисленным выше методам объектного подхода. Для этого нам необходимо выделить основные классы объектов, которые включает в себя исследуемая система и продумать методы их взаимодействия. Это уже частично было сделано в прошлом семестре и описывалось в отчете по НИРС. Список объектов исследуемой системы выглядит примерно так:

- сервер(программа обрабатывающая запросы)
- компьютер
- клиент
- глобальная сеть
- канал связи
- запрос(пакет данных от клиента к серверу)
- глобальная компьютерная сеть(интернет)

Сервер должен выполнять определенные функции и иметь определенные свойства. В свою очередь разнообразие их типов делает невозможным реализацию их как одного объекта. Наиболее рационально будет реализовать данную структуру в виде абстрактного класса `MyServer` и его потомков разных типов.

Класс `MyServer` должен включать в себя три класса включающие в себя данные о распределении трех случайных величин: длин входящих и исходящих запросов и число команд для обслуживания сервером заявки. Основными его функциями будут функции генерации случайных величин на основании свойств объекта и функции установки и изменения параметров.

У потомков данного класса все эти свойства и методы будут идентичны. Класс `MyHTTPServer` также будет включать в себя параметр, характеризующий процент запросов к нему, которые вызывают вторичные запросы к серверу баз данных. Кроме стандартных и описанных ранее серверов создадим также класс `MyUniversalServer`. Его назначение будет описано позже.

Каждый потомок основного класса также будет включать в себя значения по умолчанию объектов именно этого вида для удобства работы с ними. Данную структуру мы можем видеть на рисунке 1.

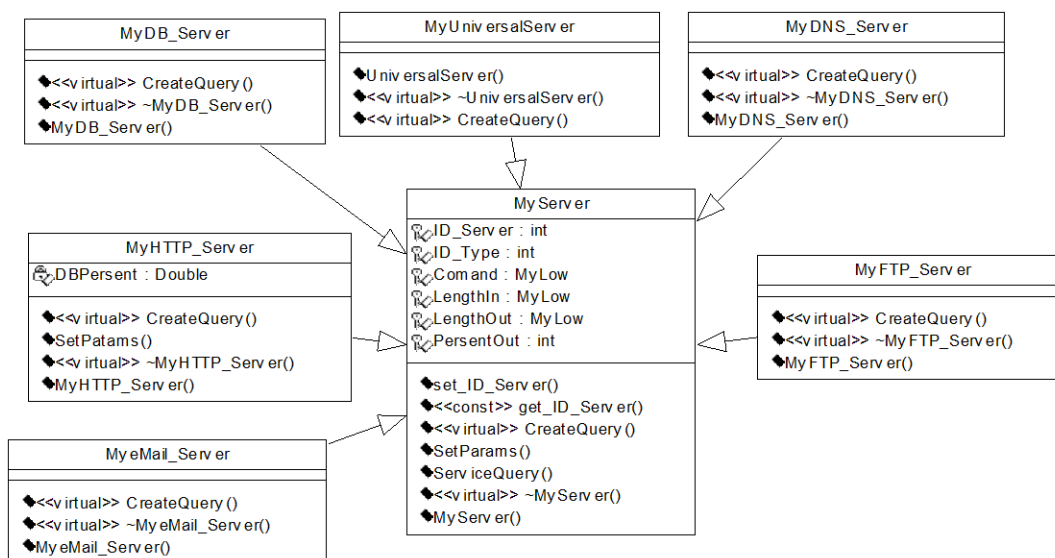


Рис. 1. Структура классов серверов.

Каналы связи, соединяющие web-узел с клиентами и глобальной сетью, отличаются только своими числовыми характеристиками. Поэтому для их реализации в объектной модели необходим один класс. Любой канал связи

характеризується скоростью передачи данных и процентом потерь, если таковые имеются. Основными функциями канала связи будет функция обслуживания заявки и функции установки и изменения параметров.

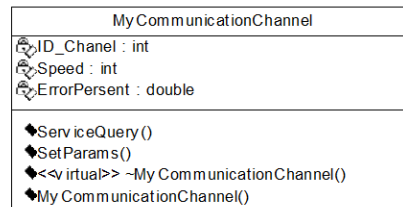


Рис. 2. Класс канала связи.

Компьютер на котором установлены программы-серверы описывается как класс со следующими свойствами: тактовая частота, процент использования операционной системой ресурсов центрального процессора, а также список заявок находящихся в данный момент на обслуживании. Этот класс как и все предыдущие должен выполнять функции установки и изменения своих параметров, а также функции обслуживания заявок.

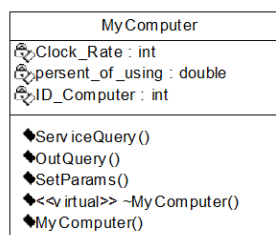


Рис. 3. Класс компьютера.

Клиенты как классы объектов включает в себя набор законов распределения случайных величин, которые описывают вероятность появления заявок к какому-либо серверу с компьютера клиента. Этот класс должен обеспечивать функции для работы с набором своих законов распределения, установку параметров и генерацию запросов.

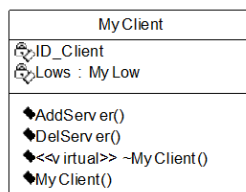


Рис. 4. Класс клиента.

Глобальная сеть представляет с собой не что иное, как клиента периодически генерирующего запросы разного типа к web-узлу и сервер, периодически обслуживающий заявки от web-узла. В общей объектной модели это можно реализовать как класс, включающий в себя еще 2 класса: MyClient и MyUniversal Server. Класс MyUniversalServer уже описывался выше и нужен для того чтобы имитировать работу глобальной сети по обработке заявок от пользователей.

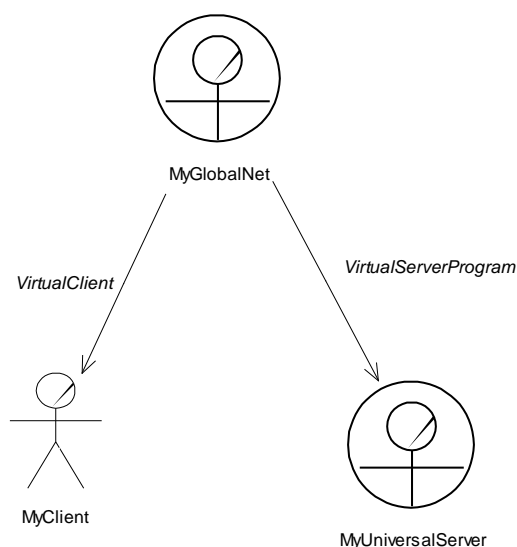


Рис. 5. Структура класса глобальной сети.

Следовательно сам класс MyGlobalNet не будет иметь никаких дополнительных функций и свойств и будет иметь вид, показанный на рисунке 6.

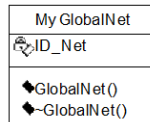


Рис. 6. Класс глобальной сети.

Все запросы, которыми обмениваются между собой компоненты системы, характеризуются источником, получателем, типом (тип сервера-приемника). При обслуживании также необходимы данные, какая часть запроса уже передана, обслужилась. Кроме функций для работы с параметрами запрос сам по себе больше никаких функций выполнять не будет.

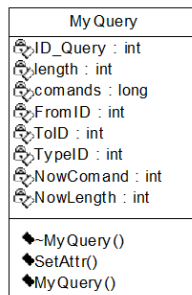


Рис. 7. Класс запроса.

При описании классов MyClient и MyServer и его потомков мы включили в них класс, описывающий законы распределения. Данный класс еще не был описан. Этот абстрактный класс будет включать в себя свойства описывающие математическое ожидание и дисперсию случайной величины, а также виртуальный метод генерирующий эту величину. В потомках класса будет реализована генерация случайной величины для конкретного закона распределения: нормального, равномерного и экспоненциального.

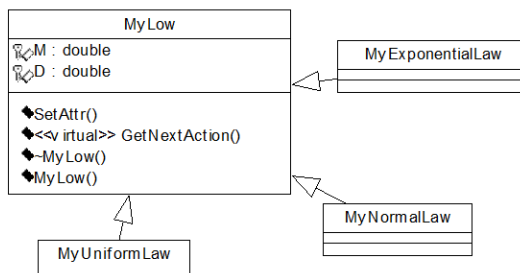


Рис 8. Классы законов распределения случайной величины.

Общую диаграмму использования для основных объектов можно представить в виде, показанном на рисунке 9.

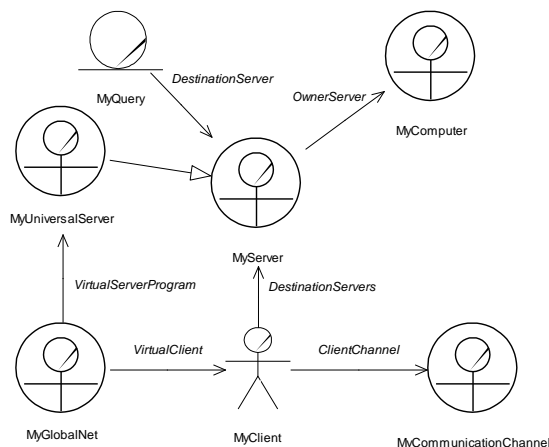


Рис. 9. Диаграмма использования для основных компонентов системы.

Проанализировав предыдущие выводы можно разработать укрупненный алгоритм функционирования web-узла, блок-схема которого приведена на рисунке 10.

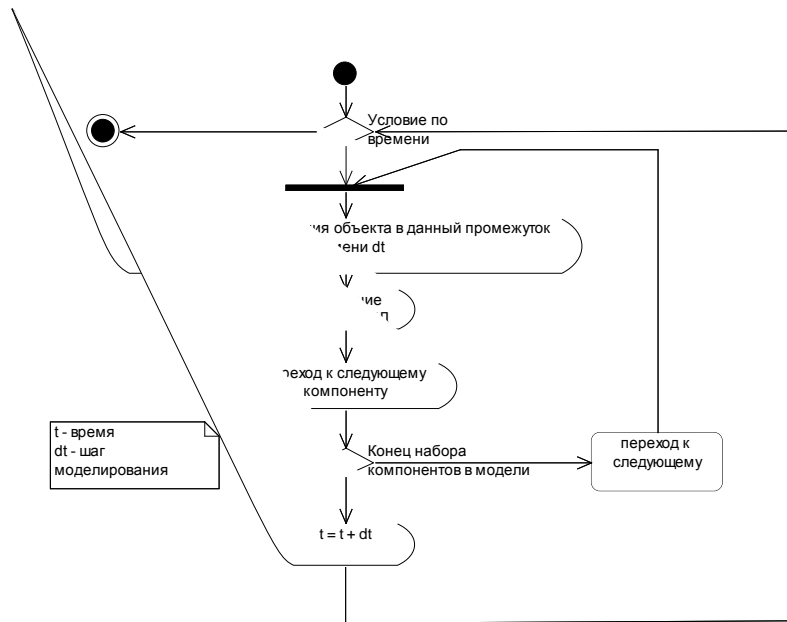


Рис. 10. Укрупненный алгоритм функционирования web-узла.

В данной статье рассмотрены основные принципы функционирования web-узла и сделана попытка разбивки его на типовые компоненты с последующей их формализацией. Также разработан укрупненный алгоритм моделирования его работы.

Перечень ссылок

1. Сайт компании ProLan: <http://www.prolan.ru>
2. Математические методы и модели, обеспечивающие оценку производительности ЭВМ: <http://masters.donntu.edu.ua/2000/fvti/sugonyak/magistr/>
3. Вводное и подробное описание принципов работы протоколов TCP/IP: <http://src.nsu.ru/ff/phti/iit/internet/tcpipdoc/index.htm>