

УНИВЕРСАЛЬНАЯ СРЕДА МОДЕЛИРОВАНИЯ ДЛЯ ПРОФЕССИОНАЛОВ

Т. В. Девятков (Казань)

Создание моделей на практике – это задача, которая включает в себя не только написание самой модели, получение результатов работы этой модели и их интерпретации, а также и такие шаги, как постановка задачи; обработка входной статистики; планирование серий экспериментов и т.п. Данные шаги подробно описаны во многих трудах (например, в [1]). Наиболее полное исследование в практическом смысле – это титанический труд, с совмещением знаний и умений работы в различных прикладных средах, решающих узкие задачи, которые позволяют автоматизировать отдельные этапы, но не дающие возможности состыковки результатов одной среды с другой, для этой задачи необходима дополнительная работа.

Таким образом, уже давно возникла необходимость автоматизации всех этапов разработки в рамках системы автоматизации имитационных исследований (САИИ), которая объела бы как можно больше этапов создания имитационных моделей. Актуальность такой задачи в настоящее время все возрастает в связи с ростом спроса на практические имитационные модели.

Так в чем же состоит суть САИИ? Перечислим основные шаги, которые помогает автоматизировать эта среда:

- 1) постановка задачи;
- 2) обработка исходных данных;
- 3) моделирование;
- 4) планирование и проведение экспериментов над моделями;
- 5) анализ и обработка результатов.

Если включить в среду моделирования данные этапы, имитационное моделирование перейдет из разряда искусства в инженерные задачи, которые доступны более широкому кругу пользователей [1,2]. Данная статья посвящена описанию универсальной моделирующей среде (далее среда) для профессионалов, созданной автором.

В нашей компании уже несколько лет ведется работа над созданием различных типов САИИ [4,5,6,7]. Есть разработки, которые автоматизируют отдельные этапы, на данный момент идет работа по стандартизованному объединению этих разработок в общий контур, посредством создания универсального приложения – программной шины.

Это приложение полностью основано на сервис-ориентированной архитектуре (SOA), которая позволяет размещать сервисы среды на различных машинах, разделяя обработку данных на большие мощности [3]. Все управление в этой среде осуществляется через специальную программную шину, которая обеспечивает взаимодействие всех узлов среды. Такой подход, к примеру, позволяет размещать в локальной сети несколько моделирующих ядер, несколько средств по планированию и проведению экспериментов и т.п., что позволяет загружать моделирующую среду максимально эффективно, без простоев локальных средств.

Основную идею шины можно изобразить следующим образом:



Рис. 1. Структура сервис-ориентированной среды моделирования

На рисунке видно, что программная шина осуществляет контроль за сервисами, которые находятся в контуре ее управления. Список сервисов пополняется за счет устанавливаемых компонент в локальной сети, после чего их надо зарегистрировать в шине, чтобы она имела доступ к новым компонентам. Компьютеры, на которых установлена клиентская программа, посылают шине задания, поступающие в очередь на обработку, при которой определяется, какому сервису надо перенаправить данную задачу. Также, если подобных сервисов несколько, определить свободный и передать задание ему. После чего результат, также через шину, возвращается назад к клиенту.

Вся информация в среде идет через программную шину, которая целиком контролирует процесс, является главным звеном в мониторинге за исполнением заданий, на основе этого мониторинга можно определить, какие задания выполняются дольше всего. Появляется возможность выявить узкие места локальной среды и пополнить по необходимости сеть дополнительными сервисами, которые помогут повысить производительность всей системы.

Всю работы универсальной среды моделирования для профессионалов можно разбить на пять прикладных уровней:

- 1) клиентский уровень – то, что видит непосредственно пользователь во время своей работы на компьютере;
- 2) транспортный уровень – скрытый от клиента протокол передачи данных в сети, мы используем TCP, HTTP;
- 3) уровень шины – ключевая программа всей системы, которая осуществляет контроль потока заданий;
- 4) уровень сервисов – уровень, позволяющий выполнять задания на другом компьютере;
- 5) уровень ядра – низший уровень, на котором собственно и происходит моделирование.



Рис. 2. Пятиуровневая структура среды моделирования

Подобная структура позволяет быстро объяснить прохождение потока данных в системе, от клиента до ядра и назад, с постоянным прохождением через программную шину.

Теперь остановимся на клиентской части. Название статьи: «Универсальная среда моделирования для профессионалов». Вот в чем состоит ее универсальность:

- распределенные вычисления, которые позволяют существенно повысить производительность системы в целом, задействуя большое количество локальных вычислительных систем;
- возможности по параллельной работе нескольких исследователей с одной версией ядра моделирования, а при необходимости использование не-

скольких ядер одним исследователем, при проведении серий экспериментов;

- удобное написание моделей в визуальной структуре, с последующей доработкой полученного кода модели;
- автоматизация создания приложений по моделированию при помощи встроенной среды создания форм.

Для реализации поставленной задачи было написано три разных компонента для работы в локальной сети. Каждый элемент данной программной структуры имеет четкое распределение полномочий [4]:

Клиент – программа, которая предоставляет возможности по вводу, корректировке моделей. Также в нее включены возможности коммуникации клиентов между собой (чат), редактор формирования и корректировки моделей, процедуры формирования запроса на исполнение моделей (или серии экспериментов с моделью) и средства анализа результатов исполнения.

Шина – программа, принимающая запросы на моделирование и распределяющая запросы по известной ей сети серверов. Диспетчеризация заявок шиной осуществляется по различным алгоритмам с учетом разных параметров заявки (приоритет, запрашиваемый квант времени и т.д.).

Сервис – сервис, обладающий доступом к ядру моделирования либо планирования и проведения серий экспериментов и т.п. Обеспечивает исполнение заявок, в соответствии с очередностью и условиями, определенными шиной. Формируемые результаты передаются сервером через шину клиенту.

Шин в данной архитектуре может быть несколько, клиентов и серверов – ограниченное локальной сетью количество. Количество клиентов определяется запросами и требованиями пользователей, например, при проведении лабораторных работ число клиентов должно соответствовать числу компьютеров в дисплейном классе. Количество серверов определяется числом лицензий используемой системы моделирования.

Важно отметить следующее – не только клиент выполняет какие-то задания на сервере, но и компьютер, на котором находится шина, может отправлять задания на выполнение сервисам; сервер, обладая ядром, может выполнять моделирование у себя, предварительно отправив уведомление шине о своей занятости.



Рис. 3. Встроенный редактор языка PascalScript



Рис. 4. Встроенный редактор GPSS

Работа шины заключается в следующем: на нее поступают заявки на выполнение действий от клиентов, они регистрируются в очереди. После чего идет просмотр списка свободных сервисов, посылается запрос на возможность выполнения задания, если нужный сервис свободен, шина берет задачу из очереди и отправляет на выполнение, так продолжается до тех пор, пока не опустеет очередь или же пока не закончатся свободные сервисы. По окончании работы сервис посылает шине сообщение о завершении задания и отправляет результат шине. Шина, получив данное сообщение, проверяет на наличие задач в очереди. Если задачи в очереди есть, то сразу посылает задачу на выполнение на запрашиваемый сервис, если задач нет, то указывает в своем списке сервисов, что приславший сообщение сервис свободен для выполнения задач.

По завершении моделирования происходит перехват результатов приложением, после чего результаты отправляются клиенту для анализа, т.е. пользователь получает полный отчет GPSS World в стандартном виде, кроме того, используется специальный сервис для динамического сбора данных из модели по очередям, устройствам и т.п. Данные о том, какие стандартные числовые атрибуты и с какой интенсивностью надо собирать, указываются пользователем или же используются по умолчанию. Все данные, собранные системой динамического мониторинга, сохраняются в базе данных и могут быть использованы в дальнейших исследованиях, например, для построения регрессионной модели. Также в приложении реализуется перехват системных сообщений GPSS об остановке моделирования, например, таких, как HALT и т.п.

Сервис-ориентированный подход при разработке САИИ дает большие преимущества по использованию программных продуктов, вследствие чего данное направление является в настоящее время приоритетным для нашей компании. Сейчас идет доработка приложений, которые были разработаны до этого, их адаптация к структуре SOA, а также написание новых.

Литература

1. **Нейлор Т.** Машинные имитационные эксперименты с моделями экономических систем. М.: Мир, 1975. 500 стр.
2. **Шеннон Р.** Имитационное моделирование систем – искусство и наука. М.: Мир, 1978. 418 стр.
3. **Колесов А.** SOA: переход от теории к практике. Byte Magazine, № 6, 2008. // <http://www.bytemag.ru/articles/detail.php?ID=12164>
4. **Власов С. А., Девятков В. В., Девятков Т. В.** Универсальная моделирующая среда для разработки имитационных приложений// Информационные технологии и вычислительные системы, № 2. 2009. С. 5–12.
5. **Vlasov S. A., Deviatkov V. V.** Experience in the development and application of simulation in Russia: review analysis of prospects: Proceedings 19th European Conference on Modeling and Simulation, ECMS 2005, June 1–4 2005, pp. 23-27.
6. **Власов С. А., Девятков В. В., Девятков Т. В.** Концепция и методы разработки распределенных имитационных приложений с использованием среды GPSS World//Тезисы Международной научно-технической конференции «Информационные технологии и математическое моделирование систем», Майорка, Испания. Сентябрь, 2008.
7. **Козлов А. Н., Девятков Т. В., Кейер П.А.** Исследование функционирования центра коллективной обработки информации методом имитационного моделирования// Третья всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности. Сб. докладов. СПб, 2007. Т. 2. С. 96–100.