

МОДЕЛИРОВАНИЕ АСИНХРОННОЙ СЕТИ АВТОМАТОВ ОБРАБОТКИ ДАННЫХ*

И.В. ЗИМАЕВ[♥], А.А. ВОЕВОДА[♦]

На примере простой сети терминалов подключенных к серверу рассмотрен модульный подход к моделированию дискретных систем с использованием аппарата цветных сетей Петри. Показано, как каждый модуль подобной системы может быть разработан и протестирован независимо от других, а объединение модулей в систему осуществляется на основе схемы обмена сообщениями.

ВВЕДЕНИЕ

Надежность и эффективность работы аппаратных средств и программного обеспечения многопользовательских систем должны быть высокими. Разработка и поддержка современных программных комплексов, обрабатывающих колоссальные объемы данных с высокой степенью отказоустойчивости остается крайне дорогой. Использование современных методов программной инженерии и имитационного моделирования призвано облегчить проектирование и тестирование сложных программных систем.

В основе используемого в статье подхода лежит объектно-ориентированная парадигма. Согласно этой уже устоявшейся, но всё еще современной и перспективной методологии, каждая система состоит из набора блоков, каждый из которых является относительно независимым, обладает определенным набором атрибутов, и связан с остальными блоками системы посредством обмена сообщениями. Процесс объектно-ориентированного проектирования облегчается тем, что выделяемые блоки модели соответствуют объектам реального мира [1].

* Работа выполнена при финансовой поддержке Федерального агентства по образованию, ГК № П694 от 12.08.2009, конкурс НК-81П

[♥] Аспирант кафедры автоматики

[♦] Профессор кафедры автоматики, д-р техн. наук

1. АРХИТЕКТУРА СЕТИ

Простейшая сеть терминалов удаленных операций имеет топологию «звезда»: несколько терминалов подключены к одному серверу, но не связаны между собой напрямую (рис. 1). Физическая природа канала связи в рамках рассматриваемой задачи значения не имеет, однако необходимо учесть требования оптимального использования ресурсов. Фактически это означает неспособность сети функционировать при полной нагрузке, когда все терминалы одновременно пытаются установить связь с сервером.

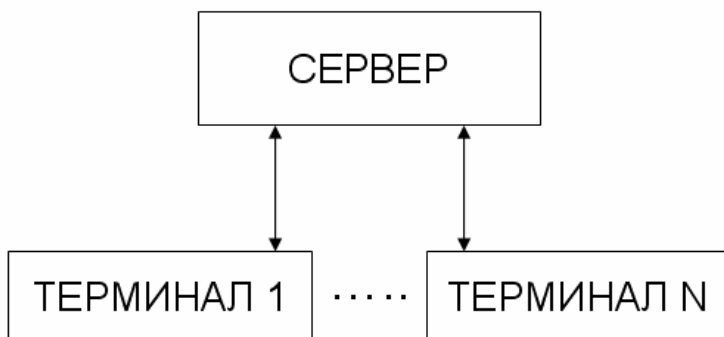


Рис. 1. Топология сети

Поскольку подобная ситуация маловероятна и наличие непрерывного канала связи не является обязательным, целесообразно устанавливать соединение терминала с сервером по мере необходимости (в момент использования терминала).

2. СИСТЕМА И ЕЕ МОДУЛИ

Обычно разработка модели системы начинается с создания UML-диаграммы вариантов использования (use-case). В случае многотерминальной системы обработки данных действующими лицами (т. е. внешними относительно системы сущностями) являются пользователи терминалов, а базовых вариантов использования два: запрос некоторой информации, необходимой для совершения операции, и сама операция (рис. 2).

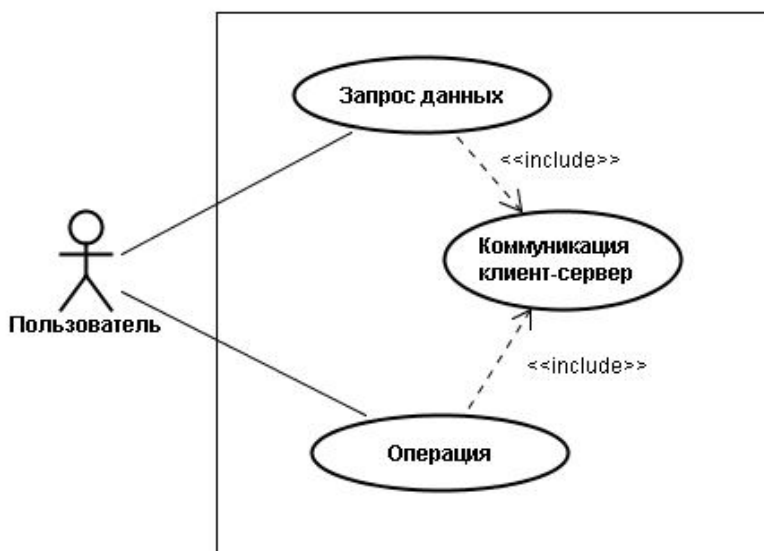


Рис. 2. Диаграмма вариантов использования

Запрос данных и запрос-уведомление об операции подразумевают пересылку информации от клиента (терминала) к серверу и обратно. При этом клиент и сервер можно представить абстракцией черного ящика. Модульный подход к проектированию системы подразумевает выделение функций, выполняемых модулем и правил взаимодействия с ним, а внутренняя реализация этих функций вторична. Описав входы и выходы каждого модуля, из них возможно скомпоновать модель всей системы, что упрощает процесс разработки и тестирования. Корректная работа модуля гарантирована соответствием его внутренней реализации заданному внешнему интерфейсу. Минимальный набор сообщений в коммуникации терминала и сервера довольно прост (рис. 3).

В процессе взаимодействия терминал посылает определенные сигналы серверу (запросы данных, запросы операций, уведомления о совершении операций) и получает также строго определенный набор ответных сигналов (запрошенные данные, подтверждения операций).

Модульный подход позволяет проектировать каждый модуль отдельно, с учетом, разумеется, требования последующей совместимости всех модулей.

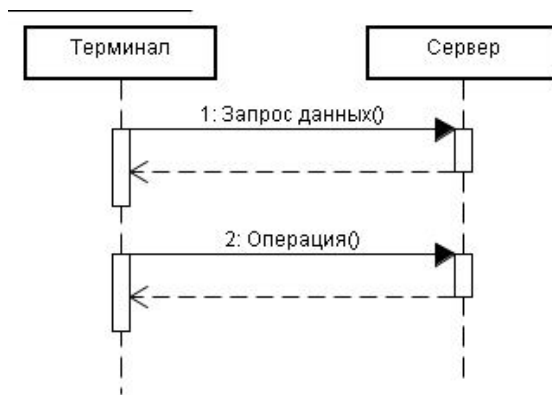


Рис. 3. Диаграмма взаимодействия терминала и сервера

Более детализированное представление работы клиента и сервера можно получить при выделении в них набора сменяющих друг друга состояний, что аналогично представлению системы как конечного автомата (рис. 4).

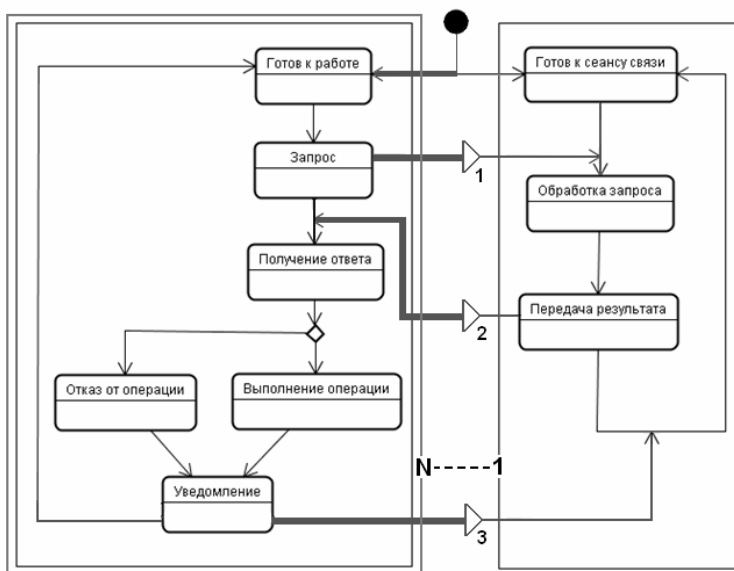


Рис. 4. Модифицированная диаграмма состояний работы терминала и сервера

Данная диаграмма несколько отличается от традиционной диаграммы состояний, поскольку ее графическая нотация дополнена некоторыми элементами. Рассматривая автомат состояний взаимодействующих клиента и сервера, целесообразно отметить не только их причинные взаимосвязи (когда состояние одной системы является инициатором смены состояний в другой системе), но и наличие множества терминалов, подключенных к одному серверу. На схеме это достигается следующими элементами:

1) двойной контур автомата работы клиента с индексом N (связанный с единичным индексом сервера, связь «многие к одному»);

2) утолщенные линии связи терминала и сервера (словно иллюстрирующие «широкий» поток данных). При этом внутренние линии автомата терминала имеют обычную толщину, поскольку демонстрируют его работу безотносительно факта наличия прочих терминалов.

Начальное состояние терминала – «готов к работе», а сервера – «готов к сеансу связи». Условие перехода терминала в состояние «запрос» на диаграмме не обозначено, поскольку оно по своей сути является внешним относительно изображенных систем (речь о пользователе). Запрос терминала переводит сервер в состояние обработки запроса и передачи результата на терминал, который, в свою очередь, переходит в состояние обработки полученных данных. При этом либо происходит отказ от запрошенных действий, либо они выполняются. При обоих вариантах высылается уведомление серверу, оно также выдается пользователю терминала, после чего терминал переходит в начальное состояние готовности. Сервер, получив сигнал о завершении работы, также переходит в состояние начальной готовности.

Наличие сигналов от множества терминалов к одному серверу изображено на схеме в виде треугольников (1-3), при этом фактическое направление потока данных определяется не направлением треугольника, а стрелкой на одном из концов линии:

1 – различные терминалы могут одновременно посылать запросы серверу;

2 – сервер, параллельно обслуживая множество терминалов, способен передавать данные сразу нескольким из них;

3 – несколько терминалов могут уведомлять о завершении операций.

Отображение этих потоков параллельных данных делает модель системы более наглядной, и может быть особенно полезным при последующем перехо-

де от UML диаграммы к сети Петри [1, 2], являющейся имитационной динамической моделью системы.

3. ДИНАМИЧЕСКАЯ МОДЕЛЬ

Рассмотрим простейшую динамическую модель системы терминалов в виде цветной сети Петри (рис. 5).

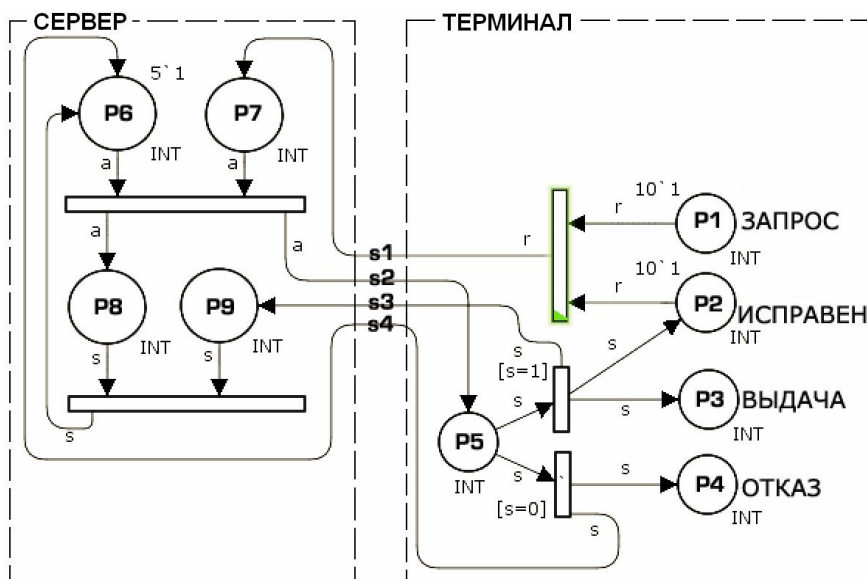


Рис. 5. Сеть Петри как динамическая модель модулей системы

Для наглядности схемы сохранена одноуровневая структура сети, в действительности же целесообразно прибегнуть к возможности иерархических сетей Петри.

Между терминалом и сервером имеется интерфейс из четырех сигналов (физически передаваемых, разумеется, по одному каналу связи):

- S1 – запрос у сервера возможности операции;
- S2 – ответ сервера на запрос S1;
- S3 – уведомление сервера о проведенной операции;
- S4 – уведомление сервера о завершении сеанса работы.

Рассмотрим работу сети более подробно.

Начальное состояние сети – это наличие маркеров в позициях P1 и P2, что соответствует соблюдению условий: терминал исправен (P2) и пользователь терминала совершил запрос информации (P1). Срабатывание объединяющего их перехода помещает маркер в позицию P7 (сигнал S1), которую можно обозначить как «входящие запросы от клиентов». Поскольку для каждого клиента-терминала серверу необходим отдельный канал связи (не обязательно физически различные проводники сигнала), условием установления сеанса работы является наличие свободного канала, что определяется наличием маркера в позиции P6. Если свободный канал связи имеется, то на поступивший от клиента запрос сервер помещает маркер в позицию P8 (готовность к дальнейшей работе), а также в позицию P5 (сигнал S2), что является уведомлением терминала о запрошенной информации. В данном примере значение маркера 0 или 1 является признаком того, возможно ли совершение запрошенной на терминале операции. В зависимости от этого маркер переходит либо в позицию P3 (операция успешна), либо в позицию P4 (отказ выполнения операции). По завершении операции маркер переходит в позицию P2 (исправен, готов к работе) и в позицию P9 передается сигнал S3 (операция закончена). Наличие маркеров в позициях P8 и P9 приводит к срабатыванию перехода и добавлению маркера в позицию P6 (канал связи), т. е. соединение между сервером и терминалом освобождается для следующих запросов. Если же операция не совершена (позиция P4), маркер через сигнал S4 выставляется в позицию P6, канал связи также освобождается.

Для имитации работы нескольких терминалов начальная маркировка позиций P1 и P2 задана как 10^1 , что соответствует работе 10 терминалов, при этом маркировка позиции P6 задана как 5^1 , что соответствует наличию только 5 каналов связи сервера с терминалами.

На уровне модели сети Петри модульное разделение системы сохраняется, что позволяет помещать фрагменты сети в отдельные блоки, определять для них набор входов и выходов, а затем строить из таких блоков иерархическую сеть. Подобный модульно-ориентированный подход в использовании сетей Петри для моделирования динамики систем может быть особенно полезным в ряде предметных областей, за счет создания набора стандартных «блоков», с формально специфицированными входами и выходами, что позволит более эффективно конструировать модели сложных систем [2]. При этом значительно облегчается верификация сложной системы, потому что при ее тестировании отдельные «блоки» могут заменяться на имитирующие их объекты, при-

нимающие и выдающие набор сигналов в соответствии с заданным интерфейсом. Предложенный подход в построении сетей Петри известен в разработке программного обеспечения как модульное проектирование и тестирование, и отличается высокой эффективностью.

[1] *Питерсон Дж.* Теория сетей Петри и моделирование систем / пер. с англ. – М.: Мир, 1984.

[2] *Воевода А.А., Зимаев И.В.* Об особенностях преобразования UML-диаграмм деятельности в сети Петри // Сб. науч. тр. НГТУ. – 2009. – 2(56). – С. 79 – 89.