МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАТИКЕ
(Вычислительные системы)

2002 год Выпуск 169

УДК 519.683.2

КОМПОНЕНТНАЯ МОДЕЛЬ РАСПРЕДЕЛЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

А.Г.Сердюк

Введение

Широчайшее распространение и повсеместное внедрение распределенных информационных систем приводит к необходимости математического моделирования таких систем. Построение и исследование модели разрабатываемой системы может помочь избежать архитектурных ошибок, наличия "бутылочных горлышек", замедляющих работу системы в целом. Моделирование существующей системы способно подсказать способы оптимизации ее функционирования.

Адекватным средством для построения моделей распределенных информационных систем является математический аппарат сетей Петри, предложенный К. А. Петри (ФРГ) в 1962 г. [1]. Настоящая статья предполагает знакомство читателя с формализмом сетей Петри; для ознакомления с ним можно рекомендовать работы [2,3].

Впоследствие были предложены различные расширения обыкновенных сетей Петри, увеличивающие их выразительные свойства. Однако расширения имеют существенный недостаток — они усложняют модель, приближая ее по мощности к машинам Тьюринга, что делает ряд проблем, связанных с исследованием свойств сетей, алгоритмически неразрешимыми.

В настоящей статье рассматривается построение модели распределенной информационной системы на базе обыкновенных сетей Петри. Модель является компонентной — она комбинируется из отдельных сходных блоков, соответствующих серверам и клиентам моделируемой распределенной информационной системы. Будет показано, что предложенный способ построения приводит к созданию моделей, обладающих рядом "хороших" свойств — ограниченностью, обратимостью и активностью.

1. Структура распределенной информационной системы

Начнем построение дискретной динамической модели распределенной информационной системы с неформальной модели, демонстрирующей базовые свойства и принципы функционирования распределенной информационной системы.

Распределенная информационная система представляет собой совокупность компонент, действующих как единое целое. Характерными чертами её являются такие, на первый взгляд противоположные свойства, как независимость и взаимосвязанность компонент. Сочетать эти свойства распределенной информационной системе позволяет использование стандартных протоколов взаимодействия компонент. Каждая из компонент распределенной информационной системы представляет собой "черный ящик", взаимодействие которого с другими подсистемами возможно только через предоставляемый внешний интерфейс. Независимость компонент выражается в том, что реализация каждого из них может быть в любой момент изменена — другие компоненты будут без затруднений взаимодействовать с изменившимся до тех пор, пока он поддерживает прежний интерфейс. Взаимосвязь же компонентов выражается в необходимости следования принятым соглашениям об интерфейсах взаимодействия.

В дальнейшем мы будем рассматривать только архитектуру взаимодействия "клиент-сервер", имеющую в настоящее время наиболее широкое применение. В рамках этой архитектуры взаимодействия организуются между парами компонент. Компонент, называемый *клиентом*, посылает другому компоненту, называемому *сервером*, сообщение с требованием выполнить определенные действия (*запрос*). Сервер производит действия и, возможно, в свою очередь тоже посылает клиенту сообщение (*ответ сервера*).

Для распределенных информационных систем характерно то, что многие компоненты могут являться одновременно клиентами и серверами в указанном выше смысле. Получив запрос от клиента, в процессе его обработки и подготовки ответа они обращаются с запросами к другим серверам, тем самым выступая в роли клиентов.

Из-за указанного смешения функциональности термины "клиент" и "сервер" часто используют в более широком смысле (что и мы будем делать в дальнейшем). Сервером в широком смысле называют компонент распределенной информационной системы, способный предоставлять серверные услуги в узком смысле термина (хотя, возможно, одновременно этот компонент производит обращения к другим серверам, играя роль клиента). Клиентом же в широком смысле называют компонент, не способный выступать в роли сервера.

В соответствии с этой классификацией компонентов распределенной информационной системы для построения модели будут использованы компоненты двух типов — клиенты (см. раздел 2) и сервер (см. раздел 3).

Неформально, модель распределенной информационной системы состоит из взаимосвязанных клиентских и серверных компонент. Клиентские компоненты обращаются к серверным компонентам, запрашивая предоставляемые теми сервисы. В процессе выполнения запросов серверные компоненты могут обращаться к другим серверным компонентам, распределяя нагрузку. Формальное построение модели распределенной информационной системы будет проведено в разделе 4.

Для каждой из рассматриваемых моделей будет показано выполнение свойств ограниченности, обратимости и активности. Для моделируемой информационной системы эти свойства означают невозможность переполнения буферов, невозможность необратимых изменений, взаимных блокировок и тупиков.

2. Клиент

Построение компонентной дискретной динамической модели распределенной информационной системы начнем с модели клиента.

Выделяются две разновидности клиентов — синхронные и асинхронные.

Синхронный клиент (рис. 1) работает в цикле "запрос–ответ". Послав запрос серверу, он ожидает ответа и, лишь получив ответ сервера, продолжает работу. Асинхронный же клиент (рис. 2) взаимодействует с сервером иначе: послав запрос серверу, он может не ожидать ответа, а продолжать работу. Когда ответ сервера будет получен, он будет обработан. Таким образом, асинхронный клиент имеет очередь исходящих запросов и очередь входящих ответов от серверов.

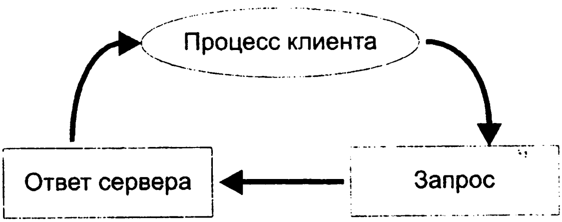


Рис. 1.

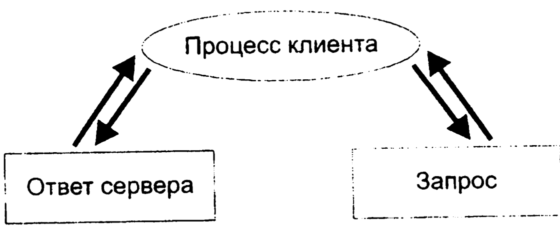


Рис. 2.

Нетрудно заметить, что синхронный клиент является частным случаем асинхронного с единичными размерами очередей исходящих запросов и входящих ответов и бездействием между отправкой запроса и получением ответа. Исходя из этого, в дальнейшем без ограничения общности будем рассматривать только асинхронных клиентов.

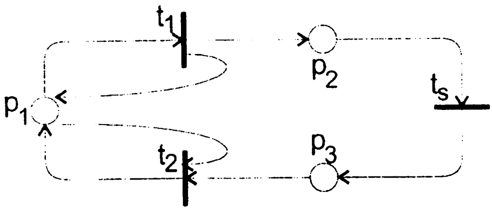


Рис. 3.

Простейшая упрощенная модель синхронного клиента, построенная в соответствие с описанной схемой его действия, приведена на рис. 3.

Позиция p_1 содержит маркеры, соответствующие клиентским процессам. Позиция p_2 соответствует очереди исходящих запросов, позиция p_3 — очереди входящих ответов серверов. Переход t_1 моделирует внутреннее событие клиента, требующее отправки запроса серверу. Переход t_s моделирует обработку сервером полученного запроса. Наконец, переход t_2 моделирует обработку клиентом ответа сервера.

На этом этапе построения модели мы сознательно упрощаем функционирование серверной стороны и считаем, что на каждый запрос клиента следует ответ сервера, причем сервер не может прислать данные клиенту без запроса последнего. В дальнейшем мы откажемся от этих предположений.

Внесем в построенную модель клиента несколько уточнений (см. рис. 4). Во-первых, введем понятие состояния активности клиента. Будем говорить, что клиент находится в активном состоянии, если он нормально функционирует, способен генерировать запросы и обрабатывать ответы серверов. В противном случае будем говорить, что клиент находится в неактивном состоянии (например, отключен от сети).

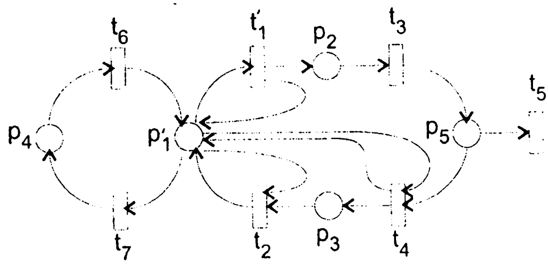


Рис. 4.

Для моделирования состояния активности клиента введем дополнительную позицию p_4, соответствующую неактивным клиентским процессам, а также переходы t_6 и t_7, моделирующие переход клиента из неактивного состояния в активное и обратно, соответственно. Кроме того, внесем несколько деталей в описание связи между клиентом и сервером. Для этого заменим переход t_3 тремя новыми переходами и одной позицией. Переход t_3 моделирует передачу запроса на обработку серверу. Позиция p_5 содержит запросы, обрабатываемые сервером. Переход t_4 моделирует передачу результата обработки сервером клиенту. Наконец, переход t_5 моделирует окончание обработки запроса без передачи ответа клиенту — такая ситуация возможна для некоторых типов запросов, вернее сказать, команд серверу; кроме того, этот переход может сработать, если к моменту окончания работы сервера клиент перешел в неактивное состояние и ответ не может быть ему передан.

Следующее уточнение касается ограничения емкостей буферов модели. Положим, что позиции p_2, p_3 и p_5 могут содержать не больше V_{out}, V_{in} и V_{serv} маркеров соответственно, где V_{out}, V_{in} и V_{serv} — некоторые натуральные числа. Известно, что исходная сеть Петри с ограничениями может быть преобразована в некоторую сеть Петри без ограничений введением ряда дополнительных позиций и переходов [3], таким образом, ограничение на емкость позиций не выводит модель из класса обыкновенных сетей Петри.

Начальная маркировка сети имеет вид $M_0 = (0, 0, 0, N, 0)$, т.е. в позиции p_4 имеется N маркеров, соответствующих N неактивным клиентским процессам, а остальные позиции пусты.

Покажем, что построенная модель клиента является ограниченной, обратимой и активной.

Ограниченность. Емкость позиций p_2, p_3 и p_5 ограничена соответственно величинами V_{out}, V_{in} и V_{serv} по построению сети. Заметим, что для позиций p_1 и p_4 при произвольном функционировании сети выполняется условие $M(p_1) + M(p_4) = const$, т.е. сумма маркеров в позициях p_1 и p_4 не изменяется. В самом деле, при срабатывании переходов t_1, t_2, t_3, t_4 и t_5 количество маркеров в позициях p_1 и p_4 не изменяется. При срабатывании перехода t_6 один маркер перемещается из позиции p_4 в p_1, при этом сумма маркеров в этих позициях остается неизменной. При срабатывании перехода t_7 один маркер перемещается обратно из p_1 в p_4, также не меняя сумму. Итак, при функционировании сети сумма маркеров в позициях p_1 и p_4 остается постоянной и равна сумме при начальной маркировке, то есть N; соответственно, количество маркеров в каждой из двух рассматриваемых позиций не может превышать N. Итак, сеть действительно является ограниченной.

Обратимость. Покажем, что для любой маркировки M, достижимой из начальной маркировки M_0, M_0 будет достижима из M.

Сначала покажем, что множество всех маркировок, достижимых из начальной, имеет вид

$$R(M_0) = \{(k, l, m, N - k, n) \mid \\ 0 \le k \le N,\ 0 \le l \le V_{out},\ 0 \le m \le V_{in},\ 0 \le n \le V_{serv}\}.$$

В самом деле, как было показано при доказательстве ограниченности сети, каждая достижимая маркировка входит в указанное множество. Обратно, для любой маркировки $M = (k, l, m, N - k, n)$ из указанного множества существует последовательность переходов $\sigma(M)$ такая, что $M_0 \xrightarrow{\sigma(M)} M$:

$$\sigma(M) = t_6\,(t_1 t_2 t_4)^m\,(t_1 t_2)^n\, t_1^l t_7 t_6^k$$

Итак, $R(M_0)$ действительно имеет указанный вид.

Для любой маркировки $M = (k, l, m, N-k, n) \in R(M_0)$ укажем последовательность переходов $\tau(M)$ такую, что $M \xrightarrow{\tau(M)} M_0$:

$$\tau(M) = \begin{cases} t_2^m (t_4 t_2)^n (t_3 t_4 t_2)^l t_7^k, & k > 0; \\ t_6 t_2^m (t_4 t_2)^n (t_3 t_4 t_2)^l t_7, & k = 0. \end{cases}$$

Итак, показано, что модель клиента является обратимой.

Активность. Покажем, что исследуемая сеть Петри является активной, т.е. для любой достижимой маркировки $M \in R(M_0)$ и любого перехода t имеется последовательность дальнейших переходов $\mu(M, t)$, включающая t.

Легко заметить, что для начальной маркировки M_0 и любого перехода t последовательность дальнейших переходов, включающая t, действительно существует:

$$\begin{aligned} &\mu(M_0, t_1) = t_6 t_1, \\ &\mu(M_0, t_2) = t_6 t_1 t_3 t_4 t_2, \\ &\mu(M_0, t_3) = t_6 t_1 t_3, \\ &\mu(M_0, t_4) = t_6 t_1 t_3 t_4, \\ &\mu(M_0, t_5) = t_6 t_1 t_3 t_5, \\ &\mu(M_0, t_6) = t_6, \\ &\mu(M_0, t_7) = t_6 t_7. \end{aligned}$$

Для произвольной достижимой маркировки M воспользуемся обратимостью сети. Последовательность переходов

$$\mu(M, t) = \tau(M) \mu(M_0, t)$$

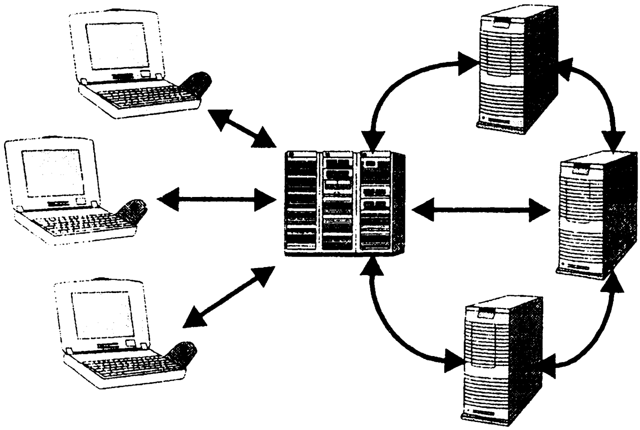
может быть выполнена, начиная с разметки M, и эта последовательность включает переход t.

Итак, показано, что построенная модель клиента действительно обладает свойствами ограниченности, обратимости и активности.

3. Модель сервера

Для построения дискретной динамической модели сервера рассмотрим сначала некоторую неформальную модель (рис. 5).

Обобщенный сервер принимает запросы клиентов, и либо непосредственно реагирует на них, выполняя какие-либо действия



Запросы клиентов

Взаимодействие с другими серверами

Рис. 5.

и/или посылая ответы, либо связывается с другими серверами, распределяя работу. Например, сервер, реализующий внутреннюю логику работы системы, может связываться с серверами баз данных, с почтовым сервером и т.д. Получив ответ от другого сервера, рассматриваемый сервер может тем или иным способом обработать его или использовать для подготовки ответа клиенту.

Кроме обработки запросов клиентов сервер может реагировать и на какие-либо внутренние события, например, на события, связанные со временем.

Сеть Петри, формализующая описанную неформальную схему, изображена на рис. 6.

Переход t_4 моделирует поступление клиентского запроса, переход t_3 — наступление внутреннего события сервера. Позиция p_1 содержит необработанные запросы (в том числе и внутренние). Переход t_5 соответствует передаче запроса на другой сервер, переход t_1 — подготовке ответа без обращения к другим серверам.

Переход t_6 соответствует получению данных с другого сервера. Позиция p_2 содержит необработанные ответы других серверов. Переход t_2 моделирует подготовку ответа клиенту с использованием данных, полученных от другого сервера. Позиция p_3

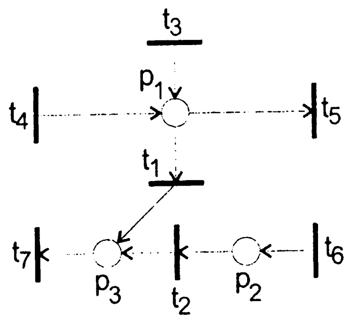


Рис. 6.

содержит ответы, готовые для отправки клиентам. Переход t_7 моделирует отправку ответа клиенту.

Естественно полагать, что емкость буферов сервера ограничена. Это предположение влечет ограничения на емкость позиций: $M(p_1) \le V^c_{in}$, $M(p_2) \le V^s_{in}$, $M(p_3) \le V_{out}$.

Начальная маркировка имеет вид $M_0 = (0,0,0)$, т.е. все буферы сервера пусты.

Исследуем свойства описанной модели.

Ограниченность. Очевидно, модель сервера ограничена по построению: $M(p_1) \le V^c_{in}$, $M(p_2) \le V^s_{in}$, $M(p_3) \le V_{out}$.

Обратимость. Покажем, что построенная сеть Петри обратима. Для этого сначала заметим, что множество маркировок, достижимых из начальной, имеет вид

$$R(M_0) = \{(k,l,m) \mid 0 \le k \le V^c_{in},\ 0 \le l \le V^s_{in},\ 0 \le m \le V_{out}\}.$$

В самом деле, любая маркировка, достижимая из начальной, принадлежит указанному множеству в силу ограниченности сети. Обратно, для любой маркировки $M = (k,l,m) \in R(M_0)$ укажем последовательность переходов $\sigma(M)$ такую, что $M_0 \xrightarrow{\sigma(M)} M$: $\sigma(M) = (t_3 t_1)^m\, t_6^l t_3^k$.

Таким образом, $R(M_0)$ действительно имеет указанный вид.

Для любой маркировки $M = (k, l, M) \in R(M_0)$ укажем последовательность переходов $\tau(M)$ такую, что $M \xrightarrow{\tau(M)} M_0$:

$$\tau(M) = t_7^m \, (t_2 t_7)^l \, (t_1 t_7)^k \, .$$

Итак, модель сервера обратима.

Активность. Для доказательства активности сервера воспользуемся, как и в случае с клиентом, его обратимостью. Для начальной маркировки M_0 и перехода t последовательностью последующих переходов $\mu(M_0, t)$, приводящей к срабатыванию t, будет

$$\begin{aligned}
\mu(M_0, t_1) &= t_3 t_1, \\
\mu(M_0, t_2) &= t_6 t_2, \\
\mu(M_0, t_3) &= t_3, \\
\mu(M_0, t_4) &= t_4, \\
\mu(M_0, t_5) &= t_3 t_5, \\
\mu(M_0, t_6) &= t_6, \\
\mu(M_0, t_7) &= t_3 t_1 t_7.
\end{aligned}$$

Для произвольной достижимой маркировки M и перехода t последовательностью последующих переходов, приводящей к срабатыванию t, будет $\mu(M, t) = \tau(M)\mu(M_0, t)$.

Итак, показано, что модель сервера обладает свойствами ограниченности, обратимости и активности.

4. Модель распределенной информационной системы

Используем описанные модели клиента и сервера как блоки для построения модели распределенной информационной системы. Как было указано в разделе 1, в такой системе имеется некоторое количество клиентов и серверов, каждый из клиентов может обращаться с запросами к одному и более серверам, каждый сервер принимает запросы от одного и более клиентов (возможно, клиентами являются другие сервера) и возвращает клиентам ответы.

Введем естественное предположение: будем считать, что моделируемая информационная система связна, не распадается на несколько совершенно независимых блоков. В противном случае построение дискретной модели следует проводить отдельно для каждого из связных компонент.

Формальное построение модели распределенной системы начнем с простейшей модели, включающей один клиент и один сервер. После этого некоторое количество раз может быть выполнено одно из следующих действий:

A) подключение нового клиента к серверу;

B) подключение нового сервера к клиенту;

C) подключение нового сервера к серверу;

D) введение новой прямой связи между клиентом и сервером;

E) введение новой обратной связи между клиентом и сервером;

F) введение новой прямой связи между двумя серверами;

G) введение новой обратной связи между двумя серверами.

Точное описание каждого из указанных действий будет дано ниже в ходе доказательства.

Покажем, что итоговая модель распределенной информационной системы сохраняет свойства ограниченности, обратимости и активности, свойственные базовым компонентам. Доказательство проведем индукцией по построению сети.

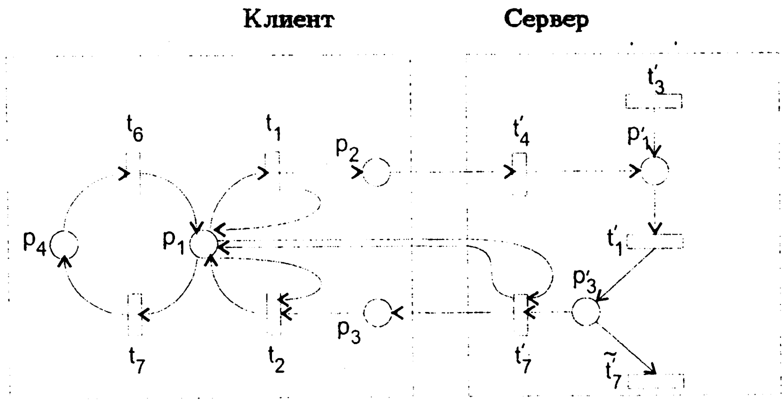


Рис. 7.

Базой индукции является простейшая модель, включающая одного клиента и один сервер.

При соединении моделей клиента и сервера у модели клиента (рис. 4) удаляются позиция p_5 и переход t_5, соответствовавшие недетализированному функционированию сервера, а переходы отправки запроса клиентом t_3 и получения ответа от сервера t_4 отождествляются с переходами t_4^l получения запроса от клиента и t_7^l отправки ответа клиенту модели сервера (рис. 6) соответственно. Кроме того, переход t_7^l модели сервера расщепляется на два — собственно t_7^l, отождествляющийся, как было указано, с переходом t_4 модели клиента, и $\tilde{t}_7^l$, моделирующий удаление из выходного буфера сервера ответа, который не может быть или не должен быть доставлен клиенту. Переходы t_2, t_5 и t_6, а также позиция p_2 модели сервера удаляются, так как они предназначены для моделирования взаимодействий с другими серверами, а рассматриваемая модель включает только один сервер.

Полученная в результате описанного процесса модель системы "клиент–сервер" показана на рис. 7.

Начальная маркировка модели $M_0 = (0, 0, 0, N, 0, 0)$, т.е. маркеры имеются только в позиции p_4, соответствующей неактивным клиентам. На емкость позиций накладываются те же ограничения, что и на емкость соответствующих позиций моделей клиента и сервера.

Ограниченность. Ограниченность базовой модели "клиент-сервер" следует из ограниченности моделей клиента и сервера.

Обратимость модели "клиент-сервер" доказывается аналогично обратимости моделей клиента и сервера. Укажем лишь основные моменты (в тех же обозначениях).

Множество достижимых маркировок

$$R(M_0) = \{(k, l, m, N-k, r, s) \mid 0 \le k \le N,\ 0 \le l \le V_{out}^{client},\ 0 \le m \le V_{in}^{client},\ 0 \le r \le V_{in}^{serv},\ 0 \le s \le V_{out}^{serv}\}.$$

Для $M = (k, l, m, N-k, r, s)$ имеем

$$\sigma(M) = t_6\left(t_1 t_4^l t_1^l t_7^l\right)^m t_1^l t_7 t_6^k \left(t_3^l t_1^l\right)^s \left(t_3^l\right)^r, \text{ где } M_0 \xrightarrow{\sigma(M)} M;$$

$$\tau(M) = \begin{cases} t_2^m t_7^k \left(\tilde{t}_7^l\right)^s \left(t_1^l \tilde{t}_7^l\right)^r \left(t_4^l t_1^l \tilde{t}_7^l\right)^l, & k > 0, \\ t_6 t_2^m t_7 \left(\tilde{t}_7^l\right)^s \left(t_1^l \tilde{t}_7^l\right)^r \left(t_4^l t_1^l \tilde{t}_7^l\right)^l, & k = 0. \end{cases}$$

где $M \xrightarrow{\tau(M)} M_0$.

Таким образом, модель обратима.

Активность. Покажем, что модель "клиент–сервер" активна, воспользовавшись обратимостью. Доказательство проводится аналогично моделям клиента и сервера; в тех же обозначениях имеем

$$\begin{aligned}
&\mu(M_0, t_1) = t_6 t_1,\\
&\mu(M_0, t_2) = t_3 t_1 t_4' t_1' t_7' t_2,\\
&\mu(M_0, t_6) = t_6,\\
&\mu(M_0, t_7) = t_6, t_7\\
&\mu(M_0, t_1') = t_3' t_1',\\
&\mu(M_0, t_3') = t_3',\\
&\mu(M_0, t_4') = t_6 t_1 t_4',\\
&\mu(M_0, t_7') = t_3' t_1' t_7',\\
&\mu(M_0, \tilde{t}_7') = t_3' t_1' \tilde{t}_7';
\end{aligned}$$

$$\mu(M, t) = \tau(M)\mu(M_0, t).$$

Итак, модель "клиент–сервер", являющаяся базой при индуктивном построении модели распределенной информационной системы, имеет свойства ограниченности, обратимости и активности.

Рассмотрим теперь подробно каждый из случаев "A"–"G", которые могут встретиться при конструировании компонентной модели, и покажем, что в каждом из этих случаев ограниченность, обратимость и активность сети Петри сохраняются.

Случай А — подсоединение нового клиента к серверу. Порядок подсоединения изображен на рис. 8А, значение позиций и переходов очевидно из сравнения с моделью "клиент–сервер" (рис. 7). Начальная маркировка старых позиций сохраняется неизменной, начальная маркировка новых позиций p_1, p_2, p_3 и p_4 — $(0, 0, 0, N)$, где N — некоторое натуральное число. На новые позиции накладываются такие же ограничения по емкости, как и на соответствующие позиции модели одиночного клиента.

Ограниченность модели сохраняется, так как позиции p_2 и p_3 ограничены по построению, а сумма маркеров в позициях p_1 и p_4 остается постоянной (показывается так же, как для модели одиночного клиента).

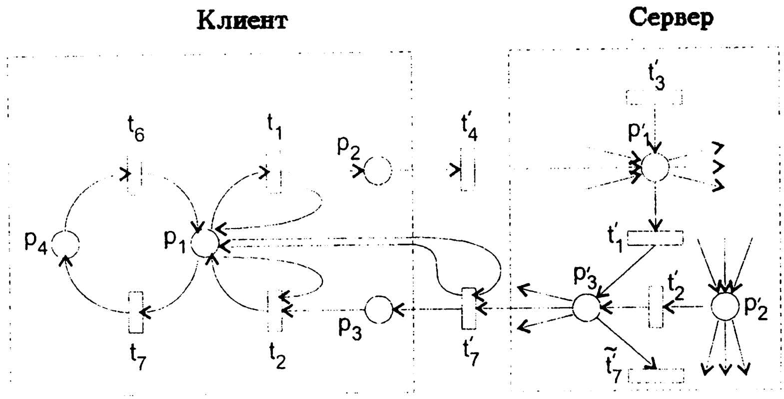


Рис. 8А.

Обратимость. Пусть $M = ((M')\,k, l, m, N-k)$ — произвольная достижимая маркировка расширенной сети, где M' — некоторая достижимая маркировка исходной подсети. Тогда существует последовательность переходов $\tau(M)$ такая, что $M \xrightarrow{\tau(M)} M_0$:

$$\tau(M) = \begin{cases} \tau'(M')t_2^m\,(t_4't_1't_7't_2)^l\,t_7^k, & k > 0, \\ \tau'(M')t_6t_2^m\,(t_4't_1't_7't_2)^l\,t_7, & k = 0, \end{cases}$$

где $\tau'(M')$ — последовательность переходов, переводящая маркировку M' в начальную маркировку M_0' в исходной подсети.

Активность. Для всех переходов исходной подсети активность следует из индукционного предположения. Для новых переходов имеем $\mu(M) = \tau(M)t_6t_1t_4't_1't_7't_2t_7$ —последовательность переходов, начинающаяся с произвольной достижимой маркировки M, в которой каждый новый переход срабатывает по крайней мере один раз.

Случай B — подсоединение нового сервера к клиенту. Порядок подсоединения изображен на рис. 8B, значение позиций и переходов очевидно из сравнения с моделью "клиент–сервер" (рис. 7). Начальная маркировка старых позиций сохраняется неизменной, новые позиции при начальной маркировке пусты. На новые позиции накладываются такие же ограничения по емкости, как и на соответствующие позиции модели одиночного сервера.

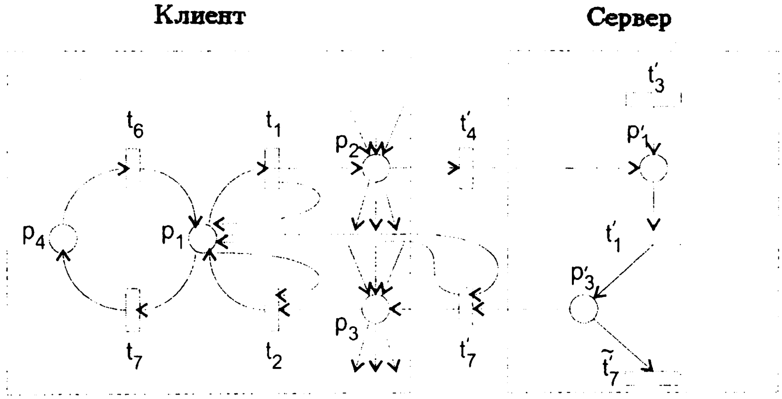


Рис. 8В.

Ограниченность следует из ограниченности исходной подсети и ограниченности новых позиций по их построению.

Обратимость показывается аналогично случаю А. В тех же обозначениях для $M = ((M'), k, l)$ имеем

$$\tau(M) = \tau'(M')\left(\tilde{t}'_7\right)^l \left(t'_1\tilde{t}'_7\right)^k.$$

Активность. Для всех переходов исходной подсети активность следует из индукционного предположения. Для новых переходов имеем $\mu(M) = \tau(M)t_6 t_1 t'_4 t'_1 t'_7 t'_2 t_1 \tilde{t}'_7$ —последовательность переходов, начинающаяся с произвольной достижимой маркировки M, в которой каждый новый переход срабатывает по крайней мере один раз.

Случай С — подсоединение нового сервера к серверу. Порядок подсоединения изображен на рис. 8С (новый сервер изображен справа), значение позиций и переходов очевидны из сравнения с моделью одиночного сервера (рис. 6). Необходимо сделать замечание: если блок исходного сервера (на рисунке — слева) не имел ранее связей с другими серверами, то к нему будут добавлены новые позиция p_2 и переход t_2, моделирующие очередь сообщений от других серверов и подготовку ответа клиенту с использованием сообщения от другого сервера соответственно (аналогично модели одиночного сервера).

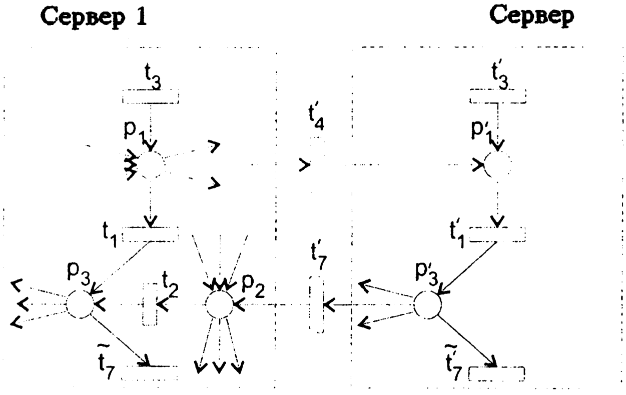


Рис. 8С.

На новые позиции p'_1, p'_3 (и, возможно, на новую позицию p_2) налагаются те же ограничения, что и в модели одиночного сервера. При начальной маркировке новые позиции пусты.

Ограниченность следует из ограниченности исходной подсети и ограниченности новых позиций по их построению.

Обратимость. Аналогично предыдущим случаям, для произвольной достижимой маркировки $M = ((M'), k, l)$ имеем $\tau(M) = \tau'(M')\left(\tilde{t}'_7\right)^l \left(t'_1\tilde{t}'_7\right)^k$.

Активность. Аналогично предыдущим случаям, имеем $\mu(M) = \tau(M)t_3t'_4t'_1t'_7t'_3t'_1\tilde{t}'_7$ —последовательность переходов, начинающаяся с произвольной достижимой маркировки M, в которой каждый новый переход срабатывает по крайней мере один раз.

Случай D — введение новой прямой связи клиента и сервера — изображен на рис. 8D. Вводится новый переход t и дуги A, B, связывающие буфер исходящих запросов клиента и буфер входящих сообщений сервера.

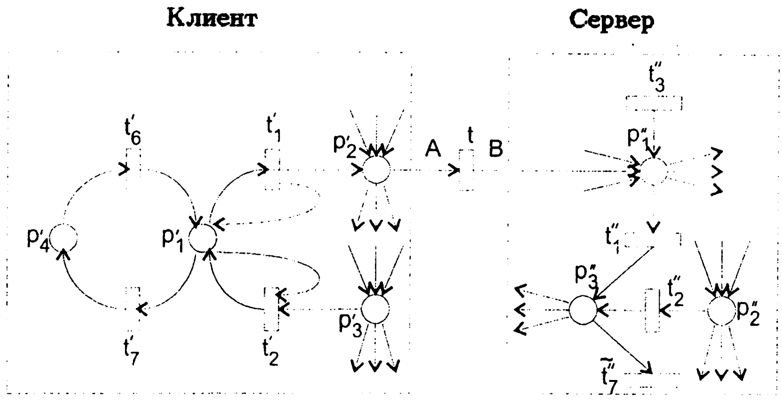


Рис. 8D.

Ограниченность и *обратимость* сети следуют из ограниченности и обратимости исходной подсети.

Активность. Аналогично предыдущим случаям, имеем $\mu(M) = \tau(M)t'_6t'_1t$, где $M \xrightarrow{\tau(M)} M_0$, $\tau(M)$ существует, так как сеть обратима.

Случай E — введение новой обратной связи клиента и сервера — изображен на рис. 8E. Вводится новый переход t и дуги A, B, C и D, связывающие буфер исходящих сообщений сервера, буфер входящих сообщений клиента, а также позицию, содержащую активные клиентские процессы.

Ограниченность и *обратимость* сети следуют из ограниченности и обратимости исходной подсети.

Активность. Аналогично предыдущим случаям, имеем $\mu(M) = \tau(M)t''_3t''_1t$, где $M \xrightarrow{\tau(M)} M_0$, $\tau(M)$ существует, так как сеть обратима.

Случай F — введение новой прямой связи двух серверов — изображен на рис. 8F. Вводится новый переход t и дуги A, B, соединяющие буфера входящих запросов двух серверов.

Ограниченность и *обратимость* сети следуют из ограниченности и обратимости исходной подсети.

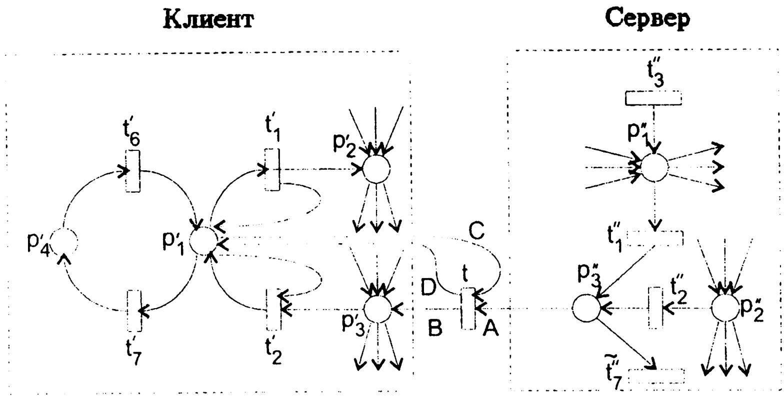


Рис. 8E.

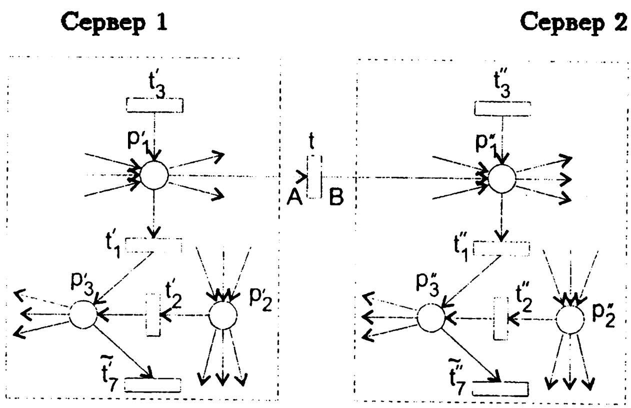


Рис. 8F.

Активность. Аналогично предыдущим случаям, имеем $\mu(M) = \tau(M)t_3t$, где $M \xrightarrow{\tau(M)} M_0$, $\tau(M)$ существует, так как сеть обратима.

Случай G — введение новой обратной связи двух серверов — изображен на рис. 8G. Вводится новый переход t и дуги A, B. Если сервер 1 не имел ранее входящих результатов обработки от других серверов, вводится также позиция p'_2 и переход t'_2, а также связывающие их дуги (значение этих элементов оче-

видно из сравнения с моделью одиночного сервера, рис. 6). Если вводится новая позиция p'_2, то на нее накладывается такое же ограничение, как и в модели одиночно сервера; при начальной маркировке она пуста.

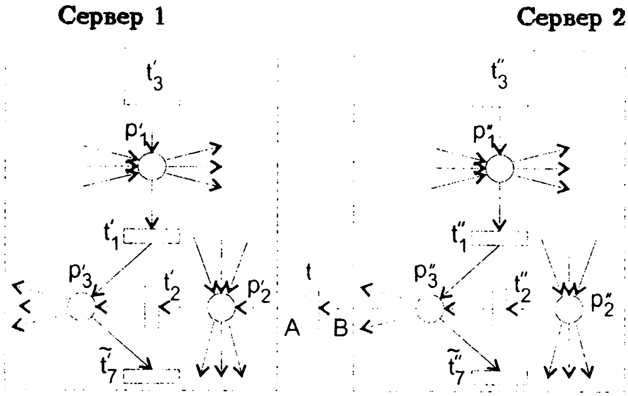


Рис. 8G.

Ограниченность модели следует из ограниченности исходной подсети и ограничения на емкость позиции p'_2 при ее введении.

Обратимость. Если позиция p'_2 не вводится, то обратимость сети следует из обратимости исходной подсети. Если позиция p'_2 вводится, то для произвольной достижимой $M = ((M'), k)$, где M' — некоторая достижимая маркировка исходной подсети, а k — количество маркеров в p'_2, последовательность переходов $\tau(M) = \tau'(M')\left(t'_2\tilde{t}'_7\right)^k$ возвращает модель к начальной маркировке.

Активность. Аналогично предыдущим случаям, имеем $\mu(M) = \tau(M)t''_3t''_1tt'_2$, где $M\xrightarrow{\tau(M)}M_0$, $\tau(M)$ существует, так как сеть обратима.

Рассмотрев все случаи, возможные при построении модели, мы показали сохранение при них свойств ограниченности, обратимости и активности. Тем самым показано, что предложенная компонентная дискретная динамическая модель распределенной информационной системы будет ограничена, обратима и активна при произвольной структуре моделируемой системы.

5. Возможные расширения модели

Расширение предложенной модели возможно как в рамках обыкновенных сетей Петри, так и с выходом в другие классы сетей.

Оставаясь в рамках обыкновенных сетей Петри, можно углубиться в моделирование деталей архитектуры отдельных клиентов и серверов, что позволит получить более точные модели систем.

Из расширений сетей Петри весьма удобным представляется использование временных стохастических сетей, позволяющих моделировать случайные временные задержки. Недостатком таких сетей является невозможность их аналитического исследования, приводящая к необходимости применения имитационного моделирования.

Литература

1. PETRI C.A. "Kommunikation mit Automaten". — Bonn. — Institut für instrumentelle Mathematik, 1962.

2. JOHNSONBAGH R., MURATA T. "Petri nets and marked graphs — mathematical models of concurrent computations" //The American Math. Monthly, Oct. 1982. —Vol. 89, №. 8. — P. 552–566.

3. МУРАТА Т. "Сети Петри: Свойства, анализ, приложения" //ТИИЭР. — 1989. — Т. 77, № 4. — С.

Поступила в редакцию
24 декабря 2001 года