

Спецификация и моделирование распределенных информационных систем на основе сетей Петри

Зыбарев Ю.М., Зыбарев Е.Ю., Чернев С.П.

Институт вычислительной математики и математической геофизики СО РАН

1. Введение

Концепция разработки современных информационных систем опирается на методы и инструментарий современной информационной логики, архитектуру «клиент-сервер», сервисно-ориентированные архитектуры, возможности глобальной сети Интернет, сетевых и Web-технологий. Распределенные многопользовательские информационные системы реализуются на базе сетевой инфраструктуры в виде виртуальных информационных сред. С интенсивным развитием Web-технологий одной из ключевых подсистем виртуальной информационной среды стал информационный портал, который ориентирован на интеграцию различных проблемно-ориентированных информационных систем, сервисов и информационных ресурсов с организацией консолидированной точки доступа к ним пользователей различных целевых групп с учетом их полномочий и решения задач информационной безопасности.

Одной из значимых проблем эффективного построения и функционирования на всех стадиях жизненного цикла информационной среды является сбалансированность всех её компонент с учетом архитектурных и технологических решений, архитектурных и ресурсных характеристик сетевой и серверной среды, возможностей масштабирования решений. Данный класс задач трудно решать без развитых средств спецификации и моделирования. Для решения задач анализа и выбора проектных решений при проектировании сложных динамических систем одним из наиболее эффективных инструментов является имитационное моделирование [1, 2]. Современные системы имитационного моделирования основаны на различных математических схемах. Для анализа и проектирования различных систем широкое распространение получил аппарат сетей Петри [2–5]. Наличие различных обобщений (классов) сетей Петри делает возможным на их основе осуществлять спецификацию сложных систем в виде математического (формализованного) представления, что дает возможность качественного анализа. Анализ сетей Петри и их реализация в виде компьютерной имитационной модели помогает получить важную информацию о структуре и динамическом поведении моделируемой системы и оценивать варианты организации системы с целью поиска наиболее эффективных решений.

В данной статье рассматриваются вопросы спецификации и моделирования на основе сетей

Петри распределенных информационных систем, реализуемых в архитектуре «клиент-сервер».

2. Определения и краткая характеристика сетей Петри

В классическом варианте [3, 4] простой сетью Петри называется набор $\langle P, T, A, F, M_0 \rangle$, где:

1. $P = \{p_1, \dots, p_m\}$ – множество позиций, соответствующее множеству условий (состояний) в системе.
2. $T = \{t_1, \dots, t_n\}$ – множество переходов, соответствующее множеству процессов моделируемой системы.
3. $A = \{(p_i, t_j) \text{ или } (t_k, p_l) | t_j, t_k \in T, p_i, p_l \in P\}$ – конечное множество дуг сети.
4. F – функция кратности дуг, которая ставит в соответствие каждой дуге (p_i, t_j) (или (t_k, p_l)) неотрицательное целое число f_j (или f_{kl} соответственно).
5. M_0 – функция начальной маркировки сети Петри, которая ставит в соответствие каждой позиции $p \in P$ целое неотрицательное число $M(p)$; таким образом, маркировка сети Петри представляется вектором $M = (M(p_1), M(p_2), \dots, M(p_m))$.
6. В маркированной сети происходит движение маркеров, которое подчиняется следующим правилам – законам функционирования сети Петри:
 - Переход $t_j \in T$ считается возбужденным при маркировке M , если в каждой его входной позиции $p_i \in \bullet t_j$ число маркеров равно или превышает кратность дуги f_j или в векторном виде $M \geq F(_, t_j)$, где $F(_, t_j) = (F(p_1, t_j), \dots, F(p_m, t_j))$.
 - Возбужденный при маркировке M переход t_j срабатывает, изымая из каждой его входной позиции $p_i \in \bullet t_j$ f_{ij} маркеров и помещая в каждую его выходную позицию $p_k \in t_j \bullet$ f_{jk} маркеров. При этом маркировка M заменяется на M' , что обозначается $M|t_j$, причём $M' = M - F(_, t_j) + F(t_j, _)$, где вектор $F(t_j, _)$ определен по аналогии с вектором $F(_, t_j)$.
 - Переход не может находиться в состоянии возбуждения бесконечно долго; он должен либо сработать, либо возбуждение с него должно быть снято срабатыванием другого перехода.
 - Срабатывание происходит мгновенно, а в один и тот же момент времени срабатывает только один переход.

Как видим, сеть Петри представляется ориентированным маркированным двудольным мульт-

тиграфом, который имеет вершины двух типов: вершины-позиции, их множество $P = \{p_1, \dots, p_m\}$ и вершины-переходы, их множество $T = \{t_1, \dots, t_n\}$. Дуги графа могут связывать только разнотипные вершины, при этом кратность дуги из позиции в переход равна $F(p, t)$, а кратность дуги из перехода в позицию – $F(t, p)$ (если $F(p, t) = 0$ или $F(t, p) = 0$, то дуги не существует). Графовое представление сетей Петри удобно для визуального анализа моделируемой системы. При графическом представлении сети Петри позиции изображают в виде кружочков, а переходы – в виде черточек или прямоугольников, дуги помечаются значениями функций $F(p, t)$ и $F(t, p)$. Маркировка на графе обозначается точками (фишками или маркерами) в позициях.

Для спецификации и моделирования сложных систем возможностей простых оказывается недостаточно, поэтому в аппарате сетей Петри сформулированы и развиваются различные варианты их обобщения: раскрашенные, временные, иерархические и т.д. В частности, используемая в данной статье модификация раскрашенных сетей Петри оперирует с маркерами различных цветов. В отличие от простых сетей Петри, эта модель снабжена функцией охраны переходов и функцией цвета $C: P \rightarrow \Sigma$. Функция охраны переходов позволяет переходу выполниться только при соблюдении определённых дополнительных условий. Функция цвета определяет допустимое множество цветов $C(p)$ маркеров для каждой позиции $p \in P$. В результате функция кратности дуг усложняется, превратившись в функцию выражений дуг, которая фиксирует не только число маркеров, но и их цвет. Начальная разметка раскрашенной сети Петри реализуется в виде функции, которая каждой позиции ставит в соответствие набор маркеров различных (определённо заданных) цветов в виде мультимножества, определённом над $C(p)$, $p \in P$.

В раскрашенной сети движение маркеров определяется совокупностью предикатов/функций, определённых в пространстве мультимножеств $(C(p), p \in P)$ над заданным множеством цветов:

- при этом функции охраны переходов задают условия возбуждения переходов, проверяющие для текущей маркировки сети наличие необходимых наборов мультимножеств (в соответствии с допустимыми для рассматриваемой позиции цветами) и их количественных характеристик для сопряжённых (входных и выходных) с выделенным переходом позиций;
- для возбуждённых переходов (т.е. с истинными значениями функции охраны переходов) правила (алгоритмы) их срабатывания определяют какого цвета и в каком количестве маркеры из входных позиций изымается и что поступает в выходные позиции.

Более подробную информацию о раскрашенных и других классах сетей Петри можно найти в [3, 4]. Для спецификации и моделирования рассма-

триваемого нами класса информационных систем использован класс раскрашенных временных сети Петри.

3. Спецификация распределённых информационных систем, реализованных в архитектуре «клиент-сервер», на основе сетей Петри

Рассматриваемый класс распределённых информационных систем и информационных сред реализуется на базе сетевых технологий и технологий Интернет в архитектуре «клиент-сервер». Модель такой архитектуры описывает процессы функционирования и взаимодействия двух подсистем: «клиентской» и «серверной». В общем случае, моделируемая информационная среда содержит некоторое (заданное) количество «клиентов» и «серверный кластер», при этом каждый из «клиентов» может обращаться с запросами к одному или более серверам, а каждый сервер серверного кластера взаимодействует с разными «клиентами» и другими серверами.

«Клиент», как правило, реализует синхронный или асинхронный режим функционирования. При синхронном режиме «клиент» посылает запрос «серверу» и ожидает ответа и только после получения ответа продолжает работу. Для асинхронного режима «клиент», послав запрос «серверу», может не ожидать ответа и продолжить работу, после получения ответа обработать его. Серверная подсистема обрабатывает информацию, обеспечивает поддержку базы данных, формирует информационные сообщения для «клиентов», принимает и обрабатывает запросы «клиентов». Если серверная подсистема реализована в виде серверного кластера, т.е. в виде нескольких взаимосвязанных серверов (например, сервер общего назначения, сервер баз данных, почтовый сервер и т.д.), то реализуются процессы взаимодействия серверов. Кроме этого, серверная подсистема может реализовывать внутрисистемные процессы.

3.1. Базовая модель в виде сети Петри

Базовая модель распределённой информационной среды, реализованной на базе сетевых технологий и технологий Интернет в архитектуре «клиент-сервер», представлена в виде сети Петри на рис. 1.

Спецификация позиций сети Петри для модели информационной системы в архитектуре клиент-сервер.

В данной сети Петри все маркеры характеризуются цветом (ID, Data). При этом ID определяет идентификатор клиента (целое число), Data – непосредственно полный набор данных передаваемой информации между «клиентом» и «сервером» в парах:

- $Data = list\ of\ (ids, key, value)$, где: ids – идентификатор набора данных в подсистеме клиента (целое число); key – название ключевого зна-

чения (строковый тип); *value* – значение поля (двоичный набор данных).

Рассмотрим позиции $P = \{p_0, \dots, p_m\}$, маркеры которых интерпретируются следующим образом:

p_0 – маркеры данной позиции содержат в себе информацию каждого приложения «клиента». Фактически, данная позиция представляет собой локальную базу данных «клиента». Наличие поля ID в цвете маркера позволяет отличать данные различных клиентов при моделировании, а *ids* в поле Data – наборы данных.

p_1 – маркеры, каждый из которых интерпретируется как программный модуль «клиента», реализующий активный клиентский процесс.

p_2 – позиция, маркеры которой интерпретируются как исходящие запросы, т.е. информационные сообщения сформированные «клиентом» для передачи серверной подсистеме. При передаче информации каждый маркер несет в себе информацию о номере «клиента», инициировавшего передачу, в поле ID. При необходимости, устанавливается ненулевое значение *ids* в Data.

p_3 – маркеры данной позиции интерпретируются как сообщения, поступающие «клиенту» от «сервера».

p_4 – каждый из маркеров в данной позиции интерпретируется как программный модуль из библиотеки «клиента».

p_5 – маркеры данной позиции интерпретируются как сообщения, поступившие от «клиентов» к «серверу».

p_6 – маркеры этой позиции соответствуют информационным сообщениям, сформированным «сервером» для «клиентов». При этом каждый маркер содержит непустое поле ID, фактически определяющее получателя для данного сообщения.

p_7 – каждый из маркеров в данной позиции интерпретируется как программный модуль из библиотеки «серверной подсистемы», соответствующий активному (выполняемому) процессу.

p_8 – маркеры данной позиции интерпретируются как программные модули из библиотеки «серверной подсистемы».

p^* – маркеры данной позиции содержат в себе информацию каждого серверного приложения. Фактически, данная позиция представляет собой базу данных (локальную или глобальную) «сервера».

pr_s – маркеры данной позиции соответствуют спецификации вычислительных ресурсов серверного кластера (т.е. характеризуют архитектуру и производительность компьютерного кластера).

pr_k – маркеры данной позиции характеризуют ресурсные возможности сетевой инфраструктуры для работы с клиентом с номером ID, в частности, это может быть пропускная способность канала и т.д.

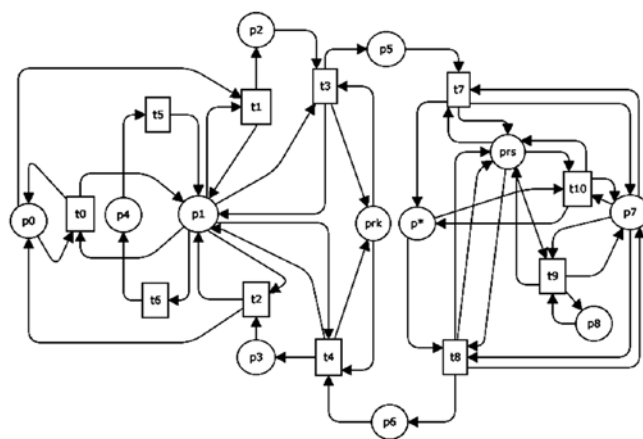


Рис. 1. Сеть Петри для информационной системы, реализованной в архитектуре «клиент-сервер»

Переходы и правила срабатывания для них сети Петри для модели информационной системы в архитектуре «клиент-сервер»:

Переходы $T = \{t_0, \dots, t_m\}$ данной сети Петри описывают (моделируют) следующие процессы информационной системы:

t_0 – этот переход моделирует работу проблемной системы на стороне клиента. Его работа перехода определяется бизнес-логикой «клиента». При срабатывании данного перехода происходит формирование набора записей Data, в зависимости от внешних условий или внутренних вычислений в переходе. Важно отметить, что ни один переход данной сети не меняет значение поля ID.

t_1 – переход моделирует формирование запроса «клиента» к «серверу».

t_2 – этот переход моделирует процесс обработки «клиентом» сообщений, поступивших от «сервера».

t_3 – переход реализует процесс передачи информационных сообщений, сформированных «клиентом», серверной подсистеме.

t_4 – переход реализует процесс передачи «клиентам» информационных сообщений, сформированных «сервером».

t_5 и t_6 – переходы, моделирующие процессы смены состояния «клиента», соответственно, из пассивного состояния в активное и обратно.

t_7 – данный переход моделирует процесс приема сервером поступающих от «клиентов» информационных сообщений и их первичной обработки.

t_8 – срабатывание данного перехода означает выполнение процесса формирования «сервером» информационных сообщений для «клиентов».

t_9 – переход, моделирующий работу внутрисистемных процессов серверной подсистемы по обработке запроса «клиента» с номером ID.

t_{10} – моделирует работу функциональной подсистемы проблемной системы, реализованной на серверном кластере. Ниже в работе рассмотрим декомпозицию данного перехода.

Времена выполнения каждого перехода могут быть определены фиксированными величинами, что фактически означает независимость продол-

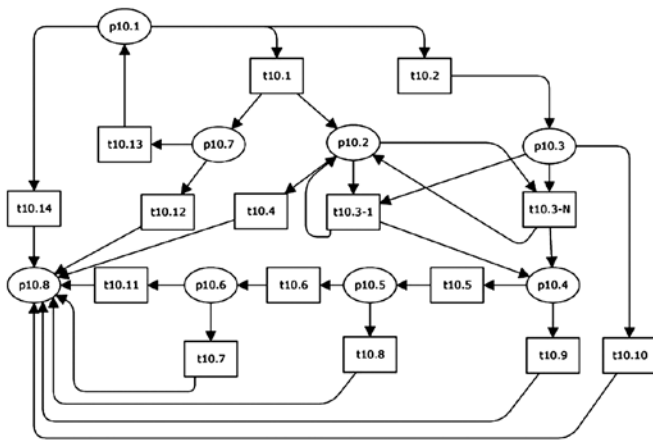


Рис. 2. Обобщенная модель информационного портала

жительности обработки запроса от сложности задачи, или рассматриваться в виде случайной величины. При этом наличие переходов prg и prk в модели позволяет учитывать загруженность сетевой инфраструктуры и наличие свободных ресурсов вычислительной системы серверного кластера.

3.2. Модель информационного портала – одной из основных составляющих серверной подсистемы информационной системы

На рис. 2 представлена обобщенная модель информационного портала в виде сети Петри. По существу, данная модель является более детализированным описанием перехода t_{10} сети Петри, представленной в п 3.1 (см. рис. 1)

В данной сети Петри маркеры могут иметь цвета двух типов: (ID, Req) и (ID, Data). В обоих случаях ID обозначает идентификатор сетевого соединения, по этому идентификатору Web-сервер определяет, кому направить ответные данные. Req – это сам запрос, Data – полученные данные в результате обработки запроса сервером. Для описания процесса функционирования сети определим допустимые в сети цвета:

- $ID = (Id_client, Id_req, Full, Number, Id_app)$, где: Id_client – целое неотрицательное число; Id_req – целое неотрицательное число; $Full$ – логического типа с набором значений {true, false}; $Number$ – целое неотрицательное число; Id_app – целое неотрицательное число;
- $Req = list\ of\ (key, value)$, где: key – название ключевого значения (строковый тип); $value$ – значение поля (двоичный набор данных).;
- $Data = list\ of\ (ids, key, value)$, где: ids – идентификатор набора данных в приложении (целое число); key – название ключевого значения (строковый тип); $value$ – значение поля (двоичный набор данных).

Спецификация позиций сети Петри для модели информационного портала:

p10.1 – маркеры позиции соответствуют запросам к portalу от пользователя, т.е. появление нового маркера вызывается внешней по отношению к

portalу системой; маркеры этой позиции характеризуются цветом (ID, Req).

p10.2 – позиция, где маркеры описывают сохраненные результаты заполнения промежуточных форм и промежуточных результатов работы приложений; при этом цвет маркетов позиции определяется как (ID, Req).

p10.3 – позиция, маркеры которой инициируют запуск приложения, и их цвет фишек определяется как (ID, Data).

p10.4 – маркеры позиции характеризуют наличие результата работы приложения и возможность дальнейшего преобразования ответа системами персонализации и кастомизации; цвета маркетов позиции определяются как (ID, Data).

p10.5 – маркеры позиции определяют наличие персонализированного ответа для пользователя, цвет маркетов этой позиции имеют вид (ID, Data).

p10.6 – маркеры позиции характеризуют наличие кастомизированного ответа для пользователя; позиция, так же как и позиция **p10.5**, содержит маркеры цвета (ID, Data), включающие набор данных в дальнейшем отображаемых для пользователя.

p10.7 – маркеры позиции определяют обязательность передачи дополнительных данных пользователем, т.е. при наличии маркетров в этой позиции портал генерирует форму для пользователя и требует ее заполнения; цвет маркетров позиции определяется как (ID, Data), а записи поля Data определяют характеристики формы дополнительных запросов пользователю.

p10.8 – маркеры этой позиции определяют результат работы portalа по предоставлению информации пользователю и имеют цвет (ID, Data), а данные поля Data есть законченный HTML-документ.

Переходы и правила срабатывания для них сети Петри для модели информационного portalа:

t10.1 – переход выполняет анализ маркетров позиции **p10.1** на основе поля *Full* и срабатывает только в том случае, если поле *Full* для маркетров входной позиции имеет значение *false*. В этом случае, переход копирует маркер в позицию **p10.2** и создает новый маркер в позиции **p10.7** со значением $Number = Number + 1$ и полем *Data*, содержащим дополнительную информацию для пользователя, которому отобразится форма.

t10.2 – переход срабатывает в случае поступления завершенного ответа от пользователя в том случае, если поле *Full* для маркера позиции **p10.1** будет иметь значение *true*; в этом случае переход изымает маркер из входной позиции и создает маркер в позиции **p10.3**, наличие которого инициирует процедуру запуска одного из приложений.

Переходы (**t10.3-1**)–(**t10.3-N**) моделируют работу информационных сервисных служб portalа (подсистемы новостей, документов, форумов, почтовой службы и т.д.) и проблемной информационной системы серверной подсистемы виртуальной

информсреды. В качестве требования к срабатыванию переходов этой группы является наличие маркеров в позиции $p_{10.3}$ и в позиции $p_{10.2}$ со значением поля Id_App равному идентификатору приложения App_N . Работа перехода ($t_{10.3-i}$) определяется бизнес-логикой соответствующего i -го приложения. Обязательным требованием для приложения является требование к подготовке ответа в виде маркера для позиции $p_{10.4}$.

$t_{10.5}$ – переход моделирует работу компоненты (подсистемы) персонификации и требованием к срабатыванию перехода является наличие маркера в позиции $p_{10.4}$, а результатом работы является создание маркера (ID, Data) в позиции $p_{10.5}$.

$t_{10.6}$ – переход моделирует работу компоненты (подсистемы) кастомизации и срабатывает при наличии маркера в позиции p_5 и выполняет подготовку кастомизированного ответа в позицию p_6 .

$t_{10.11}$ – переход выполняет визуализацию ответа, находящегося в поле Data, позиции $p_{10.6}$. На основе данных выполняется подготовка отображаемого ответа в формате HTML или в виде бинарных данных мультимедиа-потока.

$t_{10.13}$ – переход, моделирующий подготовку ответа пользователя, на основе данных, запрашиваемых сервером в позиции $p_{10.7}$, т.е. переход моделирует работу пользователя с дополнительными данными, запрашиваемыми порталом.

$t_{10.4}$, $t_{10.7}$, $t_{10.8}$, $t_{10.9}$, $t_{10.10}$, $t_{10.12}$ и $t_{10.14}$ – переходы – таймеры, которые реализуют механизмы удаления устаревшей информации из позиций в тех случаях, если информационные сообщения, моделируемые соответствующими маркерами из входных позиций, не обрабатываются в течение времени больше заданной пороговой величины базовыми функциональными подсистемами портала (переходами приложений, кастомизации и др.). Функция выражения дуг для выходных дуг перехода $t_{10.4}$ равна пустому мультимножеству, а для остальных – (ID, Data), где Data имеет вид отрицательного ответа.

В данном варианте сети все переходы-таймеры имеют заданные времена задержки при выполнении, которые равны времени ожидания ответа, а все остальные переходы выполняются сразу после активизации.

Практическая реализация. На основе предложенных сетей Петри для моделирования распределенных многопользовательских информационных систем в программной инструментальной системе Design/CPN разработаны компьютерные модели, которые позволяют анализировать проектные решения с учетом параметров информационной среды (характеристик информационной системы, состава и архитектуры серверного кластера, емкости используемых каналов передачи данных и т.д.). Такой анализ позволяет дать оценку масштабированных вариантов проектных решений сбалансированных относительно основных компонент рассматриваемой информационной системы. Разработанные модели применялись

в рамках проекта INTAS Ref. № 1000013-9357 «Distributed (Internet-based) automated system for acquiring and analysis of eye tracking data» при создании виртуальной информационной среды в сети Интернет, ориентированной на организацию информационно-технологической поддержки проведения прикладных исследований на основе регистрации параметров взгляда испытуемого (оператора) при его работе с информацией, размещенной на экране монитора.

Литература

1. Bulgren W.G. Discrete system simulation. – Englewood Cliffs: Prentice-Hall, 1982.
2. Bernard P. Zeigler, Tag Gon Kim, Herbert, Theory of Modeling and Simulation. 2 edition, Academic Press, 2007.
3. Jensen K. Coloured Petri Nets – Berlin a. o.: Springer-Verlag, 1996.
4. Зыбарев Е.Ю., Зыбарев Ю.М. Сети Петри как язык спецификации дискретных систем. Сборник трудов «Теория вычислений и языки спецификаций». – Новосибирск: изд-во ИМ СО РАН, 1995. – С. 186-201.
5. Зыбарев Ю.М., Чернев С.П. Спецификация функциональной модели информационного портала сетями Петри, Исследовано в России, 2003.