

УДК 004.942

И.В. Бухтияров

Сети Петри как инструмент спецификации специализированной сервис-ориентированной информационной среды для производства программных продуктов

Рассматриваются вопросы спецификации в виде высокоуровневых сетей Петри функциональной модели специализированной сервис-ориентированной информационной среды в Интернет/Интранет для организации производства программных продуктов виртуальными коллективами специалистов. В основе использования функциональных возможностей сервисов предложенной среды лежит проектный подход, цель реализации которого заключается в получении конечного продукта. В рамках предложенного подхода, главным объектом проектной деятельности, является операционная задача, и завершение проекта возможно только после итерационного выполнения набора жизненных циклов всех задач. Сети Петри рассматриваются не только как средство спецификации, но и в качестве инструмента имитационного моделирования, которое реализовано за счет применения программного инструментария CPN Tools. С помощью проведения численных экспериментов на базе данного программного обеспечения удалось получить различные количественные оценки параметров и свойств разрабатываемых аппаратных и программно-технологических решений.

Ключевые слова: производство программного продукта, проектная деятельность, функциональная модель, сети Петри, имитационное моделирование.

В настоящее время весомый вклад в динамичное развитие индустрии разработки программного обеспечения вносят малые и средние ИТ-компании, успешная деятельность которых существенным образом зависит от организации информационной и инструментально-технологической поддержки проектной деятельности команд специалистов. Особый интерес представляет организация работы при выполнении проекта по производству программного продукта территориально разобщенными коллективами. В связи с этим актуальной задачей является создание специализированной сервис-ориентированной информационной среды в Интернет/Интранет, ориентированной на информационно-технологическую и организационную поддержку проектной деятельности таких виртуальных команд. Автором разработана и реализована функциональная модель такой сервис-ориентированной информационной среды в виде программно-информационного комплекса (далее именуемой виртуальной технологической площадкой – ВТП) [1, 2].

При разработке и анализе проектных решений ВТП с учетом перспективы оценки вариантов её масштабирования для спецификации отдельных сервисов и всей функциональной модели ВТП обычно используется аппарат высокоуровневых сетей Петри, в частности временных раскрашенных сетей Петри [3, 4, 6].

Временной раскрашенной сетью Петри [6] называется набор $TCPN = (CPN, Time)$, где $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ – раскрашенная сеть Петри, а $Time: T \rightarrow Interv(N_+)$ – функция временных интервалов. Здесь $Interv(N_+) = \{[t_1, t_2], [t_3, \infty) \mid t_1, t_2, t_3 \in N_+, t_1 < t_2\}$.

Границы временного интервала трактуются как раннее и позднее время срабатывания перехода раскрашенной сети Петри. Для структурирования модели, наглядности ее визуального представления и анализа используется функция иерархии в раскрашенных сетях Петри, которая позволяет некоторый переход (называемый переходом-заменителем) в сети заменять подсетью. Она позволяет более полно детализировать данный переход, не усложняя при этом саму сеть. Отдельные неиерархические подсети называются страницами. Формальное определение иерархической раскрашенной сети Петри здесь не приводится, однако отметим, что в [5] доказано: для каждой иерархической сети Петри существует соответствующая ей неиерархическая сеть, которая имеет тот же набор маркеров, шагов и последовательностей событий, т.е. обе сети имеют эквивалентное поведение.

Далее в тексте используется терминология аппарата сетей Петри, принятая в монографии К. Jensen [3, 0].

Построение сетей Петри в графическом представлении (в виде графов) и реализация компьютерного (имитационного) моделирования обеспечивается за счет наличия программного инструментария CPN Tools [7]. Применение данного программного обеспечения предоставляет возможность для получения различных количественных оценок параметров и свойств разрабатываемых аппаратных и программно-технологических решений. Имитационное моделирование на основе аппарата сетей Петри позволило получить различные оценки параметров масштабируемости аппаратного и программно-информационного окружения ВТП.

Функциональная модель предложенной виртуальной технологической площадки в предложенной версии регламентирует и информационно поддерживает процессы поэтапного итерационного выполнения набора взаимосвязанных и контролируемых задач проекта разработки ПО. Она предусматривает сервисы по поддержке управления организационно-информационной деятельностью в проекте, хранения электронных документов и файлов программного кода, а также управления их версионностью. В рамках предложенной функциональной модели инструментальная поддержка разработки программного кода и последующего его тестирования осуществляется членами виртуальной проектной команды на их локальных рабочих станциях. Итоговой целью завершения проектной деятельности является получение конечного программного продукта из категории нематериальных активов [1]: программный продукт в виде версии инсталляционного пакета, содержащего дистрибутив файлов программных модулей с комплексом проектной и эксплуатационно-технической документации.

На содержательном уровне общая схема функциональной модели ВТП представлена на рис. 1.



Рис. 1. Функциональная модель ВТП

Остановимся кратко на содержательной спецификации базовых функциональных сервисов предложенной функциональной модели, регламентирующих политику виртуальных команд, направленную на производство программных продуктов:

1. Сервис «Поддержка управления организационно-информационной деятельностью в проекте» основывается на использовании термина «задача», означающего технологическую операцию/работу в ходе выполнения программного проекта в ВТП. Сервис реализует поддержку следующих основных функций:

- комплектование списка задач, содержащих информацию о сроках их выполнения с указанием роли ответственных за разрешение членов проектной команды (каждая задача может быть одного из двух типов: «документ» или codefix, т.е. ошибка программного кода);
- отслеживание процесса разрешения зарегистрированных задач и выполнения пожеланий пользователей;
- формирование информации о ходе выполнения как всего проекта в целом, так и его стадий в частности.

2. Сервисы «Управление версионностью электронных документов» / «Управление версионностью исходного кода» реализуют в составе ВТП функции создания и сопровождения баз дан-

ных объектов проектной деятельности двух типов соответственно: «документ» и «файл программного кода». Функциональные возможности каждого из сервисов позволяют хранить несколько версий текстового файла соответствующего типа, возвращаться к более ранним версиям, производить их автоматическое слияние и построчное сравнение и т.д.

Рассмотрим спецификацию предложенной выше модели функционирования ВТП с учетом отображения механизмов функционирования ее базовых сервисов в виде иерархических раскрашенных временных сетей Петри. Предложенный вариант реализации в среде CPN Tools представлен на рис. 2.

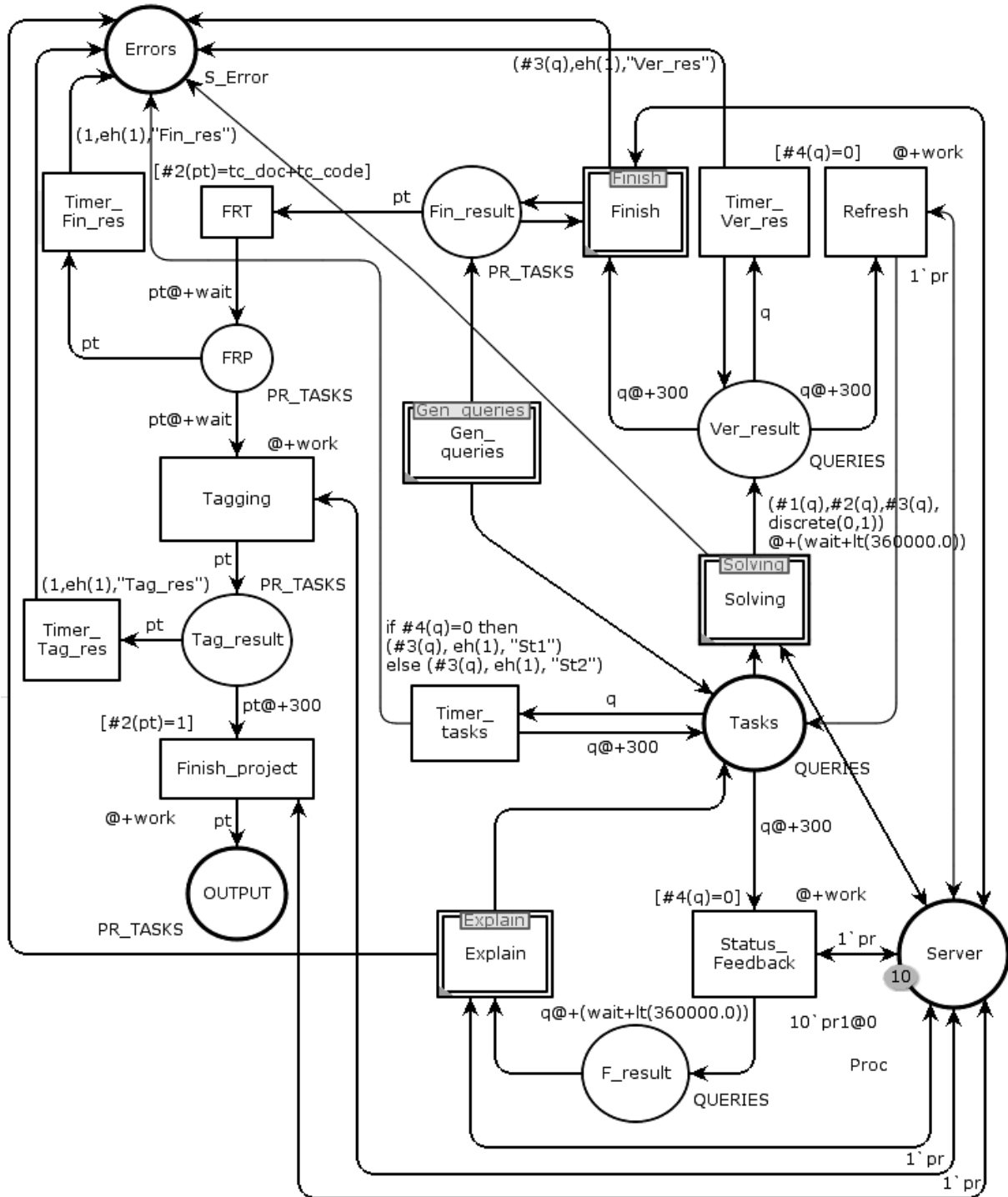


Рис. 2. Обобщенная модель ВТП в виде сети Петри

Простые переходы сети реализуют функции сервиса ВТП «Поддержка управления организационно-информационной деятельностью в проекте». В зависимости от типа задачи программного проекта следующие подсети реализуют функционирование остальных сервисов ВТП:

- подсети Solving и Finish являются спецификацией сервиса «Управление версионностью электронных документов» для задач типа «документ»;
- подсеть Solving является спецификацией сервиса «Управление версионностью исходного кода» для задач типа codefix.

Автором разработан полный набор спецификаций функционирования сервисов ВТП в виде подсетей, который в силу ограничений объема не приводится в статье.

Приведем описание построенной сети (цвета, позиции, переходы и т.д.).

Допустимые цвета фишек определены следующим образом:

QUERIES = (Pid, Tid, Ttype, Tlogic), где:

- Pid – натуральное число (идентификационный номер проекта);
- Tid – натуральное число (идентификационный номер задачи);
- Ttype = 1 или 2 (тип задачи: 1 – «документ», 2 – codefix);
- Tlogic – логического типа с набором значений {0, 1}.

PR_TASKS = (Pid, Tcount), где:

- Pid – натуральное число (идентификационный номер проекта);
- Tcount – целое неотрицательное число (количество завершенных задач для выпуска релиза проекта).

S_Error = (Tid, Error), где:

- Tid – натуральное число (идентификационный номер задачи);
- Error – строкового типа (для обозначения ошибки обработки запроса сервером).

Proc = pr1 | pr2, где:

- pr1, pr2 – строкового типа (идентификатор сервера).

Позиции построенной сети Петри. Tasks – позиция, фишки которой характеризуются цветом (Pid, Tid, Ttype, Tlogic). Значение логического флага Tlogic определяет либо необходимость дальнейшего уточнения (составной переход **Explain**) созданной задачи, либо возможность перехода к стадии ее разрешения ответственным участником проектного коллектива, т.е. в случае, когда Tlogic = 0, фишка отождествляется странице в браузере с формой редактирования задачи с измененным статусом «Обратная связь» на локальной рабочей станции ответственного за ее разрешение. Наличие фишки с Tlogic = 1 соответствует готовности участника проектного коллектива ее разрешить (возбужденность составного перехода **Solving**).

F_result – наличие фишки в этой позиции свидетельствует об успешном изменении статуса задачи на значение «Обратная связь». Цвет фишек позиции определяется как (Pid, Tid, Ttype, Tlogic). Наличие фишки определяет начало процесса уточнения задачи ее «создателем» (возбужденность составного перехода **Explain**).

Ver_result – позиция, каждая фишка которой определяет готовность задачи для перевода в статус «Завершена». Цвет фишек позиции определяется как (Pid, Tid, Ttype, Tlogic). Значение флага Tlogic задает тот факт, что задача может быть завершена или ее следует доработать (1 или 0 соответственно). Если Tlogic = 0, то фишка соответствует странице в браузере с формой редактирования задачи, которая включает комментарий и измененный статус «Открыта заново», на локальной рабочей станции «создателя» задачи, т.е. запрос на сохранение этой страницы в системе находится в очереди для обработки сервером (возбужденность составного перехода **Refresh**).

Fin_result – позиция, фишки которой определяют возможность выпуска релиза проекта. Фишки характеризуются цветом (Pid, Tcount). Если значение поля Tcount, соответствующее количеству завершенных задач для проекта с номером Pid, равно заданной константе, тогда возможен запуск переходов **FRT** и **Tagging**.

FRP – вспомогательная позиция, которая, в силу специфики программной инструментальной среды CPN Tools, используется только для хранения фишки из предыдущей позиции **Fin_result** с добавленным значением временной задержкой для корректной обработки запроса сервером (переход **Tagging**).

Tag_result – позиция, наличие фишки в которой свидетельствует о появлении запроса к системе, соответствующего отправке формы на локальной рабочей станции менеджера проекта для его завершения. Цвет фишек позиции определяется как (Pid, Tcount).

Output – каждая фишка этой позиции определяет результат работы ВТП в случае завершения одного проекта. Фишки позиции имеют цвет (Pid, Tcount).

Server – позиция моделирует разделяемые ресурсы, в данном случае доступные процессы аппаратных ресурсов (серверов). При начальной маркировке в этой позиции фишки с нулевой временной задержкой определяются цветом $Proc = pr1 \mid pr2$. Общее количество фишек с фиксированным идентификатором сервера ($pr1$ или $pr2$) соответствует количеству процессов, доступных к выполнению на соответствующем сервере. Наличие фишек в позиции свидетельствует о возможности процесса на указанном сервере обработать очередной пользовательский запрос в системе: отправка формы создания, редактирования, разрешения, завершения задачи; извлечение, обновление рабочей копии проекта и т.д. Длительность обработки каждого запроса определяется значением временной задержки на переходе, представляющим операцию в системе.

Переходы и правила их срабатывания в построенной сети Петри

1. Переходы сети, в результате последовательного выполнения которых для каждой фиксированной фишки происходит ее перемещение из позиции **Tasks** в **Fin_result**, соответствующее завершению жизненного цикла задачи с идентификатором Tid , созданной для проекта с заданным номером Pid , интерпретируются следующим образом:

- **Status_Feedback** – переход анализирует фишки из позиции **Tasks**, и если значение флага $Tlogic = 0$, то переход выполняет обработку формы редактирования задачи с измененным статусом «Обратная связь». Запуск перехода перемещает фишку из позиции **Tasks** в **F_result**.

- **Explain** – составной переход, моделирующий процесс уточнения задачи со статусом «Обратная связь» пользователем, ее создавшим. В результате срабатывания фишка с цветом (Pid , Tid , $Ttype$, $Tlogic = 0$) перемещается из позиции **F_result** в **Tasks**. При этом генерируется новое значение флага $Tlogic = 1$, которое определяет возбужденность перехода **Solving** (задача больше не требует уточнения).

- **Solving** – составной переход, моделирующий сценарий разрешения задачи ответственным участником проектного коллектива. В результате срабатывания перемещает фишку из позиции **Tasks** в **S_result**.

- **Refresh** – переход, отвечающий за обработку формы редактирования задачи, включающей комментарий и измененный статус «Открыта заново». Его запуск доступен в случае $Tlogic = 0$ для фишки в **Ver_result** (задача требует доработки) и создает фишку в позиции **Tasks** с новым значением поля $Tlogic$, равным 0 или 1 с вероятностью 0,5 (задача требует уточнения или готова к разрешению соответственно).

- **Finish** – составной переход, моделирующий сценарий завершения разрешенной задачи. Доступен для запуска, только если $Tlogic = 1$ фишки в **Ver_result**. При каждом срабатывании переход удаляет фишку из **Ver_result** и инкрементно на 1 увеличивает значение $Tcount$ для фишки (Pid , $Tcount$) в позиции **Fin_result**. Таким образом, моделируется итерационное выполнение набора жизненных циклов всех задач для выпуска релиза проекта с номером Pid .

2. Переходы, последовательное выполнение которых для каждой фиксированной фишки соответствует завершению проекта с идентификатором Pid , интерпретируются следующим образом:

- **FRT** – вспомогательный переход, в силу специфики интерпретации времени в CPN Tools используемый только для задания временной задержки фишки из предыдущей позиции **Fin_result**. Переход является возбужденным только в том случае, если проект с номером Pid для фишки (Pid , $Tcount$) позиции **Fin_result** имеет достаточное количество завершенных задач $Tcount$. Его запуск перемещает фишку из **Fin_result** в позицию **FRT**. Вместе с позицией **FRP** переход **FRT** обеспечивает корректную обработку запроса сервером (переход **Tagging**) и срабатывание перехода-таймера **Timer_Fin_res** по истечении времени ожидания.

- **Tagging** – переход, моделирующий выпуск релиза проекта. При его срабатывании происходит перемещение фишки со значением (Pid , $Tcount$) из позиции в **FRP** в позицию **Tag_result**.

- **Finish_project** – срабатывание перехода перемещает фишку из позиции **Tag_result** в **Output**. Соответствует выполнению запроса по отправке в браузере формы завершения проекта на локальной рабочей станции менеджера проекта.

3. Переходы-таймеры (**Timer_create**, **Timer_tasks**, **Timer_Ver_res**, **Timer_Fin_res** и **Timer_Tag_res**) введены в модель сети Петри в целях реализации механизма отказа сервера (удаления фишек из позиций). Эти переходы срабатывают в том случае, если запросы к сервисам ВТП (отправка различных форм через клиент-браузер или операций с версионным хранилищем), моделируемые фишками во входных позициях, не обрабатываются сервером системы (позиция **Server**) в течение продолжительного времени. Функция выражений для выходных дуг по отношению к пере-

ходам-таймерам равна ($Tid, Error$), где $Error$ имеет соответствующее запросу значение отрицательного ответа. Все переходы-таймеры имеют временную задержку до запуска, равную изначально заданному времени ожидания для обработки запроса сервером, а системные переходы игнорируют эту задержку, т.е. являются возбужденными в момент наступления соответствующего значения модельного времени. Такое поведение сети достигается выражением функции времени $@+(\text{wait}+\text{work})$ на входной дуге любой позиции, связанной с переходом-таймером, и выражением $@+\text{wait}$ на ее выходной к системному переходу дуге (игнорирование значения временного штампа, равного wait). Таким образом, если фишка в какой-либо позиции не успевает быть обработанной системным переходом, пока модельное время равно любому значению из интервала $[\text{timestamp_фишки}-\text{wait}; \text{timestamp_фишки}]$, тогда она удаляется из нее срабатыванием перехода-таймера. В этом случае пользователь получает информационное сообщение об отказе обработки запроса сервером в связи с окончанием (истечением) времени ожидания, равного wait . При этом переход-таймер вернет фишку в позицию с новым временем ожидания для обработки сервером (выражение $q@+\text{wait}$ на его выходной дуге), т.е. попытка отправить запрос на исполнение системным переходом будет возникать до тех пор, пока не произойдет его запуск.

Генерация начального состояния для обобщенной сети Петри осуществляется переходом **Gen_queries**, выполнение которого создает поток запросов к системе для обработки сервером (позиция **Server**). Этот переход является подсетью для сети обобщенной модели ВТП, и соответствующая ему подсеть представлена на рис. 3.

В качестве начальных данных для подсети **Gen_queries** при реализации в CPN Tools задаются значения трех параметров: prc – количество проектов; tc_doc – количество задач типа «документ»; tc_code – количество задач типа codefix. Следовательно, общее количество задач в системе равно $\text{prc}*(\text{tc_doc}+\text{tc_code})$. Начальная маркировка сети содержит:

- 1^1 в позиции **Pr_count**, 1^0 в **Tasks_Ids**.
 - $1^0(0,0)+1^0(0,1)+1^0(0,2)+1^0(0,3)$ в позиции **Intensity**. Первая компонента цвета каждой фишки соответствует количеству запросов в конкретный час завершения проекта в ВТП, определяемый значением второй компоненты (нулевой, первый и ... час).
 - Пустые мультимножества в остальных позициях.
- Сначала генерируется количество задач обоих типов для каждого проекта с номером $\text{pid} \in [1, \text{prc}]$.

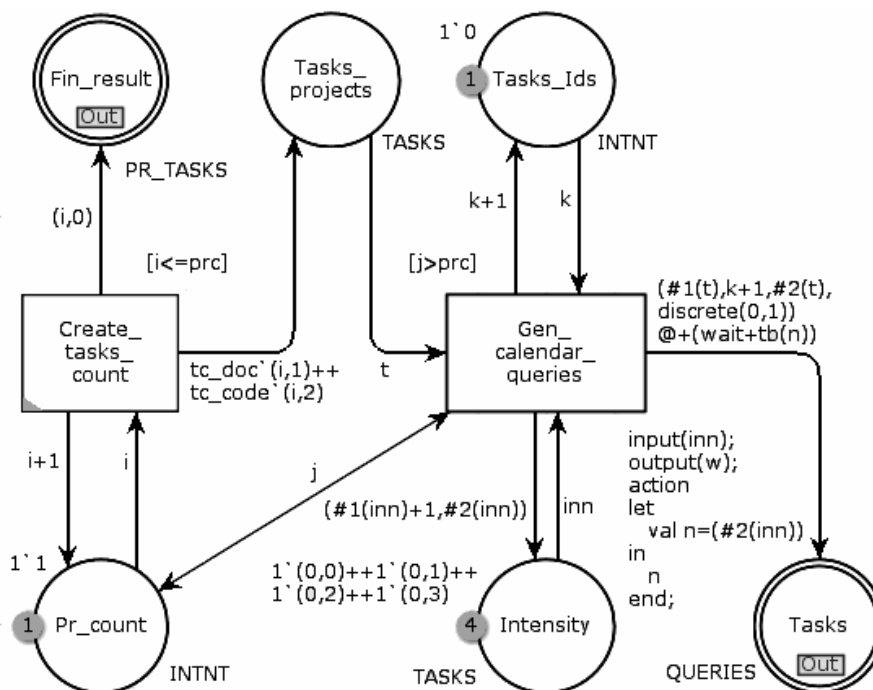


Рис. 3. Сеть Петри для генерации начального состояния модели

При каждом запуске перехода **Create_tasks_count**:

- Функция дуги в позицию **Fin_result** создает в этой позиции фишку $(\text{Pid}, 0)$ (ноль завершенных задач проекта с фиксированным номером Pid).

• Функция дуги в позиции **Pr_count** инкрементно на 1 увеличивает *Pid* проекта пока $i < prc$ (выражение $i+1$). При этом в **Tasks_projects** создается *tc_doc* фишек цвета (*Pid*,1) и *tc_code* фишек (*Pid*,2).

Функция охраны перехода **Create_tasks_count** ограничивает поток запросов по количеству проектов (*prc*), а значения *tc_doc* и *tc_code* задают количество задач типа «документ» и *codefix* соответственно. Это дает возможность проводить численные эксперименты на ограниченном количестве запросов.

Каждая фишка из **Tasks_projects** трансформируется в запрос пользователя на разрешение созданной в ВТП задачи и характеризуется номером проекта, номером задачи, ее типом и логическим флагом, определяющим дальнейший сценарий выполнения запроса. При этом значение временной задержки, определяемое значением выражения $@+(wait+tb(n))$ на дуге перехода **Gen_calendar_queries** в позицию **Tasks**, позволяет регулировать интенсивность поступления запросов к аппаратным ресурсам системы. Функция *tb(n)* принимает значения в соответствии с нормальным законом распределения, для которого математическое ожидание равно *n*-му часу поступления запроса, а дисперсия исчисляется в минутах. При этом значение аргумента *n* определяется значением второй компоненты случайно выбранной фишки из позиции **Intensity**. Возвращение фишки в позицию **Intensity** происходит с инкрементным увеличением значения первой компоненты. После $prc*(tc_doc+tc_code)$ запусков перехода **Gen_calendar_queries** это значение определяет общее количество запросов в конкретный час функционирования ВТП (значение второй компоненты).

В результате работы сети **Gen_queries** моделируется поток пользовательских запросов, представляющий начальное состояние сети Петри функциональной модели ВТП, т.е. в позиции **Tasks** генерируется $prc*(tc_doc+tc_code)$ запросов с равномерным распределением временной задержки по количеству фишек в **Intensity**, причем каждое значение полученной случайной величины имеет нормальное распределение с математическим ожиданием, равным *n*-му часу поступления запроса в ВТП.

Имитационное моделирование обобщенной модели ВТП. Процесс имитационного моделирования на основе сетей Петри обеспечивается за счет наличия программного инструментария CPN Tools, применение которого дает возможность не только для описания статической топологии предложенной спецификации, но и для получения различных количественных оценок параметров и свойств разрабатываемых аппаратных и программно-технологических решений.

Входными параметрами для проведения численных экспериментов на построенной сети Петри, как это уже частично обозначено выше при описании процесса формирования начального состояния, являются:

- *prc* – количество проектов в ВТП;
- *tc_doc* – количество задач типа «документ» для любого проекта с номером *pid* $\in [1,prc]$;
- *tc_code* – количество задач типа *codefix* с номером *pid* $\in [1,prc]$;
- *work* – среднее время обработки любого системного перехода, соответствующего запросу пользователя к функциям сервисов ВТП;
- *wait* – время ожидания обработки запроса сервером;
- *k* и *n* – задают количество одновременно доступных процессов для каждого из серверов с номерами *pr1* и *pr2* соответственно;
- *hours* – количество часов поступления запросов в ВТП.

В качестве демонстрационного примера оценки масштабируемости программно-информационного комплекса ВТП при реализации (на базе) предложенной функциональной модели приведем результаты проведения численных экспериментов на компьютерной модели сети Петри с различными наборами входных параметров. В предлагаемых численных экспериментах анализируется вариант реализации программно-информационного комплекса ВТП, установленного на одном сервере (фишки цвета *pr1* в позиции **Server**) с различными показателями производительности, которая определяется средним значением времени обработки каждого запроса в ВТП (параметр *work* в модели). Объемом памяти, выделяемой на аппаратном ресурсе при функционировании ВТП, задается максимальное количество одновременно доступных процессов на Web-сервере для обработки пользовательских запросов в системе (количество *k* фишек в позиции **Server**). Начальная интенсивность поступления запросов к проблемно-ориентированным сервисам ВТП задается следующим алгоритмом (срабатывание составного перехода **Gen_queries**): все $prc*(tc_doc+tc_code)$ запросов имеют равномерное распределение временной задержки по количеству *hours* (количество фишек в позиции

Intensity). Предполагается, что каждое значение полученной случайной величины имеет нормальное распределение с математическим ожиданием μ , равным i -му часу поступления запроса в ВТП ($i \in [0, \text{hours}-1]$) и дисперсией σ^2 , равной 400 с (т.е. величина отклонения запросов по времени равна минуте -3σ). Количество ошибок обработки запросов серверами будет увеличиваться до тех пор, пока не будут решены все (tc_doc+tc_code) задач для каждого проекта.

В качестве показателей качества модельных вариантов решений рассматриваются количество успешных ответов сервера и уровень ошибок при заданной интенсивности поступления запросов. Результаты проведения численных экспериментов с различным набором входных данных представлены в таблице. Выбор количественных параметров имитационного моделирования осуществляется на основе анализа журнальных файлов разработанных серверов и самых распространенных информационных ресурсов в сети Интернет: среднее значение времени обработки запроса $wait$, количество одновременно доступных процессов на сервере k . Также время ожидания ответа обработки пользовательского запроса к системе (параметр $wait$ в модели) не должно превышать 30 с.

Данные проведения численных экспериментов в среде CPN Tools

prc	tc_doc	tc_code	work, c	wait, c	k	n	hours	Ошибки
10	100	100	0,8–1,0	30	10	0	4	850–1127
10	100	100	0,8–1,0	30	15	0	4	9–34
10	100	100	0,5–0,8	30	10	0	4	42–101
10	100	100	0,5–0,8	30	15	0	4	0
10	150	50	0,8–1,0	30	10	0	4	293–503
10	50	150	0,8–1,0	30	10	0	4	1312–1823
10	500	200	0,8–1,0	30	10	0	4	63009–63693
10	500	200	0,8–1,0	30	25	0	4	9659–10290
10	500	200	0,8–1,0	30	40	0	4	466–708
10	500	200	0,5–0,8	30	10	0	4	35494–38355
10	500	200	0,5–0,8	30	25	0	4	2304–2797
10	500	200	0,5–0,8	30	40	0	4	0

Из данных, приведенных в таблице, можно сделать вывод, что количество процессов, одновременно доступных на сервере (параметр k), является определяющим параметром для минимизации количества ошибок обработки запросов. Уменьшение времени обработки запроса (параметр $work$) не приводит к подобному значительному сокращению потока аппаратных ошибок.

Таким образом, для компьютерной модели сети Петри был проведен ряд имитационных экспериментов с целью анализа и оценки вариантов масштабируемости нагрузки, состава серверов и структуры программно-технического комплекса в зависимости от характеристик аппаратных ресурсов (время обработки запросов и количество процессов на серверах) и количественных параметров модели (количество проектов и задач). Результаты численных экспериментов на модели позволяют рекомендовать ее в качестве рабочего инструмента решения задачи масштабируемости ВТП и адаптации ее архитектуры к числу поддерживаемых проектов и размеров совокупности команд исполнителей виртуального предприятия.

Литература

1. Бухтияров И.В. Сервис-ориентированная среда для организации виртуального предприятия по производству программных продуктов / И.В. Бухтияров, Ю.М. Зыбарев // Программная инженерия. – 2014. – № 10. – С. 11–18.
2. Кратов С.В. Технологическая площадка разработки ПО в СО РАН / С.В. Кратов, И.В. Бухтияров // Матер. VII Азиатской междунар. школы-семинара «Проблемы оптимизации сложных систем»: Труды ИВМ и МГ СО РАН. Сер. Информатика. – 2011. – Вып. 10. – С. 68–73.
3. Peterson J.L. Petri Net Theory and the Modelling of Systems. – Englewood Cliffs: Prentice-Hall, 1981. – 291 p.
4. Цветные сети Петри в моделировании социально-экономических систем / Ю.П. Ехлаков, В.Ф. Тарасенко, О.И. Жуковский, П.В. Сенченко, Ю.Б. Гриценко // Доклады ТУСУРа. – 2013. – № 3 (29). – С. 83–92.
5. Jensen K. Coloured Petri Nets: A High Level Language for System Design and Analysis // Advances in Petri nets 1990. – Springer Berlin Heidelberg, 1991. – P. 342–416.

6. Jensen K. Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use [Электронный ресурс]. – Режим доступа: http://books.google.ru/books/about/Coloured_Petri_nets.html?id=6ck3wqmIchYC, свободный (дата обращения: 27.10.2014).

7. CPN Tools Homepage – CPN Tools is a tool for editing, simulating, and analyzing Colored Petri nets [Электронный ресурс]. – Режим доступа: <http://cpntools.org/>, свободный (дата обращения: 27.10.2014).

Бухтияров Иван Валерьевич

Мл. науч. сотрудник Института вычислительной математики и математической геофизики СО РАН,

г. Новосибирск

Тел.: 8-913-710-49-61

Эл. почта: bukhtiyarov_i@mail.ru

Bukhtiyarov I.V.

Petri nets specification of service-oriented environment for the software production

The article is devoted to the problems of specification of a functional model of service-oriented informational environment in the Internet / Intranet in terms of Petri nets. This environment is created for the organization of software production by virtual teams of experts. Using the functional capabilities of the environment services is based on the project approach. The purpose of implementation of such an approach is to develop software products. The main object of the project activity is operational task. Project can be completed only after iterative executing of a set of lifecycles for all tasks. Petri nets are considered not only as a tool of specification, but also as a tool for simulation which is implemented by applying software called CPN Tools. Conducting numerical experiments based on using this software enables to get a variety of quantitative estimation of parameters and properties of the developed hardware and software technological solutions.

Keywords: software production, project activity, functional model, Petri nets, simulation.
