

УДК 004.4

МОДЕЛЬ МУЛЬТИВЕРСИОННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ИЗМЕНЕНИЯМИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ИНФОРМАЦИОННЫХ ПРОЕКТАХ

Л. Р. Зарипова, К. И. Бушмелева

Сургутский государственный университет, bkiya@yandex.ru, zlr23.07@gmail.com

В работе представлена разработка модели мультиверсионной информационной системы управления изменениями программного обеспечения в информационных проектах (IT-проекты) с учетом рабочего времени посредством языка UML и сетей Петри. Данная модель взаимодействия пользователей и информационной системы управления изменениями IT-проекта была реализована с использованием UML-диаграмм. Также представлена интерпретация UML диаграмм в сети Петри, сделан анализ сетей Петри с помощью дерева достижимости.

Ключевые слова: мультиверсионная информационная система, информационные проекты, программное обеспечение, язык UML, сети Петри, UML диаграммы, дерево достижимости.

MULTIVERSION INFORMATION SYSTEM MODEL FOR MANAGING SOFTWARE CHANGES IN INFORMATION PROJECTS

L. R. Zaripova, K. I. Bushmeleva

Surgut State University, bkiya@yandex.ru, zlr23.07@gmail.com

The article considers the development of multiversion information system for managing software changes in information projects (IT-projects) and management of working time using the UML and Petri nets. This model of interaction between users and information system for managing software changes of an IT-project is implemented using UML diagrams. The interpretation of UML diagrams in Petri network is shown. Petri networks analysis using the reachability tree is made.

Keywords: multiversion information system, information projects, software, UML language, Petri nets, UML diagrams, reachability tree.

В настоящее время на рынке IT-технологий представлен огромный выбор информационных продуктов для быстрой и эффективной работы различных предприятий, которые необходимы для управленческой, экономической и технической деятельности.

IT-компании, используя в своей работе множество программных продуктов (ПП), разрабатывают различные подходы к интеграции нескольких программ для получения аналитической, управленческой отчетности по предприятию. В результате возникает проблема в управлении действиями, которые выполняются на этапах проектирования, разработки, тестирования и внедрения. В постоянных доработках программ теряется последовательность вносимых изменений, т. е. становится неясно, на основе какого технического задания произошли определенные изменения в ПП.

Целью данного исследования является разработка программного обеспечения (ПО) мультиверсионной информационной системы управления изменениями программного обеспечения в информационных проектах с использованием языка UML и сетей Петри.

Для достижения поставленной цели решаются следующие задачи:

1. Проектирование программного обеспечения с использованием UML.
2. Анализ поведенческих UML-диаграмм с помощью сети Петри.

В работе получены следующие результаты, отличающиеся научной новизной: модель организации взаимодействия пользователей и мультиверсионной системы с использованием

языка UML и модель маркированной сети Петри для оценки вероятностных характеристик системы, которые позволят обеспечить выбор рационального варианта построения системы.

Создание информационной системы на основе совместного использования UML-диаграмм и сетей Петри проходило в несколько этапов:

- выбор и разработка требуемых UML-диаграмм;
- трансляция полученных UML-диаграмм в сети Петри;
- анализ дерева достижимости в сети Петри;
- корректировка UML диаграмм.

Для выбора и разработки UML диаграмм можно обратиться к работам [1–8], изучение которых позволило провести анализ диаграмм последовательности и активности с использованием сетей Петри [9] и выявить определенные правила преобразования, которые будем использовать в данной работе.

В состав организационной структуры реализуемого IT-проекта входят следующие группы: проектирование, разработка, тестирование и сопровождение. Рассмотрим сценарий работы мультиверсионной информационной системы в организации взаимодействий пользователей (рис. 1). На диаграмме последовательности показаны объекты и классы, используемые в сценарии, и последовательность сообщений, которыми обмениваются объекты для выполнения сценария. Сценарий – это единичный проход по потоку событий для прецедентов.

Далее необходимо наглядно отобразить методы классов с использованием диаграмм действий.

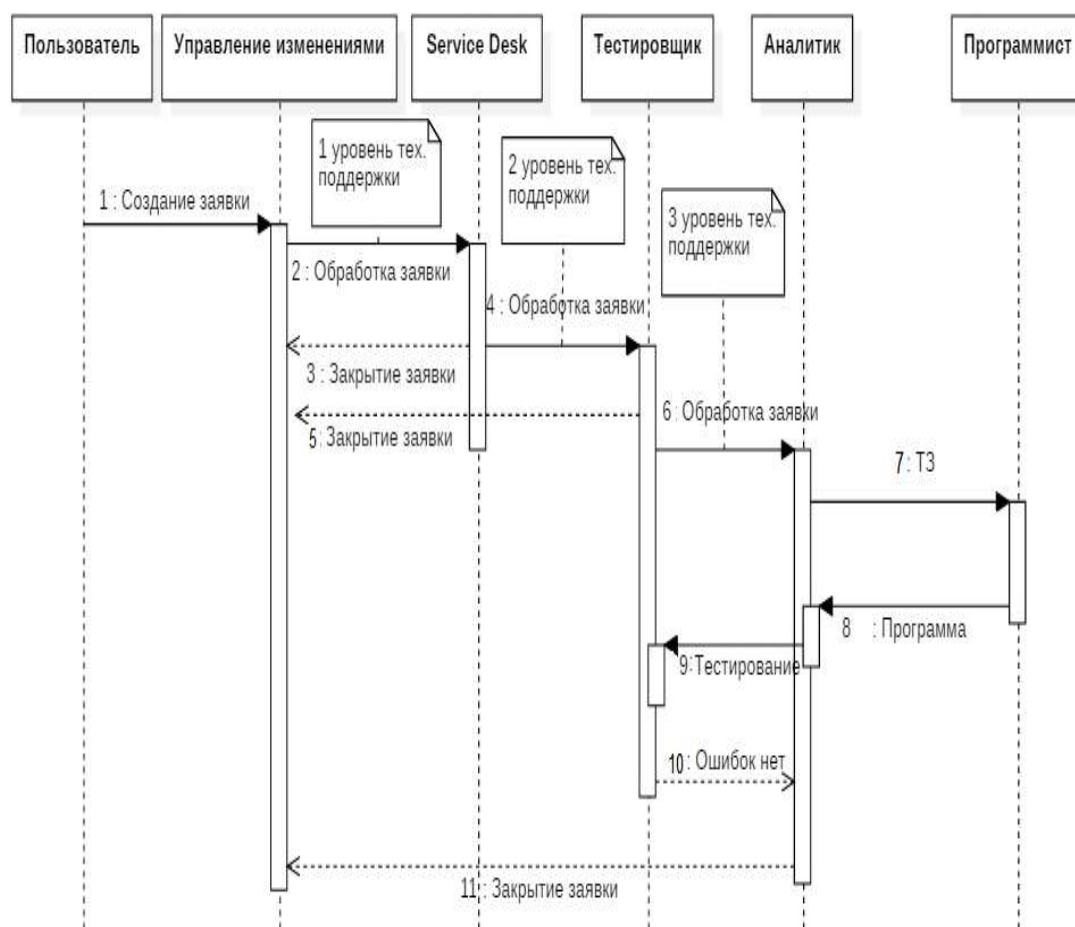


Рис. 1. Диаграмма последовательности

Рассмотрим диаграмму действий пользователей в реализуемой мультиверсионной информационной системе (рис. 2).

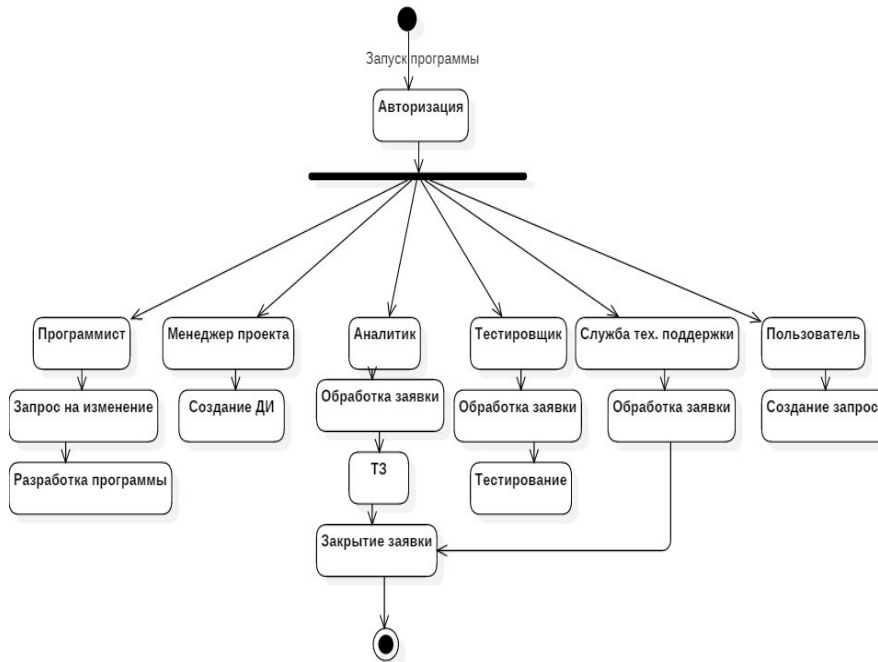


Рис. 2. Диаграмма действий пользователей в мультиверсионной информационной системе

Использование диаграммы действий применяется для линейного исполнения ПО и описания алгоритмической части информационной системы.

При разработке модели посредством сети Петри состояние действий диаграммы UML преобразуется в переход, отображающий начало действия, выполнение действия и завершающее действие.

В результате реализуемые модели организации взаимодействия пользователей и мультиверсионной системы с использованием языка UML и маркированной сети Петри для оценки вероятностных характеристик системы позволяют обеспечить выбор рационального варианта построения информационной системы.

На рис. 3 представлена маркированная модель сети Петри, позволяющая отобразить организацию взаимодействия пользователей и мультиверсионной системы.

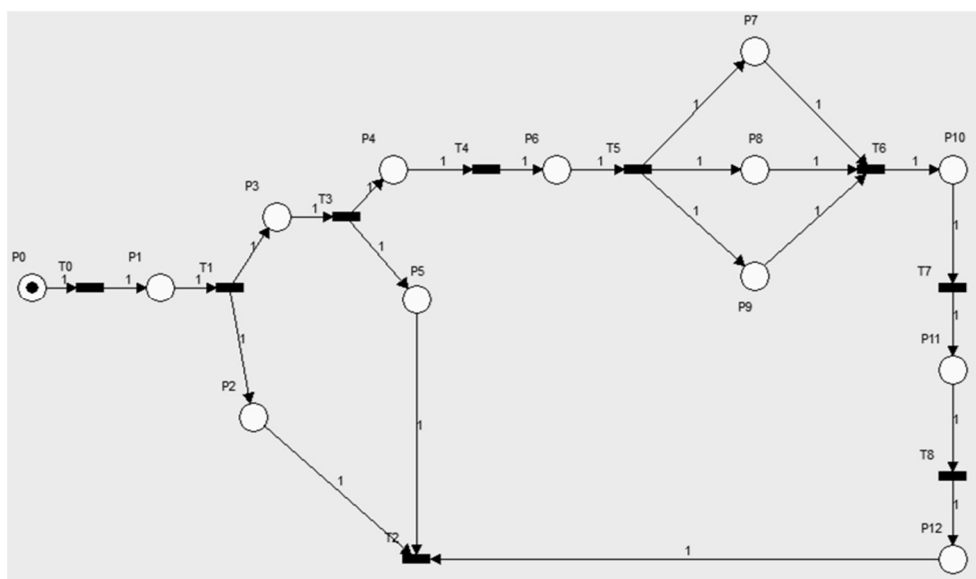


Рис. 3. Модель организации взаимодействия пользователей и мультиверсионной системы с использованием маркированной сети Петри

Анализ модели позволяет сделать вывод, что в начальный момент времени пользователь (заказчик) создает заявку, например, по ошибке в исходных данных в системе. Это показано наличием фишки в позиции P_0 и отсутствием других фишек в модели (табл. 1).

Таблица 1

**Интерпретация соответствия «позиция – событие – переход»
в мультиверсионной системе**

Позиция	Событие	Переход
P_0	Создание заявки	T_0
P_1, P_3	Обработка заявки	T_1, T_3
P_2, P_5, P_{12}	Закрытие заявки	T_2, T_9
P_4	Создание, редактирование ДИ (документа изменений)	T_4
P_6	Создание ТЗ	T_5
P_7, P_8, P_9	Доработка/разработка программы	T_6
P_{10}	Тестирование программы	T_7
P_{11}	Внедрение программы	T_8

При создании заявки срабатывает переход T_0 , и в позиции P_1 появляется новая фишка. Затем осуществляется переход T_1 , означающий обработку заявки. Далее фишка T_2 переходит в событие P_{12} , что приводит к закрытию заявки первым уровнем технической поддержки (в случае успешной обработки заявки). Если заявку не удалось решить, то происходит переход на 2 уровень технической поддержки, который обозначается переходом T_3 . В случае предложенного решения заявка переходит в событие P_5 , что соответствует закрытию заявки в системе. В событие P_4 создается или корректируется документ изменения проекта, к которому привязаны технические задания P_6 . В следующий момент времени осуществляется переход T_6 , в зависимости от требований ТЗ и как следствие программа разрабатывается или дорабатывается. Затем происходит тестирование программы (P_{10}), в случае успешного тестирования срабатывает переход T_8 . Закрытие заявки осуществляется в переходе P_{12} .

Следующей задачей исследования являлся анализ полученных результатов модели сети Петри с помощью построения дерева достижимости, который в свою очередь позволит наглядно показать эффективность работы мультиверсионной системы.

В общем виде дерево достижимости модели мультиверсионной системы, реализованной посредством сети Петри, представлено на рис. 4а, в результате добавления специализированной переменной x (рис. 4 б) происходит упрощение дерева достижимости.

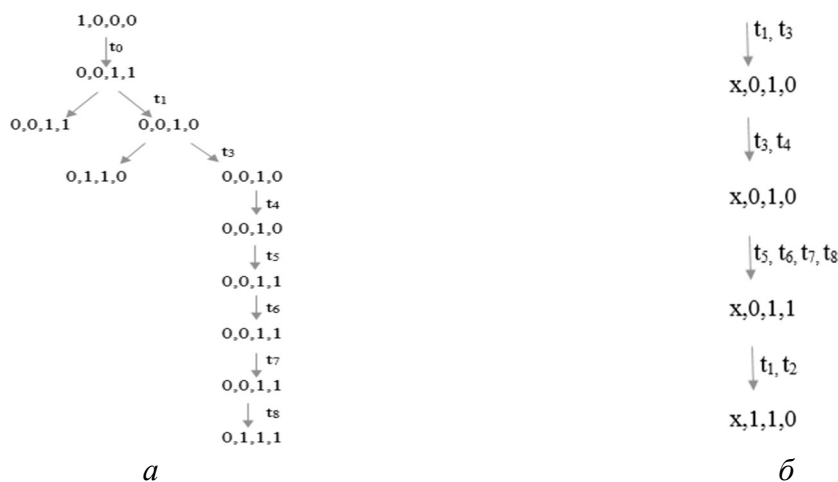


Рис. 4. Дерево достижимости модели

а – дерево достижимости модели на основе сети Петри;
б – представление дерева достижимости с использованием специальной переменной x

Необходимо отметить, что сети Петри также позволяют решать некоторые задачи анализа, такие как: безопасность, ограниченность, сохранение, покрываемость [1–2, 9].

Рассмотрим задачу покрываемости, которую решаем с помощью проверки дерева достижимости, для маркировки интенсивности обработки заявок μ' , в результате определяем маркировку достижимости $\mu'' \geq \mu'$, после чего находим такую вершину u , что выполняется условие $\mu[u] \geq \mu'$. Если бы результаты анализа показали, что такой вершины не существует, то маркировка μ' не являлась бы покрываемой. Однако в представленном дереве достижимости для данной системы такой вершиной u являются вершины t_1 и t_3 . Отсюда следует, что $\mu[u]$ дает достижимую маркировку, которая покрывает μ' . Путь от корня к покрывающей маркировке определяет последовательность переходов, которые приводят из начальной маркировки к покрывающей маркировке, а маркировка, связанная с этой вершиной, определяет покрывающую маркировку.

Заключение. Подводя итог можно отметить, что в результате проведенной работы была разработана модель программного обеспечения с использованием UML-диаграмм, проведен анализ поведенческих UML-диаграмм с помощью сети Петри, сделан анализ сети Петри с помощью дерева достижимости, которая в свою очередь позволяет решить задачу покрываемости мультиверсионной системы.

Литература

1. Пашковская Е. С. Математическое и программное обеспечение мультиверсионной информационной системы с универсальной структурой интерфейсов на основе маркированной сети Петри. URL: <http://www.dissercat.com/content/> (дата обращения: 09.07.2018).
2. Марков А. В. Автоматизация проектирования и анализа программного обеспечения с использованием языка UML и сетей Петри. URL: <http://www.dissercat.com/content/> (дата обращения: 04.07.2018).
3. Атисков А. Ю. Разработка технологии и программной системы автоматизированной трансформации диаграмм функционального проектирования в диаграммах UML. URL: <http://www.dissercat.com/content/> (дата обращения: 02.07.2018).
4. Новиков Ф. А., Иванов Д. Ю. Моделирование на UML. Теория, практика, видеокурс. СПб. : Профессионал. лит. ; Наука и Техника, 2010. 640 с.
5. Основы UML – диаграммы использования (use-case). URL: <https://pro-prof.com/archives/2594> (дата обращения: 13.07.2018).
6. Фаулер М., Скотт К. UML. Основы. СПб. : Символ, 2006. 184 с.
7. Буч Г., Максимчук Р. А., Энгл М. У., Янг Б. Дж., Коналлен Д., Хьюстон К. А. Объектно-ориентированный анализ и проектирование с примерами приложений / 3-е изд. ; пер с англ. М. : И. Д. Вильямс, 2010. 720 с.
8. Розенберг Д., Скотт К. Применение объектного моделирования с использованием UML и анализ прецедентов : пер. с англ. М. : ДМК Пресс, 2002.
9. Мараховский В. Б., Розенблюм Л. Я., Яковлев А. В. Моделирование параллельных процессов. Сети Петри: курс для системных архитекторов, программистов, системных аналитиков, проектировщиков сложных систем управления. СПб. : Профессионал. лит. ; АйТи-Подготовка, 2014. 400 с.