

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ, ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: ТЕОРИЯ И ПРАКТИКА

УДК 519.711.3

ОБ ИСПОЛЬЗОВАНИИ СОБЫТИЙНЫХ ДИАГРАММ ДЛЯ МОДЕЛИРОВАНИЯ ПРОЦЕССОВ

**Бабкин Евгений Александрович, к.т.н., профессор кафедры ПОиАИС,
Ращупкина Валерия Витальевна, ассистент кафедры ПОиАИС,
Курский государственный университет,**

Babkin Evgeny Alexandrovich, PhD, Professor of the POiAIS Department, eababkin@gmail.com
Raschupkina Valeria Vitalievna, Assistant of the POiAIS Department, ArckhipovaLera@yandex.ru
Kursk State University

Рассматривается представление процессов в виде событийных диаграмм на основе унификации нотации событийных графов.

Ключевые слова: *событийный граф, язык UML, системы массового обслуживания.*

В процессе проектирования и анализа информационных систем с использованием объектно-ориентированного подхода используются различные модели для представления систем на разных этапах и уровнях представления систем, используемые для решения разнообразных задач. Для представления бизнес-процессов и информационных процессов широко используются диаграммы языка UML.

UML – унифицированный язык моделирования – это совокупность моделей, которую можно применять для объектно-ориентированного анализа и проектирования [1]. Его можно использовать для визуализации, спецификации, моделирования, проектирования и документирования программных систем. К диаграммам UML относят диаграммы вариантов использования, диаграммы деятельности, диаграммы последовательности, диаграммы состояний и другие. Однако ни одна из них не рассматривает процесс с точки зрения происходящих в нем событий, только в диаграммах состояний события используются как элемент описания переходов между состояниями, то есть как неосновной элемент модели. При использовании имитационного моделирования для анализа проектируемой информационной системы средств диаграмм деятельности и диаграмм состояний недостаточно для полного детализированного отображения процессов обслуживания. В этих диаграммах дисциплины обслуживания могут быть заданы только параметрически, что ограничивает возможности описания особенностей процессов обслуживания, сводит их к стандартным, упрощает и ухудшает точность имитационного моделирования. Более того, созданная модель может получиться даже неадекватной, то есть не соответствовать реальной системе.

Для описания процессов обслуживания удобными инструментами являются структурно-функциональные модели СМО [2] и событийные графы [3 - 5]. Для представления модели системы, когда исходное неформализованное описание имеет событийный характер, используют событийные графы, отражающие процесс функционирования системы в терминах событий и причинно-следственных связей между ними. В изображении событийного графа могут присутствовать три типа вершин и три типа дуг [5]. Чтобы унифицировать подобное представление с диаграммами, входящими в состав UML, необходимо ввести одинаковые обозначения элементов моделей. В связи с этим предлагается ввести следующие обозначения вершин и дуг для построения событийных диаграмм (табл.).

Таблица – Основные элементы диаграммы событий

Графическое изображение	Название и пояснения
	Событие e_i – изменение состояния объекта дискретной системы (ДС). Входов может быть $0..M$, выходов – $0..1$.
	Решение P_i – проверка условия P_i , то есть альтернативный выбор следования событий на основе вычисления логических выражений f_0 и f_1 , причем $f_0 \& f_1 = 0$. В зависимости от результата вычисления f_0 и f_1 выбирается для осуществления только одна причинно-следственная связь, для которой $f_i = 1$. Входов и выходов может быть $1..M$.
	Начало параллельных потоков (вершина синхронизации). После этой вершины начинается одновременное выполнение в процессе сразу нескольких потоков (подпроцессов) по нескольким направлениям следования событий. Выходные дуги могут быть помечены условиями следования событий f_i – тогда реализуется условное распараллеливание процессов. В вершину входит одна дуга, а выходит – не менее двух.
	Окончание параллельного выполнения процесса. В этой вершине вычисляется логическое выражение окончания распараллеливания, в зависимости от истинности которого начинается выполнение основного процесса (происходит переход к следующей вершине событийного графа).
	Мгновенное следование событий. Эта дуга отображает причинно-следственную связь между событиями с задержкой во времени $T = 0$.
	Следование событий с задержкой. Эта дуга отображает причинно-следственную связь между событиями с задержкой во времени $T_{iv} \geq 0$.
	Мгновенная отмена события. Эта дуга отображает причинно-следственную связь отмены события e_v после события e_i без задержки во времени.
	Отмена события с задержкой. Эта дуга отображает причинно-следственную связь отмены события e_v после события e_i с задержкой во времени $T_{iv} \geq 0$.

Рассмотрим представление модели процесса обслуживания с использованием предложенной системы обозначений. На рисунке 1 приведен пример простейшей модели дискретной системы в виде системы массового обслуживания (СМО), состоящей из конечного накопителя H и канала K . Источник I генерирует входной поток заявок на обслуживание N_1 , N_2 – выходной поток обслуженных заявок, N_3 – выходной поток заявок, получивших отказ в обслуживании в связи с конечностью очереди накопителя.

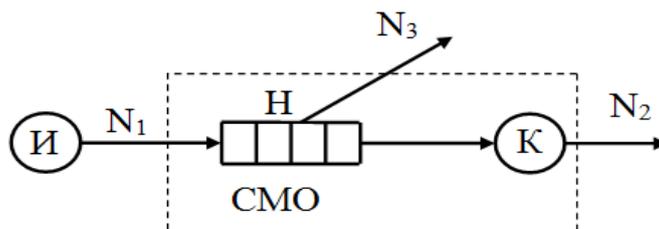


Рисунок 1. – Система массового обслуживания

Рассмотрим имитационную модель процесса обслуживания заявки в СМО. Заявки в систему поступают из источника И. При поступлении заявки в СМО выполняется анализ занятости канала К. Если канал на момент поступления заявки свободен, то заявка поступает в канал, если же занят, заявка должна встать в очередь накопителя Н. Так как очередь ограничена, то заявка может занять её только в том случае, если количество ожидающих заявок в ней не превышает максимального значения, то есть если очередь не полна. В противном случае, заявка покидает систему (поток N_3). После освобождения канала, если очередь не пуста, то в него поступает первая заявка из накопителя Н, в противном случае канал продолжает ожидать поступления заявки в очередь, оставаясь не занятым. Система будет работать до того момента, как не обслужит определенное количество R заявок.

В данной модели присутствуют статические объекты: канал К отождествляется со средством Sr , а накопитель N – с очередью Q . Динамическими объектами являются следующие: обслуживание в канале – активность A , обслуживание заявки в СМО – процесс обслуживания Π . Кроме того, в имитационной модели процесса добавляется главный процесс моделирования Π_m , внутри которого генерируются экземпляры процессов обслуживания Π .

Каждому из этих объектов соответствуют события изменения состояний статических и динамических объектов: e_{3S} – событие занятия средства Sr ; e_{oS} – событие освобождения средства Sr ; e_{3Q} – событие занятия заявкой места в накопителе Q ; e_{oQ} – событие освобождения заявкой места в накопителе Q ; $e_{на}$ – событие начала активности обслуживания A ; $e_{ка}$ – событие окончания активности обслуживания A ; $e_{нп}$ – событие начала процесса обслуживания Π ; $e_{кпN2}$ – событие окончания процесса обслуживания Π (выходной поток $N2$); $e_{кпN3}$ – событие окончания процесса обслуживания Π (выходной поток $N3$); $e_{нпм}$ – событие начала процесса моделирования Π_m ; $e_{кпм}$ – событие начала процесса моделирования Π_m .

Тогда событийная диаграмма, описывающая процессы обслуживания в СМО, выглядит так, как показано на рисунке 2. На данной диаграмме присутствуют следующие вершины: A_1 – одновременное планирование события поступления новой заявки $e_{нп}$ и выполнения процесса обработки поступившей заявки; A_2 – одновременное выполнение процессов завершения обслуживания заявки в системе ($e_{кпN2}$) и выбора для обслуживания следующей заявки из очереди (e_{oQ}), если очередь не пуста; P_1 – анализ состояния средства Sr : свободно ($Cв$) – занято ($Зан$); P_2 – анализ состояния очереди Q : неполна ($Нплн$) – полна ($Пплн$); P_3 – анализ состояния очереди Q : непуста ($Нпст$) – пуста ($Пст$); P_4 – анализ количества заявок обслуженных системой i : ($i < R$) или ($i = R$), где R – заданное число обслуженных заявок.

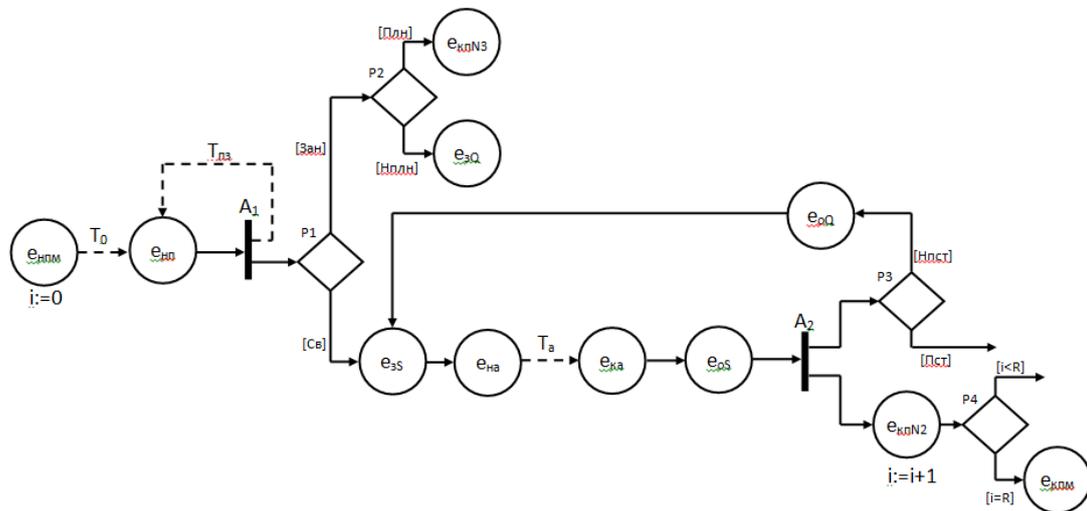


Рисунок 2. – Событийная диаграмма имитационной модели СМО

Также на диаграмме использованы следующие переменные: T_0 – время поступления первой заявки после начала процесса моделирования; $T_{пз}$ – интервал времени поступления заявок в систему; T_a – время активности обслуживания в заявке в средстве.

Событийные диаграммы позволяют представлять процессы обслуживания любой сложности.

Причем их достоинством является удобное представление в явном виде асинхронных взаимодействующих процессов и временных параметров процессов.

Таким образом, в данной работе предлагается нотация событийной диаграммы на основе унификации нотации событийного графа с учётом требований нотации UML для реализации единого подхода к описанию всех аспектов функционирования системы. Это позволит расширить возможности языка UML, так как появится возможность описать порядок и условия выполнения событий при работе модели. Событийные диаграммы позволяют дополнить язык UML средствами описания событийных асинхронных параллельно-выполняемых процессов и процессов обслуживания.

Список использованных источников

1. Буч Г., Рамбо Д., Джекобсон А. Язык UML: руководство пользователя. –2-е изд. стер. – М.: ДМК Пресс; СПб.: Питер,2004. – 432 с.
2. Бабкин Е.А., Бабкина О.М., Володина Н.В., Торубарова Е.А. Учебная система имитационного моделирования сетей массового обслуживания. Вестник Московского городского педагогического университета. Серия «Информатика и информатизация образования». № 6 (16), 2008. – Москва-Курск: Издательство Курского государственного университета, 2008. – С. 15-19.
3. Schruben L.W. Simulation Modeling with Event Graphs // Communications of the ACM. – 1983. – Vol. 26, Num. 11. – P. 957 – 963.
4. Бабкин Е.А. Методические указания по моделированию вычислительных систем на событийно-ориентированном языке. – Курск: КПИ, 1988. – 36 с.
5. Бабкин Е.А. Событийные модели дискретных систем. – Курск. гос. ун-т. Курск, 2005. – 18 с. Деп. в ВИНТИ 14.01.05, № 30-В2005.