

Использование языка UML для проектирования имитационной модели

А. С. ПОМАЗ, В. Д. ЛЕВЧУК

Введение

Структуризация сложной системы является важным и трудоемким этапом моделирования реального объекта. Известны ряд подходов к описанию структуры сложной системы. Наиболее актуальными схемами структуризации являются дискретно-событийный подход, агентный подход и системная динамика. Все данные подходы требуют привлечения специалистов по моделированию [1, 2, 3].

Система моделирования MICIC4 базируется на концепции многоуровневого представления объекта моделирования. Одним из уровней представления модели в системе MICIC4 является уровень аналитика, который разрабатывает концептуальную модель и формальное описание объекта моделирования. Данный этап, как известно, не поддается автоматизации. Аппарат формализации в MICIC4 есть транзактно-процессный способ, являющийся частным случаем дискретно-событийного моделирования. Традиционно структуризация происходит параллельно с ее формализацией. Основная цель данной статьи заключается в выделении этапа структуризации в отдельную стадию разработки имитационной модели с привлечением специалистов-предметников. Для фундамента предлагаемой схемы структуризации используется язык Unified Modeling Language (UML) [4].

UML представляет собой общецелевой язык визуального моделирования, который разработан для спецификации, визуализации, проектирования и документирования компонентов программного обеспечения, бизнес-процессов и других систем. Язык UML одновременно является простым и мощным средством моделирования, который может быть эффективно использован для построения концептуальных, логических и графических моделей сложных систем самого различного целевого назначения. Этот язык вобрал в себя наилучшие качества методов программной инженерии, которые с успехом использовались на протяжении последних лет разработке больших и сложных программных систем.

Рассмотрим предложенную схему структуризации во взаимосвязи с понятиями транзактно-процессного подхода.

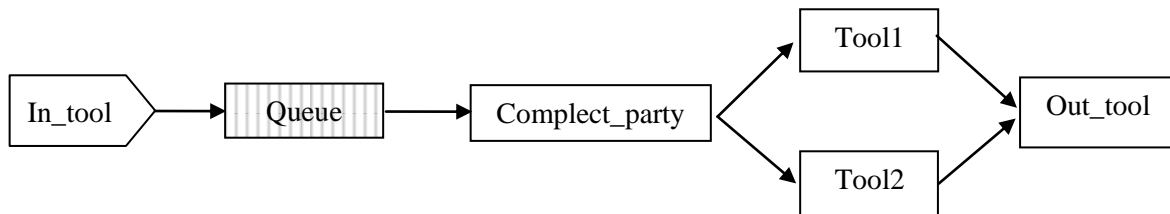
Моделирование статических аспектов модели

Статический аспект системы представляется набором компонентов модели. К компонентам модели, формализуемой с помощью MICIC4, относятся устройства и транзакты. Данные элементы формального уровня представляются классами. Использование классов позволяет задавать атрибуты и методы устройств имитационной модели. Диаграмма классов, представляющая компоненты модели, позволяет отобразить отношения между устройствами, наиболее используемыми из которых будут генерализация и агрегация.

Для повышения эффективности проектирования программы модели в диаграмме классов изначально размещаются такие классы, как Generator, Server и Absorber.

Система моделирования MICIC 4 использует многоуровневую структуру модели. Под этой особенностью понимается положение, что каждый узел может иметь собственную подструктуру. Для представления таких вложенных структур в диаграмме классов модели используется агрегация классов.

Для примера рассмотрим модель производственного участка деталей. Приведем потоковую граф-схему выбранного объекта моделирования.



In_tool – входной станок

Queue – очередь деталей типа 1 и 2

Complect_party - рабочее место комплектации партии деталей

Tool1 и Tool2 – обрабатывающие станки

Out_tool – выходной станок

Рисунок 1 – Граф-схема производственного участка деталей

Построим компоненты модели в Rational Rose с использованием UML. Для визуализации компонентов модели используем диаграмму классов, приведенную на рисунке 2. Имена классов соответствуют стандартам именования в Object Pascal.

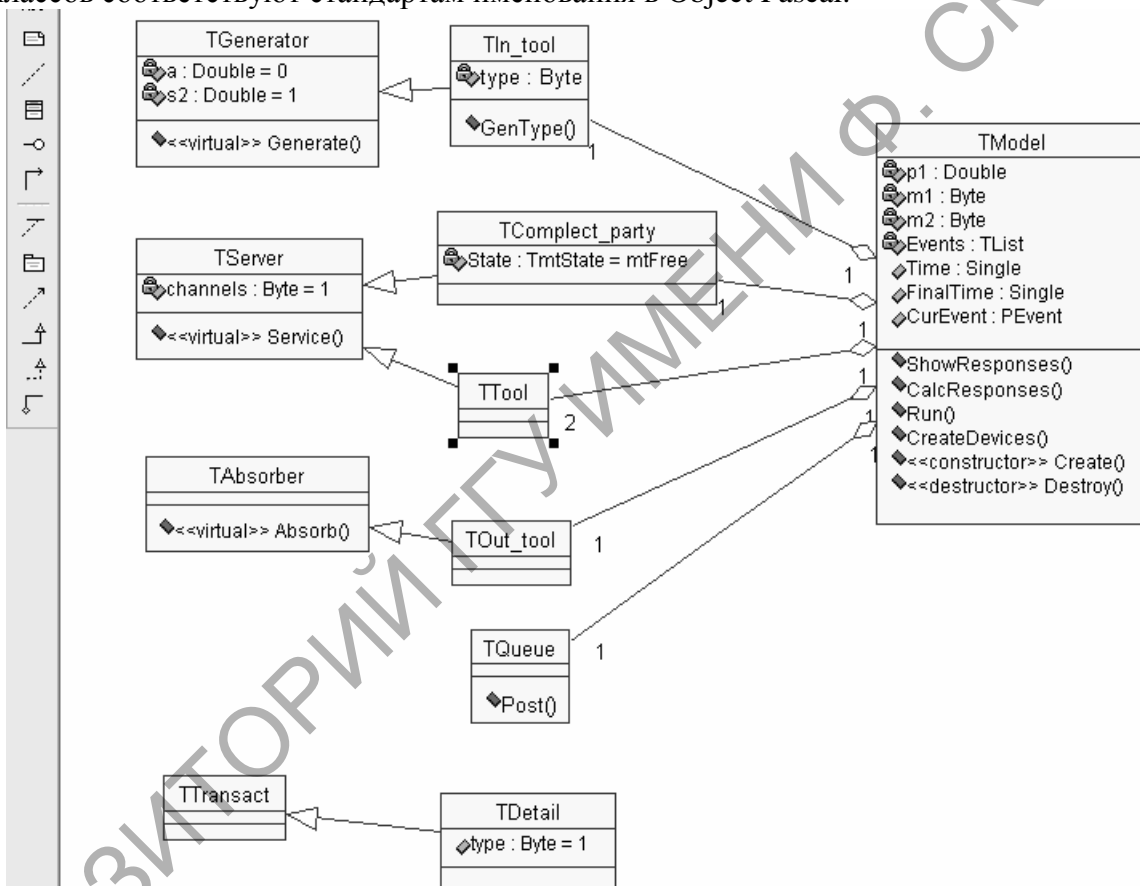


Рисунок 2 – Компоненты модели, представленные с использованием нотации UML

Представление динамики модели

Для моделирования динамики модели используется потоковая граф-схема, в которой элементы схемы – версии устройств, дуги – маршруты перемещения транзактов. Для адекватного представления потоковой граф-схемы используется диаграмма коопераций. Как известно, в данной диаграмме отражаются структурные и динамические аспекты системы. Суть диаграммы – отобразить в визуальной форме обмен сообщениями между классами, в нашем случае представляющие устройства модели. Сообщения должны быть синхронными для отображения последовательности перемещения транзактов. По определению диаграммы коопераций все сообщения имеют уникальные номера, что определяет однозначность пере-

мещения наряду с указанием направления связи. На рисунке 3 изображена потоковая граф-схема модели с использованием предложенной схемы структуризации.

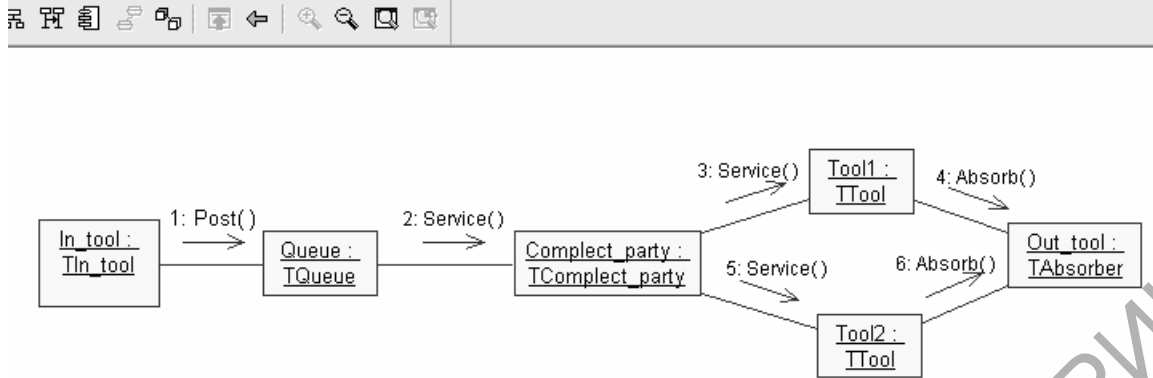


Рисунок 3 – Представление динамики модели с использованием языка UML

Рассмотрим вопрос представления механизма обслуживания с использованием языковых средств UML. Согласно концепции системы MICIC4 механизм обслуживания представляет собой процесс, то есть множество многофункциональных активностей, объединенных в виде связанного ориентированного графа, в котором дуги показывают пути передачи управления между активностями. В механизме обслуживания допускаются условные переходы и циклы. Для представления механизма обслуживания предлагается использование диаграммы деятельности, являющейся одной из разновидностей диаграмм взаимодействия. Вершины в данной диаграмме – состояния действия, в нашем случае интерпретируются как активности. В диаграмме деятельности есть возможность отображения условных переходов и распараллеливания потока вычислений. Очевидно, что обслуживаемому устройству можно соотнести набор механизмов обслуживания, который определяется набором обслуживаемых транзактов. В этом случае все механизмы обслуживания можно свести в одну диаграмму деятельности, разделив обслуживание транзактов различными плавательными дорожками на диаграмме, указав имена транзактов в качестве имен дорожек. В качестве примера приведем механизм обслуживания, построенный в среде Rational Rose для устройства TTool.

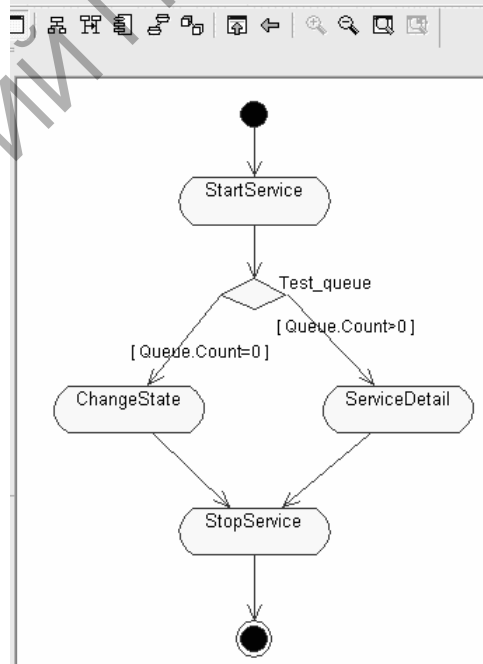


Рисунок 4 – Механизм обслуживания для устройства TTool

Заключение

Предложенная схема структуризации дает теоретический фундамент для построения специализированного CASE-средства, которое можно отнести в класс моделирующего инструментария и которое предназначено для визуального проектирования шаблона программы имитационной модели с использованием языка UML. В использовании языка UML заключается научная новизна приведенной схемы структуризации. Практическая значимость заключается в том, что для проектирования программы модели не требуется знаний по моделированию, а следовательно, на данном этапе могут использоваться специалисты в предметной области.

Новое CASE-средство снабжено механизмом генерации шаблона программы модели, что повышает эффективность создания имитационной модели в целом.

Abstract. The structuring of a simulation model based on the Unified Modeling Language is considered in this paper.

Литература

1. Лоу, А. Имитационное моделирование. Классика CS. 3-е изд. / А. Лоу, В. Кельтон // СПб.: Питер; Киев: BHV, 2004. – 847 с.
2. Максимей, И. В. Имитационное моделирование на ЭВМ / И. В. Максимей // Москва: Радио и связь, 1988. – 232 с.
3. Левчук, В.Д. Программно-технологические комплексы имитации сложных дискретных систем / В. Д. Левчук, И. В. Максимей // Гомель: ГГУ им. Ф.Скорины, 2006.– 263 с.
4. Буч, Г. Язык UML. Руководство пользователя. 2-е изд. / Г. Буч, Д. Рамбо, И. Якобсон // Пер. с англ. Мухин Н. – Москва: ДМК Пресс, 2007. – 496 с: ил.

Гомельский государственный
университет имени Ф. Скорины

Поступило 15.04.07