

Визуальный мониторинг имитационного эксперимента на основе технологии VRML

А. С. ПОМАЗ

Введение

Визуальный мониторинг имитационного эксперимента является одним из эффективных методов верификации имитационной модели [1] по той простой причине, что при имитационных исследованиях модель максимально приближается к реальной системе. Визуализация процесса моделирования позволяет гораздо проще и доступнее постигать пользователю его суть. Во-первых, визуализация позволяет исследовать внутренний механизм функционирования модели, тем самым обеспечивая её прозрачность для пользователей. Во-вторых, визуализация даёт возможность проверки адекватности модели, позволяя проникнуть в самую суть поведения системы. В-третьих, визуализация исследований находит поддержку у многих заказчиков, желающих помимо статистических данных наблюдать процесс моделирования, как один из элементов качественной презентации.

В настоящее время на рынке существуют многочисленные пакеты, предназначенные для визуального представления различной информации. В данном классе программного обеспечения наблюдаются тенденции по выводу в мировую сеть на основе языка Virtual Reality Modeling Language (VRML) [2]. Его возрастающее использование обусловлено следующими причинами:

- этот язык является открытым, и для начального процесса обучения не нужно будет покупать дорогие пакеты по созданию VRML-анимации, а для демонстрации продуктов достаточно будет иметь стандартный web-браузер;
- различные узлы VRML, а также методы прототипирования позволяют получать объекты различной степени детализации, повторно использовать объект (метод порождения), что в конечном итоге положительно сказывается на качестве анимации;
- VRML довольно широко применяется в автоматизированном проектировании, а во многих сферах деятельности именно VRML-модели становятся практически стандартной формой изображения объектов, например, в развиваемой на сегодняшний день концепции Digitale Fabrik VRML-модели занимают ведущее место;
- VRML обладает возможностью сопряжения с другими программными средствами. Это стало возможным благодаря ECMA Script – языку последовательного программирования, что отличает его от VRML, являющегося декларативным языком.

Слабая сторона VRML-моделей состоит в том, что вопросы взаимосвязи таких моделей с имитационными моделями не разработаны. Однако это возможно, так как любая VRML-модель в конечном итоге – код ASCII.

Таким образом, задача выбора платформы визуализации была решена в пользу языка VRML. Следующая задача заключается в построении концепции визуального мониторинга имитационного эксперимента на основе технологии VRML.

Обзор технологии VRML

VRML позволяет описывать трехмерные образы и текст с целью их последующего отображения с помощью различных программных средств. Это могут быть как специализированные VRML-браузеры, так и вспомогательные утилиты просмотра VRML-документов, добавленные к обычному web-браузеру. Просматриваемая трехмерная VRML-сцена пред-

ставляет собой просто совокупность сменяющих друг друга трехмерных образов. В рамках сцены, например такой, как образ дома, создатель сцены может позволить прогуливаться по дому, переходя из одной трехмерной комнаты в другую.

VRML организует узлы в иерархическую структуру, называемую графом сцены, которая определяет порядок закраски и прорисовки браузером узлов. Узел может наследовать характеристики ранее определенного узла. Другими словами, особенности оформления предыдущего узла могут повлиять на закраску последующего узла.

Узел VRML представляет собой объект. Аналогично любой объект в сцене графа представляет собой узел, который описывает специфический тип объекта. Например, узел может быть сферой, конусом, цилиндром или кубом. Эти типы узлов являются узлами простой геометрии. VRML также поддерживает другие типы узлов, такие как узлы материала и текстуры, которые являются узлами свойств. Граф сцены VRML использует узлы свойств, чтобы определить материал поверхности для последовательности всех поверхностей. Эта информация предназначена для браузера, который применяет ее ко всей последовательности изображаемых форм.

В версии 1.0 спецификации VRML определены 36 узлов. Спецификация VRML 2.0 вводит такие дополнительные узлы, как:

- элементы управления геометрией и цветовые свойства форм;
- геометрические сенсоры для поддержки drag-and-drop-операций;
- узлы группировки, которые поддерживают объект «определение столкновения».
- атрибуты освещенности (например, дымка);
- информация о сцене для браузеров.

Методика визуального мониторинга имитационного эксперимента

Методика визуального мониторинга имитационного эксперимента предполагает использование системы моделирования MICIC4 [3] и среды визуального проектирования имитационных моделей M-BUILDER [4]. Общая схема взаимодействия программных средств приведена на рис. 1. Сама методика включает следующие этапы:

- построение визуальной структуры модели в M-BUILDER;
- построение программы-имитатора средствами MICIC4;
- реализация однопрогонного имитационного эксперимента, в результате чего создается журнал событий;
- генерация VRML-сцены на основе журнала событий и визуальной структуры модели;
- мониторинг имитационного эксперимента в web-браузере либо другом приложении.

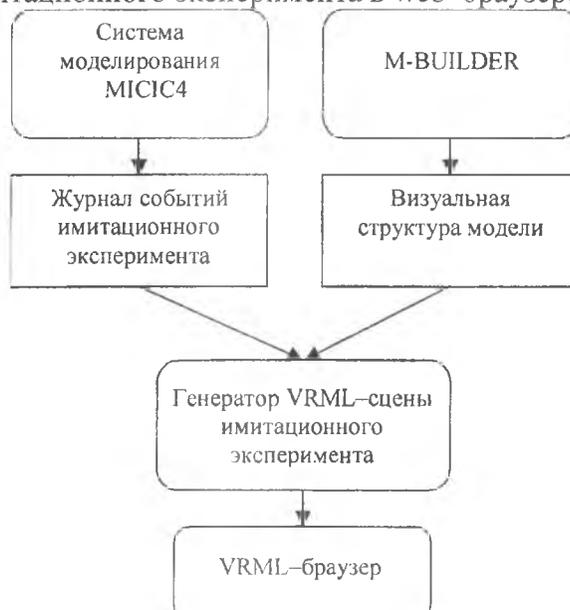


Рис. 1. Схема взаимодействия программных средств

Практические аспекты реализации методики

Входные сущности для методики следующие:

- иерархическая структура вложенности компонентов и граф-схема взаимодействия устройств и транзактов имитационной модели;
- визуальная структура модели, проектируемая в M-BUILDER и включающая описание статических объектов VRML-сцены (геометрию, связи, ...);
- журнал событий имитационного эксперимента, содержащий, в частности, все перемещения динамических элементов вместе с временами.

Основываясь на изложенном подходе, в среде программирования Delphi 6 был разработан генератор VRML-сцены эксперимента. Он анализирует визуальную структуру имитационной модели, находящуюся в базе данных, и затем создает текстовый код VRML-сцены. Таким образом, генерация VRML-сцены в данном приложении реализуется в три этапа:

1. определяется количество узлов и связи объектов посредством анализа визуальной структуры;
2. создается динамика VRML-сцены по журналу событий;
3. генерируется окончательный VRML-код визуализации.

Апробация предложенной методики визуального мониторинга была проведена на имитационной модели производственного участка обработки деталей. Программа-имитатор была реализована с помощью системы моделирования MICIC4. В результате постановки однопрогонного имитационного эксперимента был создан журнал событий в обычном текстовом формате. После конструирования в M-BUILDER визуальной структуры модели и дальнейшей обработки всех входных сущностей в генераторе VRML-сцены был получен VRML-код, проигранный в стандартном web-браузере MS Internet Explorer. Наблюдаемые маршруты движения деталей полностью соответствовали концептуальной модели.

Заключение

Разработанная методика, ядром которой является генератор VRML-сцены имитационного эксперимента, в настоящее время тестируется на ряде более сложных имитационных моделей. При этом обнаружилось, что еще одним из достоинств предложенной методики визуализации является возможность наблюдения за динамикой эксперимента как на локальной ПЭВМ, так и загрузка и контроль за динамикой модели через сеть Интернет.

Abstract. The author offers the visual monitoring technique of a simulation experiment based on the VRML modern language.

Литература

1. А. Лоу, В. Кельтон, *Имитационное моделирование. Классика CS*. 3-е изд., Санкт-Петербург; Киев, BHV, 2004.
2. Э. Тител, *Создание VRML-миров*, Киев, BHV, 2004.
3. В. Д. Левчук, *Базовая схема формализации системы моделирования MICIC4*, Проблемы програмування, № 1 (2005), 85–96.
4. А. С. Помаз, *Основные принципы разработки визуального конструктора программ имитационных моделей*, Известия Гомельского государственного университета им. Ф. Скорины, № 4(25) (2004), 35–39