

**В. Д. Левчук, П. Л. Чечет, В. В. Старченко,
В. С. Давыдов, А. С. Помаз**

ИНТЕГРАЦИЯ ИМИТАЦИОННЫХ МОДЕЛЕЙ ПРОИЗВОДСТВЕННЫХ ПРОЦЕССОВ В КОРПОРАТИВНУЮ ИНФОРМАЦИОННУЮ СИСТЕМУ ЗАКАЗЧИКА

Рассматриваются вопросы внедрения и использования имитационных моделей различных производственных процессов в корпоративной информационной системе предприятия. Предлагаемые решения позволяют значительно повысить эффективность работы с имитационной моделью для заказчика, и, таким образом, упростить решение задач исследования и поиска путей совершенствования показателей производственного процесса предприятия.

Введение

Преимущества применения имитационного моделирования известны многим специалистам и руководителям. Активное использование различных программных средств в управлении современными производственными процессами ставит задачу интеграции разработанной имитационной модели в информационную систему заказчика. Данная задача усложняется тем, что существующие информационные системы отличаются большим разнообразием возможностей по способам взаимодействия с ними.

Возможный путь решения данной задачи представляется оснащением разрабатываемых имитационных моделей специальными средствами по организации экспорта результатов и импорта входных данных с использованием популярных форматов представления данных, поддержка которых присутствует в большинстве современных информационных систем.

Контейнеры имитационных моделей

При разработке имитационных моделей с использованием системы моделирования MICIS 4 [1] имеется большая возможность по оснащению моделей модулями экспорта/импорта данных. Это объясняется базированием системы моделирования MICIS 4 на универсальном языке программирования C++. Использование полных возможностей C++ позволяет реализовывать практически любые схемы взаимодействия имитационной модели с информационной системой заказчика.

Одним из возможных путей реализации имитационной модели для упрощения ее интеграции в систему заказчика является построение программы модели в виде программного модуля, который позволяет использовать различные контейнеры для ее выполнения соответственно с требованиями. В качестве контейнеров для имитационной модели могут выступать консольное приложение, Web-приложение, приложение с графическим интерфейсом пользователя, АСУ заказчика и другие варианты.

Пример реализации программы имитационной модели сборочно-разборочного производства [2] с использованием в качестве контейнера консольного приложения приведен на рис. 1.



Рис. 1. Консольная программа имитационной модели

Наиболее простым способом хранения данных, обладающим возможностями кроссплатформенности, является использование текстовых файлов. Работа с данными из текстовых файлов доступна во многих программных продуктах, работающих не только под управлением операционных систем Microsoft. В зависимости от требований, структура текстовых файлов может быть адаптирована для просмотра пользователем или для непосредственного открытия в требуемых программах. Примером такой реализации может выступать открытие файла входных параметров, адаптированного для открытия в табличном процессоре Microsoft Excel (рис. 2).

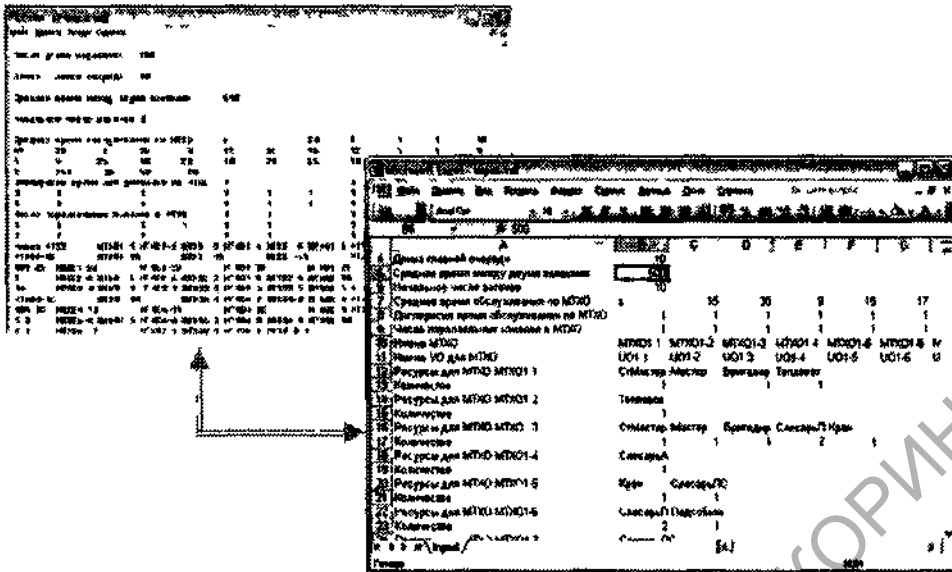


Рис 2 Адаптация структуры текстового файла

Вторым примером возможной реализации контейнера программы имитационной модели производственных процессов является Win32 программа с графическим пользовательским интерфейсом (рис 3) В таком случае пользователь получает оконное приложение, в элементы управления которого заносятся значения параметров и переменных ИМ. Нажатие кнопки «Запуск» приводит к запуску процесса моделирования, по окончании которого результаты помещаются на закладке «Отклики». Использование такого контейнера позволяет значительно упростить интерактивное взаимодействие пользователя с моделью, позволяя оперативно вручную изменять входные данные и анализировать поведение модели.

Третьим примером возможной реализации имитационной модели [3] является использование Web-технологий. В этом случае программный модуль имитационной модели встраивается в интернет-сервер а

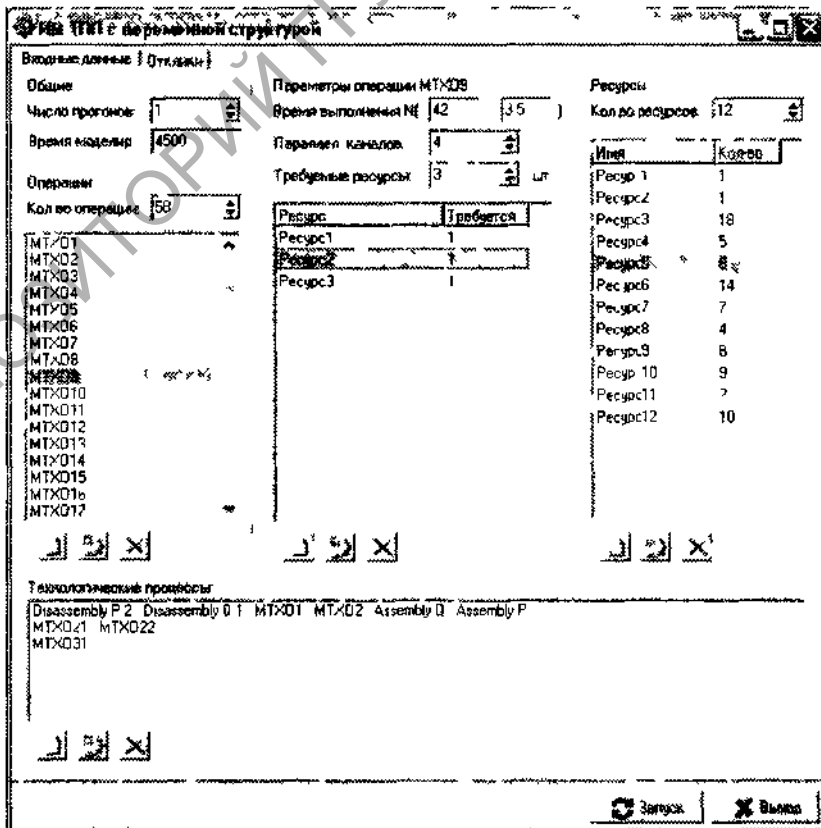


Рис 3 Оконная реализация программы модели

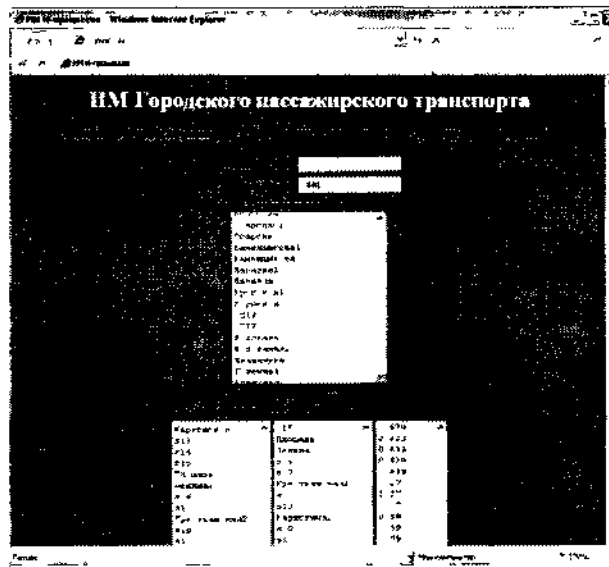


Рис 4 Web-реализация программы модели

взаимодействие пользователя с моделью происходит через сеть. При этом пользователь работает с имитационной моделью через установленный у него браузер. Пример работы с реализованной имитационной моделью через браузер Windows Internet Explorer 7 представлен на рис. 4.

Реализация импорта данных из корпоративной информационной системы

Мнообразие доступных контейнеров позволяет обеспечить удобство работы заказчика с имитационной моделью в большинстве возможных случаев. При использовании текстовых файлов для задания информации для имитационной модели может возникнуть сложность с редактированием таких файлов пользователем, если структура данных является слишком сложной. В этом случае целесообразным становится разработка отдельных автономных программ-редакторов параметров. Примером такой разработки является редактор параметров имитационной модели маршрутной транспортной сети, реализованный в среде табличного процессора Microsoft Excel с использованием встроенного языка программирования Visual Basic. Входные данные программы этой имитационной модели хранятся в текстовых файлах, для их редактирования может быть использован как произвольный текстовый редактор, так и разработанный редактор параметров. Подпрограммы на языке Visual Basic автоматически считывают и преобразуют входные данные при его запуске, а также предлагают их сохранение при закрытии редактора (рис. 5).

Также программа имитационной модели может быть оснащена модулем экспорта результатов в популярный формат HTML, данные в котором могут быть размещены на Web-узле и быть просмотрены с помощью любого браузера, например Windows Internet Explorer (рис. 6).

Пассажир	Начала движения	Выхода	Пересадки	Последн пересадки	Назначения	Остановки	Маршрут
Р	Тимирязев1				Гидроавтоматика1	1v 20v	1v 20v
Р	Вокзал1				Гидроавтоматика1	1v	1v
С	Универсам1				Гидроавтоматика1	1v 20v	1v 20v
У	Универсам1				Электростанция1	1v 20v	1v 20v
Р	Гидроавтоматика2				Гидроавтоматика2	1v 20v	1v 20v
Р	Вокзал1				Вокзал2	1v	1v
У	Универсам1				Универсам2	1v 20v	1v 20v
П	Маяковска	Тимирязев2			Огарько2	1v 30v	1v 30v
					Гидроавтоматика1	1v 20v	

Рис 5 Редактор входных данных модели городской транспортной сети

Результаты моделирования

Результаты моделирования

Результаты моделирования

ВРЕМЯ ОЖИДАНИЯ ТРОЛЛЕЙБУСА

	000-	100-	200-	300-	400-	500-	600-	700-	800-	900-	1000-	1100-	1200-	1300-	1400-	1500-	1600-	1700-	1800-	1900-	2000-	2100-	2200-	2300-	
Автоматика1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Автоматика2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Аппаратура1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	12,33	4,01	4,00	3,93	3,93	3,50	3,29	4,12	3,93	4,38	16,64	43,59	35,33	3,67	3,26	3,4		
Аппаратура2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Вк1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	14,20	5,28	4,25	4,21	4,26	4,51	6,25	4,16	3,74	4,08	3,35	0,00	106,46	2,00	3,54	5,5		
Вк2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	34,01	2,14	4,62	3,80	3,29	2,09	2,08	10,66	2,95	2,64	2,01	60,21	119,51	157,02	20,86	4,4		
Визит1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	17,13	3,74	6,06	6,47	6,94	4,77	5,16	5,14	5,68	4,25	0,00	81,50	107,96	88,84	5,94	3,5		
Визит2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	13,02	1,80	4,36	4,81	4,61	2,55	2,31	7,04	4,14	2,96	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Венко1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	22,25	5,14	4,85	5,09	4,94	4,57	8,16	7,09	5,83	2,40	30,81	70,68	107,42	135,22	8,07	3,9		
Венко2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	29,88	1,98	3,61	2,23	1,95	1,79	1,26	8,90	2,76	2,91	2,75	0,00	104,16	144,89	134,95	3,4		
Техника1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	23,54	4,81	6,35	5,16	6,76	4,59	4,65	9,16	5,27	3,07	18,34	79,99	123,21	157,32	7,52	4,6		
Тга1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	30,42	2,60	2,43	2,26	2,23	2,26	1,67	3,86	3,20	3,68	27,87	64,55	107,27	2,57	3,71	4,8		
Тга2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Техника2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
	0,00	0,00	0,00	0,00	0,00	0,00	0,00	28,96	6,44	5,60	6,00	6,47	7,10	4,50	11,12	7,47	4,61	29,06	49,11	111,54	143,73	130,95	4,6		

Рис. 6. Отклики модели в формате HTML

При реализации обеспечения импорта данных в программу имитационной модели представляется целесообразным использование данных в широко распространенных форматах XML, HTML и XLS. Классы, обеспечивающие работу программы ИМ с файлами форматов XML, HTML и XLS, взаимодействуют по следующей схеме (рис. 7).

Уровень структуры данных обслуживает интерфейс взаимосвязи MICIC4 с различными форматами входных и выходных данных. Этот уровень организуется посредством абстрагированных от MICIC4 классов для работы с документами форматов XML, XLS. Взаимосвязь между уровнем MICIC4 и уровнем структуры данных реализует класс PlanOfExperiment, который наследуется от базового класса системы MICIC4 Experiment. Базовыми классами для уровня структуры данных являются:

- PlanOfExperiment – «План Эксперимента», этот класс является наследником базового класса «Эксперимент» (Experiment) MICIC4;
- PrmRetriever используется для получения параметров, он включает методы, которые переопределены для XML и XLS формата в классах XMLRetriever и XLSRetriever соответственно;
- «Парсер» (Parser) совершает посимвольный разбор содержимого файла;
- «XLS Парсер» (XLSParser) создает дерево параметров;
- «Дерево» (Tree) представляет некоторую древовидную структуру (например, XML документ);
- «Генератор Дерева» (TreeMaker) создает дерево, определяющее структуру xml документа;
- «Узел» (Nod) представляет собой единицу (узел) древовидной структуры, реализованной классом Tree;
- «Атрибуты» (Attributes) определяет множество пар «ключ» – «значение».

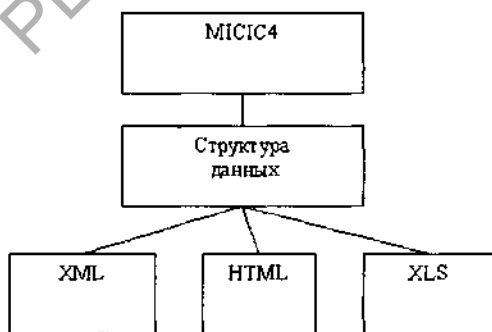


Рис. 7. Схема взаимодействия классов

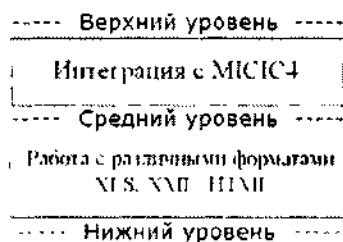


Рис. 8. Функциональное назначение уровней

Таким образом, работа с данными отделена от реализации самой модели. Причем работа с параметрами не зависит от типа файла, с которым работает пользователь. Для чтения и сохранения данных в Excel используется COM-технология (Component Object Model) [4].

Для обеспечения гибкости при разработке схемы взаимодействия классов были выделены три основных независимых слоя функционирования. Под независимостью понимается тот факт, что при использовании одного из слоев не обязательно знать, как устроены и функционируют нижние слои. Самый верхний слой имеет узкий интерфейс, который, с одной стороны, легок для использования и не требует много времени для изучения, с другой – обеспечивает достаточный набор средств для реального применения.

Классы верхнего слоя предназначены для использования в системе моделирования MICIC4. Классы среднего уровня предоставляют интерфейс для работы с данными из файлов в формате XML и XLS и формирование документов в указанных форматах. Нижний уровень представлен следующими классами:

- классы, позволяющие создавать необходимые структуры данных (ассоциативные массивы для хранения атрибутов тегов, деревья для представления XML документов);
- классы, обеспечивающие обмен данными с XLS файлами;
- классы, производящие разбор XML и XLS документов;
- классы для формирования отчетов (таблиц) в формате HTML и XLS.

Функциональное назначение уровней приведено на рис. 8.

Верхний уровень представлен всего одним классом PlanOfExperiment. Предлагается следующая модель использования данного класса:

- описывается наследник от класса PlanOfExperiment;
- в наследнике переопределяется виртуальный метод execute, именно в данном методе будут производиться все необходимые действия и формироваться отчет;
- при работе с экземпляром нового класса необходимо проинициализировать обработчик отчета.

Средний уровень представлен более разнообразным числом классов по сравнению с верхним уровнем. Сюда входят семь классов для высокоуровневой работы с получением данных из файлов разных форматов и сохранением полученных результатов.

Из рисунка 9 видно, что нижний уровень состоит из трех наборов классов:

- классы для представления структур данных;
- классы для работы с XML документами;
- классы для работы с табличным процессором Excel.

Классы для представления структур данных описывают структуру документов в виде дерева. Это позволяет единым образом организовать работу как с XML-документами, так и таблицами Excel. Взаимодействие обеспечивается посредством методов, реализующих следующую функциональность:

- вызов функции указанного объекта по его имени;
- получение свойства объекта по его имени;
- установка свойства объекта по его имени.

Заключение

Использование текстовых файлов и модулей экспорта/импорта данных в широко используемые форматы позволяет значительно упростить решение задачи интеграции программ имитационных моделей в существующую программно-информационную систему заказчика. Реализация импорта данных из форматов HTML, XML и XLS позволяет организовывать передачу данных их различных программных продуктов в программу имитационной модели. Использование контейнеров позволяет внедрить программу имитационной модели в качестве подсистемы в существующую систему заказчика. Все эти решения позволяют значительно повысить эффективность работы с имитационной моделью для заказчика и, таким образом, упростить решение задач исследования и поиска путей повышения эффективности производственного предприятия.

Литература

1. Левчук, В. Д. Программно-технологические комплексы имитации сложных дискретных систем / В. Д. Левчук, И. В. Максимей; М-во образования РБ. Гомельский государственный университет имени Франциска Скорины. – Гомель : ГГУ имени Ф. Скорины, 2006. – 263 с.
2. Чечет, П. Л. Стратегии набора ресурсов в имитационной модели технологических процессов производства с иерархической структурой / П. Л. Чечет // Известия ГГУ имени Ф. Скорины. – Гомель. – 2007. – № 6. – С. 146- 149.

3. Галушко, В. Н. Методика имитационного моделирования работы городского транспорта / В. Н. Галушко [и др.] // Электронное моделирование. – 2006. – Т. 28, № 2. – С. 79–95.

4. Оберг, Р. Д. Технология SOM+. Основы программирования: учеб. пособие: пер. с англ. / Р. Д. Оберг. – М. : Изд. дом «Вильямс», 2000. – 480 с.

Левчук Виктор Дмитриевич, заведующий кафедрой автоматизированных систем обработки информации Гомельского государственного университета имени Франциска Скорины, кандидат технических наук, доцент, lv@gsu.by

Чечет Павел Леонидович, ассистент кафедры автоматизированных систем обработки информации Гомельского государственного университета имени Франциска Скорины, pchehchet@gsu.by

Старченко Владимир Владимирович, доцент кафедры математических проблем управления Гомельского государственного университета имени Франциска Скорины, кандидат технических наук, starchenko@gsu.by

Давыдов Владимир Семенович, доцент кафедры автоматизированных систем обработки информации Гомельского государственного университета имени Франциска Скорины, кандидат технических наук, доцент, davydov@gsu.by

Помаз Андрей Сергеевич, ассистент кафедры математических проблем управления Гомельского государственного университета имени Франциска Скорины, rotaz@gsu.by