

УДК 004.89

Дата подачи статьи: 24.10.14

DOI: 10.15827/0236-235X.109.012-016

## **АРХИТЕКТУРА СИСТЕМЫ ПОДДЕРЖКИ ПРОЕКТИРОВАНИЯ АГЕНТОВ ДЛЯ ИМИТАЦИОННЫХ МОДЕЛЕЙ СЛОЖНЫХ СИСТЕМ**

*(Работа выполнена при частичной финансовой поддержке РФФИ, грант № 14-07-31298)*

*А.И. Павлов, к.т.н., старший научный сотрудник, asd@icc.ru;*

*А.Б. Столбов, младший научный сотрудник, stolboff@icc.ru*

*(Институт динамики систем и теории управления СО РАН,  
ул. Лермонтова, 134, г. Иркутск, 664033, Россия)*

В работе рассматриваются проблемы автоматизации процесса создания имитационных моделей сложных систем. В качестве основной парадигмы моделирования выбран многоагентный подход. Предложена архитектура системы поддержки проектирования многоагентных моделей. Основными задачами системы являются формализация и использование концептуальной модели предметной области при создании агентов, поддержка процесса преобразования концептуальной модели в агентную модель, снижение квалификационных требований в области программирования для специалистов-предметников, организация совместной работы коллектива исследователей. Оригинальность предлагаемой системы заключается в явном представлении концептуальной модели за счет описания сущностей и связей предметной области в форме онтологии и в использовании системной базы знаний, содержащей правила преобразования онтологии в агентную модель. При использовании системы специалисты-предметники разрабатывают структуру и поведение агентов с помощью декларативного описания. Возможность декларативного описания агента предложено осуществлять на основе хранения информации о структуре и поведении агента отдельно от его реализации. Для этого предлагается разработать механизм описания агента, позволяющий единообразно создавать классы агентов, существенно отличающихся друг от друга по своей структуре и поведению. Реализующий такой механизм типовой (унифицированный) агент будет обладать следующей структурой: блок состояний, содержащий значения характеристик агента; декларативный блок, содержащий стратегию поведения агента, формализованную в виде производственных правил; императивный блок, содержащий программную реализацию расчетных процедур, используемых в процессе логического вывода. Для реализации декларативного блока типового агента использованы современные инструментальные средства проектирования экспертных систем.

**Ключевые слова:** *моделирование сложных систем, агентное имитационное моделирование, система поддержки проектирования агентов, производственная экспертная система, системы основанные на знаниях, декларативный стиль программирования.*

При проведении исследований, моделировании и в процессе проектных разработок термин «сложная система» ассоциируется с объектом, состоящим из множества отдельных иерархически устроенных частей, функционирующих в тесном взаимодействии и составляющих единое целое [1]. Например, как сложную систему можно рассматривать город, крупное предприятие, экономическую систему и т.п. К настоящему времени накоплен богатый опыт в изучении сложных систем, разработаны разнообразные методологические подходы [1–5]. К основным этапам моделирования сложных систем относятся построение концептуальной модели системы и ее формализация, алгоритмизация модели системы и ее машинная реализация, получение и интерпретация результатов моделирования системы.

Современные средства автоматизации моделирования (AnyLogic [6], GPSS World [7], iThink [8], Matlab Simulink [9], PowerSim [10] и др.) предоставляют пользователю возможность определять отношения между компонентами модели и правила их сборки. Основными задачами, решаемыми этими средствами автоматизации, являются построение непротиворечивой и эффективной (по времени и точности) компьютерной модели, соответствующей заданной постановке, многовариантные расчеты с целью оптимизации и настройки

параметров, организация обмена данными между модулями, отображение результатов. Однако существующие средства уделяют недостаточно внимания этапу создания концептуальной модели и поддержке ее актуальности в рамках итеративного процесса разработки имитационной модели. Задача обеспечения явного использования формализованного представления концептуальной модели особенно важна при исследовании сложных систем, так как их изучение не укладывается в компетенцию ни одного конкретного специалиста и, следовательно, требует междисциплинарного подхода. Данное свойство сложных систем определяет следующую задачу, которую должны решать системы автоматизации моделирования, – организация совместной работы коллектива специалистов из разных областей знаний: экспертов-предметников, математиков, программистов, инженеров по знаниям. Использование парадигмы агентного моделирования позволяет учесть специфику междисциплинарного исследования за счет разделения областей компетенции экспертов по группам агентов. Для рассматриваемой проблемы важным преимуществом агентного моделирования является то, что в отличие от моделей системной динамики или дискретно-событийных моделей [11] нет необходимости подробно определять поведение системы в целом – разработку

агента можно осуществить, опираясь на знания об индивидуальной логике поведения объектов предметной области.

Разнообразие структур и поведения сложных систем в зависимости от задач исследования обуславливает высокую сложность автоматизации процесса разработки агентов, поэтому в большинстве современных инструментальных средств моделирования данный процесс осуществляется путем написания кода на императивном языке программирования с минимальным уровнем автоматизации. Это существенно уменьшает возможности использования подходов имитационного моделирования специалистами-предметниками в процессе исследований, так как они, как правило, редко обладают достаточным уровнем квалификации в области программирования. Проблемы минимизации участия программистов и инженеров по знаниям при создании агентов и определения поведения агентов преимущественно декларативным путем определяют еще одну задачу исследования.

**Предлагаемые методы и подходы.** Для решения задачи формализации и использования концептуальной модели предметной области при создании агентов предлагается обеспечить исследователей возможностью совместной разработки онтологии предметной области, содержащей описания понятий предметной области, описания их характеристик и типизированных связей между понятиями. Создание онтологии позволит специалистам-предметникам согласовывать используемые термины, уточнять их структуру и определять новые связи, обусловленные междисциплинарным характером исследований.

При решении задачи поддержки процесса преобразования концептуальной модели в агентную необходимо учитывать междисциплинарную специфику исследования сложной системы, когда объект моделирования представляется как система сравнительно однообразных взаимодействующих частей различной природы (агент или группа агентов), каждая из которых относится к некоторой предметной области.

Для поддержки исследователя в процессе создания агентной имитационной модели будет использоваться интегрированная экспертная система, база знаний которой содержит продукционные правила, управляющие процессом формирования структуры и поведения агента. На первом этапе этого процесса на основе понятий концептуальной модели определяются характеристики агента. Далее для полученной структуры определяется база знаний агента, которая содержит три составляющие поведения: собственная стратегия принятия решений, зависящая от структуры его характеристик; взаимодействие с внешней средой; реакция на действия других агентов. Поддержка проектирования будет заключаться в последовательном

определении каждой составляющей поведения на основе анализа концептуальной модели. Отношения между понятиями концептуальной модели (структурные, причинно-следственные, наследование и т.п.) могут служить основанием для выявления потенциальной связи между агентами, содержащими эти понятия в качестве своих характеристик. Определение конкретных связей между агентами и/или средой в общем случае производится в результате интерактивного диалога с исследователем. В ходе такого диалога система поддержки может по заданным шаблонам генерировать вопросы о характере связи, например, насколько изменится характеристика  $X_i$  агента  $I$  при увеличении характеристики  $Y_i$  внешней среды. В некоторых случаях ответы на подобные вопросы могут быть найдены при анализе онтологии предметной области и статистической информации. В процессе создания правил, описывающих поведение агента, может быть использована заранее заданная база вычислительных процедур, позволяющих преобразовывать численные значения характеристик. Для редактирования полученных правил и создания новых будут использоваться современные средства визуального представления и обработки знаний (например Protege [12]).

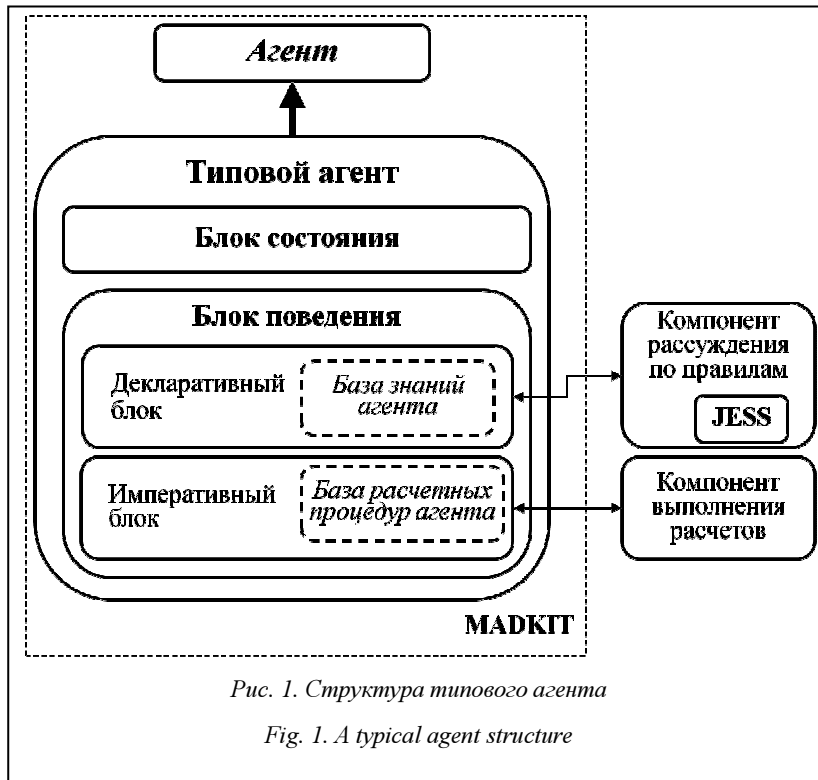
Полученное с помощью интегрированной экспертной системы декларативное описание агента обладает рядом преимуществ. Во-первых, оно позволяет снизить квалификационные требования в области программирования для специалистов-предметников. Во-вторых, так как агенты отличаются друг от друга только набором характеристик и базой знаний, появляется возможность обеспечить реализацию всех агентов на базе одного специализированного класса (типового агента).

Типовой агент обладает следующей структурой:

- блок состояния, содержащий значения характеристик агента;
- декларативный блок, содержащий стратегию поведения агента, формализованную в виде продукционных правил;
- императивный блок, содержащий ссылки на программную реализацию расчетных процедур, используемых в процессе логического вывода.

В качестве базового средства, реализующего рутинные операции, связанные с организацией взаимодействия агентов, могут использоваться существующие инструментальные средства агентного моделирования (AnyLogic, JADE, MADKIT, NetLogo, Repast, Swarm и др.). Для апробации подхода предлагается использовать свободно распространяемую систему MADKIT [13], таким образом, реализация типового агента будет осуществлена на основе класса «Агент» системы MADKIT (рис. 1).

Для реализации декларативного блока типового агента будут использованы инструменталь-



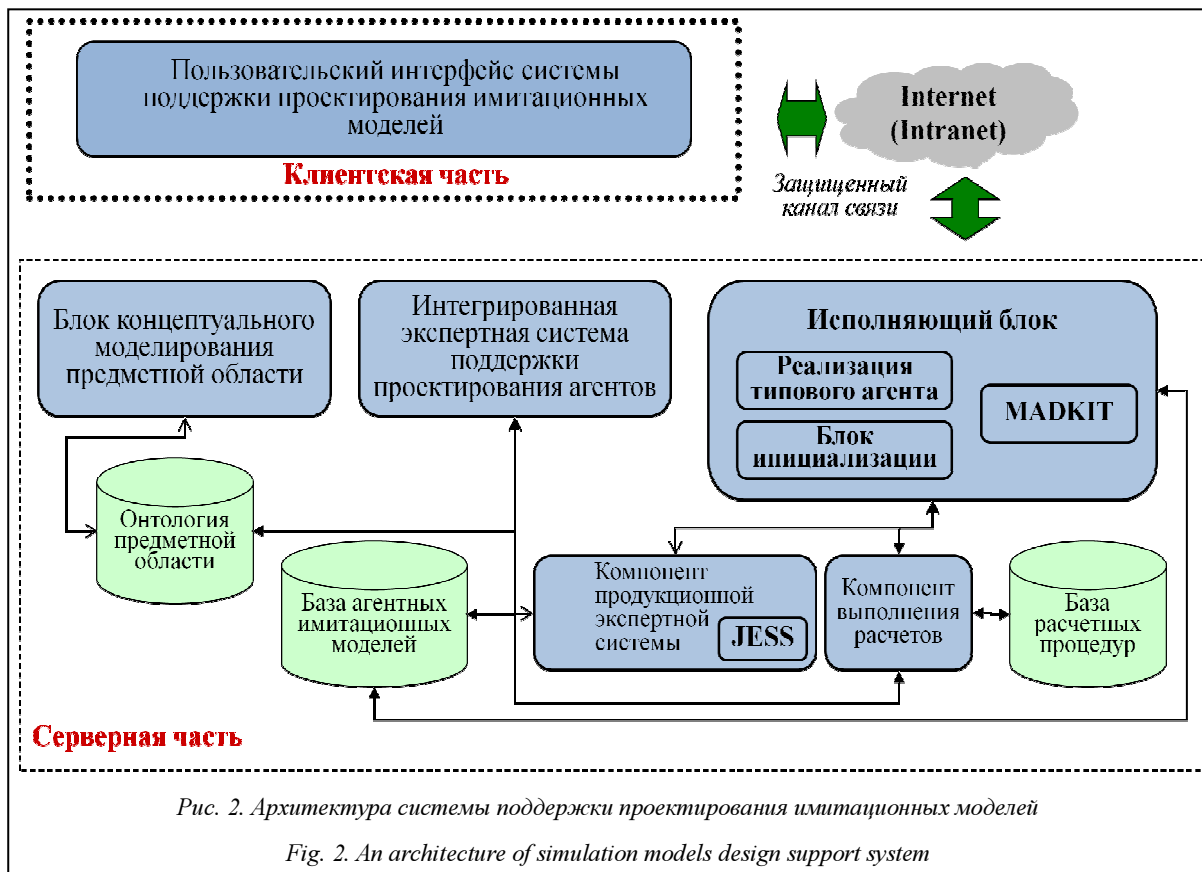
Функциональность типового агента позволяет изолировать информацию о структуре и поведении агента от его программной реализации в стороннем инструментальном средстве имитационного моделирования. Использование такого средства с отлаженной архитектурой позволяет разработчикам сконцентрироваться на решении задачи интеллектуализации и автоматизации процесса разработки имитационной модели.

Архитектура системы поддержки проектирования имитационных моделей. Предлагается использовать архитектуру системы поддержки проектирования интеллектуальных агентов, состоящую из перечисленных ниже блоков (рис. 2).

1. Блок концептуального моделирования предметной области. Результатом работы этого блока является онтология предметной области, содержащая описание рассматриваемой сложной системы в виде иерархии взаимосвязанных объектов и их свойств.

ные средства проектирования экспертных систем JESS [14]. Реализация процедур императивного блока будет осуществляться с использованием языка Java.

Результатом работы этого блока является онтология предметной области, содержащая описание рассматриваемой сложной системы в виде иерархии взаимосвязанных объектов и их свойств.



2. Интегрированная экспертная система на базе JESS, содержащая системную базу знаний, в которой находятся шаблоны, правила и факты, описывающие закономерности процесса создания агентной имитационной модели сложной системы на основе разработанной онтологии предметной области.

3. База расчетных процедур, содержащая описание и сигнатуру методов, доступных для использования агентами в процессе принятия решений. Особенностью включаемых в базу процедур является то, что они должны реализовывать методики вычисления конкретных значений, имеющих содержательный смысл для рассматриваемой предметной области, например, расчет диаметра области сплошного поражения при взрыве некоторого вещества или расчет атмосферных выбросов теплотехностанции в зависимости от ее мощности и типа используемого топлива.

4. База сформированных имитационных моделей, каждая из которых содержит описание агентов, параметры внешней среды и начальные условия.

5. Исполняющий блок, использующий инструментальное средство имитационного моделирования общего назначения MADKIT для обеспечения непосредственного выполнения процесса моделирования.

6. Блок реализации типового агента, обеспечивающий интерпретацию декларативного описания агента и вызов расчетных процедур в процессе имитационного моделирования.

7. Блок инициализации, обеспечивающий создание необходимого количества агентов и определение конкретных параметров внешней среды для выбранной имитационной модели.

Систему поддержки проектирования имитационных моделей предлагается реализовать в виде web-приложения. Разработка блока концептуального моделирования предметной области, интегрированной экспертной системы и исполняющего блока будет осуществлена по технологии тонкого клиента. Защита информации от несанкционированного доступа будет выполнена с помощью шифрования передаваемых данных на основе использования протокола HTTPS. Кроме того, реализация системы в виде web-приложения позволяет обеспечить совместную работу пользователей в процессе создания имитационных моделей.

Предлагаемый подход к разработке системы поддержки проектирования имитационных моделей сложных объектов позволит выполнить их формирование от этапа концептуального описания до получения конкретных расчетов. При этом ее программа будет реализована на основе существующих современных средств разработки экспертных систем, СУБД, систем агентного модели-

рования, программных инструментов для извлечения и представления знаний.

Оригинальность предлагаемой системы заключается в явном представлении концептуальной модели за счет описания сущностей и связей предметной области в форме онтологии и в использовании системной базы знаний, содержащей правила преобразования онтологии в агентную модель. Данные правила должны осуществлять поддержку исследователя при формировании структуры состояний агентов и среды, формировании прототипа декларативного описания поведения агента, при уточнении поведения агента в интерактивном диалоге с пользователем, выявлении потенциальных связей агент–агент и агент–среда.

Предлагаемый авторами подход к автоматизации и интеллектуализации процесса разработки агентов позволит задавать их структуру и поведение специалистам-предметникам с минимальным участием программистов и инженеров по знаниям, а также обеспечивать возможность совместной работы в рамках междисциплинарного исследования, когда каждый эксперт обладает знаниями только о части сложной системы. Неизвестная исследователю информация о деталях структуры и поведения агента может быть выявлена путем анализа онтологии предметной области.

#### Литература

1. Бусленко Н.П. Моделирование сложных систем. М.: Наука, 1978.
2. Гурман В.И., Дыхта В.А., Кашина Н.Ф. и др. Эколого-экономические системы: модели, информация, эксперимент. Н.: Наука, 1987.
3. Месарович М., Мако Д., Такахара И. Теория иерархических многоуровневых систем. М.: Мир, 1973.
4. Самарский А.А., Михайлов А.П. Математическое моделирование: идеи, методы, примеры. М.: Наука, 1997.
5. Форрестер Дж. Мировая динамика. М.: Наука, 1978.
6. Инструмент имитационного моделирования AnyLogic: обзор. URL: <http://www.anylogic.ru/overview> (дата обращения: 31.01.2014).
7. Minuteman Software: GPSS World. URL: <http://www.minutemansoftware.com> (дата обращения: 31.01.2014).
8. The world leader in systems thinking software: systems thinking for business (iThink). URL: <http://www.iseesystems.com/software/Business/IthinkSoftware.aspx> (дата обращения: 31.01.2014).
9. Simulink: simulation and model-based design. URL: <http://www.mathworks.com/products/simulink> (дата обращения: 31.01.2014).
10. Powersim Studio 10: build models with the system dynamics approach. URL: <http://www.powersim.com> (дата обращения: 31.01.2014).
11. Борщев А.В. Практическое агентное моделирование и его место в арсенале аналитика // Exponenta Pro. Математика в приложениях. 2004. № 3. С. 38–47.
12. Protégé is a free, open source ontology editor and knowledge-base framework. URL: <http://protege.stanford.edu/> (дата обращения: 31.01.2014).
13. Michel F., Ferber J., Gutknecht O. The Multiagent Development Kit. URL: <http://www.madkit.org/> (дата обращения: 31.01.2014).
14. Jess, the Rule Engine for the Java™ Platform. URL: <http://www.jessrules.com/> (дата обращения: 31.01.2014).

## ARCHITECTURE OF AGENTS DESIGN SUPPORT SYSTEM FOR COMPLEX SYSTEMS SIMULATION MODELS

*Pavlov A.I., Ph.D. (Engineering), Senior Researcher, asd@icc.ru;*  
*Stolbov A.B., Junior Researcher, stolboff@icc.ru*  
(Institute for System Dynamics and Control Theory Siberian Branch of RAS,  
Lermontov St. 134, Irkutsk, 664033, Russian Federation)

**Abstract.** The paper discusses problems of automating the creation of complex systems simulation models. The authors have chosen a multi-agent simulation approach as basic modeling tool. They propose software architecture for designing multi-agent models. The main tasks of a software system are: creating conceptual model; supporting the transformation process from conceptual model into agent model; decreasing qualification requirements in programming for domain experts; collaboration of research teams. The distinction of proposed software is conceptual model explicit representation in the concept-attribute relation form and applying expert system approach to transform designed ontology into an agent model. Researchers develop the structure and behavior of agents using declarative approach.

An agent declarative description is based on storing information about the agent structure and behavior separately from its implementation. The article proposes developing an agent describing mechanism allowing unified creating classes of agents that are significantly different from each other in structure and behavior. In this case a standard (unified) agent will have the following structure: states block containing agent characteristics; declarative block containing agent behavior strategy in the form of production rules; imperative block containing software implementation of calculation procedures used in the inference process. Modern tools for expert systems design are used for declarative block implementation.

**Keywords:** complex systems modeling, agent-based simulation, agent development support system, expert system, knowledge-based system, declarative programming style.

### References

1. Buslenko N.P. *Modelirovanie slozhnykh sistem* [Complex Systems Modeling]. Moscow, Nauka Publ., 1978.
2. Gurman V.I., Dykhta V.A., Kashina N.F. *Ekologo-ekonomicheskie sistemy: modeli, informatsiya, eksperiment* [Ecological-Economic Systems: Models, Information, Experiment]. Novosibirsk, Nauka Publ., 1987.
3. Mesarovich M., Mako D., Takakhara I. *Teoriya ierarkhicheskikh mnogourovnevnykh sistem* [The Theory of Hierarchical Multilevel systems]. Moscow, Mir Publ., 1973.
4. Samarskiy A.A., Mikhailov A.P. *Matematicheskoe modelirovanie: idei, metody, primery* [Principles of Mathematical Modeling. Ideas, Methods, Examples]. Moscow, Nauka Publ., 1997.
5. Forrester J. *World Dynamics*. 1971, Wright-Allen Press (Russ. ed.: Moscow, Nauka Publ., 1978).
6. *Instrument imitatsionnogo modelirovaniya AnyLogic: obzor* [AnyLogic Simulation Software Overview]. Available at: <http://www.anylogic.ru/overview> (accessed January 31, 2014).
7. *Minuteman Software: GPSS World*. Available at: <http://www.minutemansoftware.com> (accessed January 31, 2014).
8. *The World Leader in Systems Thinking Software: Systems Thinking for Business (iThink)*. Available at: <http://www.iseesystems.com/software/Business/IthinkSoftware.aspx> (accessed January 31, 2014).
9. *Simulink: Simulation and Model-Based Design*. Available at: <http://www.mathworks.com/products/simulink> (accessed January 31, 2014).
10. *Powersim Studio 10: Build Models with the System Dynamics Approach*. Available at: <http://www.powersim.com> (accessed January 31, 2014).
11. Borshchev A.V. Practical agent-based modeling and its place in the analytic's arsenal. *Exponenta Pro. Matematika v prilozheniyakh* [Exponenta Pro. Mathematics in Applications]. 2004, no. 3, pp. 38–47 (in Russ.).
12. *Protégé is a free, open source ontology editor and knowledge-base framework*. Available at: <http://protege.stanford.edu/> (accessed January 31, 2014).
13. Michel F., Ferber J., Gutknecht O. *The Multiagent Development Kit*. Available at: <http://www.madkit.org/> (accessed January 31, 2014).
14. *JESS, the Rule Engine for the Java™ Platform*. Available at: <http://www.jessrules.com/> (accessed January 31, 2014).