

А.Л. Куляница, О.Е. Фомичева

АРХИТЕКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ЭДА-МОДЕЛИ*

Рассмотрена архитектура многоагентной системы и системы программного обеспечения, которое реализует эту архитектуру. Целью является создание основы для разработки агентов социальной направленности. Поскольку «информация» является плохо формализуемым термином, предпочтение отдается понятию «семиотика», которое использует «знак» в качестве базового понятия. Информация, представленная в виде композиции знаков, анализируется на различных уровнях, в том числе синтаксиса, семантики, прагматики и социальном уровне. На основании различных свойств знаков, имеющихся на различных семиотических уровнях, предлагается новая агентная модель для представления информационных состояний агента и одновременно определения его концептуальной среды взаимодействий. Компоненты модели включают явное представление свойственных агентам институциональных ролей, где роль определяется как множество сервисов плюс множество политик. Ключевые слова: мультиагентное моделирование, организационная семиотика, распределенный искусственный интеллект.

Введение

Организационная семиотика – особая отрасль семиотики, формальное учение о знаках, связанная с анализом и моделированием организаций, как информационных систем. Основные характеристики информационных систем, такие как информация и взаимодействие, являются весьма сложными и плохо определяемыми понятиями, которые могут быть проанализированы посредством элементарных понятий, таких как семиотические «знаки». Бизнес-процессы тогда рассматриваются как процессы, связанные с созданием, взаимодействием и использованием знаков. Поскольку организационная деятельность является информационным процессом, основанным на понятии взаимодействующих агентов, предлагается модель, которая совмещает в себе социальный аспект в поведении организационных агентов и относительную самостоятельность, которую отдельные

агенты проявляют в реальных организациях. Предложенная модель основана на трех основных компонентах. Поскольку многоагентная логика недостаточно развита для достижения логической непротиворечивости и полноты организационных моделей, компьютерное моделирование дает возможность оценить адекватность организационных моделей. Кроме того, компьютерное моделирование позволяет получить представление об организационных явлениях и исследовать процессы и конфигурации, которые трудно или невозможно исследовать с помощью других методов.

ЭДА-модель

Социальная психология обеспечивает известную классификацию норм, разделяя их на перцепционные, оценочные, когнитивные и поведенческие нормы. Эти четыре типа норм связаны с четырьмя различными отношениями, соответственно:

* Исследования проводились в рамках гранта РФФИ 12-07-00797.

- онтологические – признание существования чего-либо;
- аксиологические – предпочтение или отвержение чего-либо в оценочном выражении;
- эпистемические – принятие степени доверия или недоверия;
- деонтические – способность действовать некоторым образом.

Разработанная агентная модель на основе этих отношений и связанных с ними норм, более подробно описана ниже:

- Перцепционные нормы, руководствуясь оценочными нормами, определяют какие знаки выбирает агент для восприятия. Когда знак воспринимается, прагматичная функция обновляет соответствующие компоненты агентной ЭДА модели.

- Когнитивные нормы определяют структуры сущностей, семантические значения и причинно-следственные связи, в том числе представления как о текущем состоянии, так и о будущем. Коэффициенты уверенности, обычно представленные в правилах, допускают существование исключений.

- Поведенческие нормы определяют, как агент предполагает действовать. Эти нормы в виде абстрактных планов описывают идеальное состояние, определяя тем самым, что агент должен делать. Деонтическая логика это модальная логика, изучающая формальные свойства нормативного поведения и состояний.

- Оценочные нормы необходимы для выбора агентом своих действий, основанных и на эпистемных и деонтических отношениях. Если рассматривать рационального агента, то он действует на основе максимизации некоторой функции, неявно определяемой как совокупность аксиологических отношений агента. Используя эту таксономию норм, и базируясь на предположении, что организационное поведение агента определяется оценкой деонтических норм данного эпистемического состояния агента, мы предлагаем модель намерений агента, которая декомпозируется на три компонента: эпистемологический, деонтические и аксиологический. Вместе эти компоненты включают все агентское информационное содержимое, в соответствии с семиотической структурой, изображенной на рис. 1, где показано, что информация представляет собой сложную концепцию, и требует различных точек зрения, чтобы быть полностью проанализированной.

Ψ является прагматической функцией фильтрации восприятия, в соответствии с перцептуальными и аксиологическими нормами агента, обновляет один или более компонентов модели.

Σ является аксиологической функцией, которая используется в двух случаях: чтобы решать, какие знаки воспринимать и выбирать, какие действия выполнять.

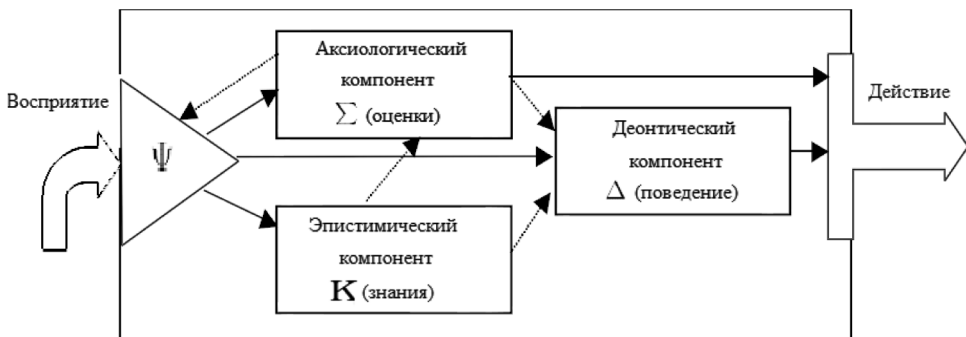


Рис. 1. ЭДА-агентная модель

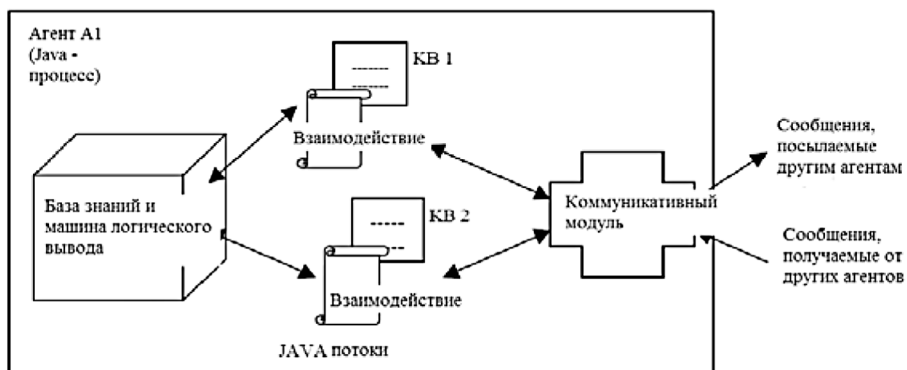


Рис. 2. Агентная архитектура программного обеспечения

К является компонентом, основанном на знаниях, где агент явно и неявно хранит уверенность, в виде потенциальных дедукций на основе логических рассуждений.

Δ представляет собой множество доступных явных и неявных планов, которые агент может выбрать для выполнения.

Для кодирования архитектуры интеллектуального агента мы должны использовать более абстрактные конструкции, чем те, что в обычных языках программирования, включая Java. Поскольку модель ЭДА подразумевает интеллектуальные отношения между агентами, представляется целесообразным использовать парадигму представления знаний искусственного интеллекта. Все компоненты модели требуют как процедурных, так и декларативных знаний. В то время как процедурные знания являются более эффективным, декларативное знание предоставляет большую гибкость и адаптируемость, таким образом, решено использовать гибридную парадигму, сочетая фреймовые системы с системами на основе правил, что позволяет отделять машину вывода декларативные знания от процедурных, кроме того обеспечивает возможность использования процедурных подпрограмм в случае необходимости.

Структура программного обеспечения

Структура, разработанная для поддержки ЭДА-агентов основывается на Java средах: одна из которых обеспечивает коммуникационную инфраструктуру, а другая предоставляет поддержку в виде продукционно-фрейм-овой базы знаний и машины вывода, способной осуществлять прямой и обратный логический вывод (рис. 2).

Коммуникационная инфраструктура

Коммуникационная инфраструктура обеспечивается коммуникативным модулем, который базируется на сетевой центрической вычислительной архитектуре, отражающей измененную вычислительную парадигму, функционирующую в зависимости от социальной среды. Ключевым аспектом этой парадигмы является то, что социальные системы не являются статичными: агенты и действия имеют динамику, где начало и окончание события определяется либо откликами агентов, либо автоматически с помощью социальных норм. Поскольку агенты автономны, они могут решать динамически формировать группы, и коммуникативный модуль способствует этому. Сеть динамически подключается, доступ к сервису клиентов осуществляются по

их Java интерфейсам, сервис может быть, например, аппаратным или программным агентом, каналом связи, или пользователем. Сервисы сети могут использоваться другими службами или клиентами. Сервисы доступны через подстановки сервисов, которые выступают в качестве посредников для поиска. Есть три основных процесса, определенные в спецификации коммуникативного модуля:

- Обнаружение: Устройства / Клиенты осуществляют поиск и получают ссылки для поиска сервисов.

- Объединение: Устройства регистрируют свои сервисы с поиском сервисов.

- Поиск: Клиенты определяют местонахождение и взаимодействуют с сервисами.

Новый сервис дает о себе знать многоадресной рассылкой «объявлений о присутствии» в то время как поисковые сервисы слушают объявления, и взаимодействуют с отправителями непосредственно, посылая им объект регистрации сервиса. Сервисы могут зарегистрироваться (посредством объединения) через сервис регистратора. После того как сервис зарегистрирован, клиенты могут получить к нему доступ. Сервисный объект может затем определить протокол доступа. Хотя сервисные протоколы являются приватными, максимальная гибкость достигается, когда сервис реализуется прозрачно и клиент просто должен вызывать методы, объявленные в Java-интерфейсе. При программировании коммуникативного модуля использовались некоторые новые распределенные управляющие механизмы и структуры данных, в том числе: распределенный лизинг, распределенные события и распределенные транзакции. Существует услуга, которая встраивается в рамки среды и использует эти программные конструкции, основана на концепции доски объявлений, разработанной в

области ИИ, способствует проектированию распределенных алгоритмов. Сервер содержит входы: типизированные группы объектов. Клиент может сравнить входы с шаблонами: входные объекты, в которых заполнены некоторые или все поля, значения которых должны точно совпадать. Возможно сделать запрос среды для подтверждения, что вход, соответствующий шаблону записан (используются распределенные события). Все операции, которые изменяют среду сделаны транзакционно безопасным способом в отношении этого пространства, с учетом того, что среда и клиенты могут использовать транзакции для мультиопераций в мульти-среде индивидуально. Входы, записанные в среде являются распределенными.

Управление взаимодействиями

Агенты являются Java программами, их взаимодействия – это Java потоки между агентами, использующие инфраструктуру коммуникативного модуля для организации каналов связи с другими агентами. Содержимое сообщений, проходящих по этим каналам связи кодируются на языке взаимодействия агентов (ACL). Независимыми разработчиками были предприняты шаги по стандартизации ACL – интер-языка, который используется гетерогенными агентами. Однако, поскольку ACL до сих пор не был широко принят в качестве стандарта, мы решили использовать наши собственные спецификации для содержания сообщений, основываясь на теории речевых воздействий [1, 2]. В результате был разработан интерпретатор сообщений. Это разработанная на Java среда, обеспечивающая работу как с фреймами, так и с правилами продукциями, а также машину вывода, которая может работать и в прямом, и в обратном направлении.

Нормативная структура: Роли, Сервисы и Политики

Организация нормативной структуры по существу определяется организационными ролями и их взаимосвязями. Множество внешних функций и процедур (методов), которыми обладает объект, составляют его интерфейс. Роль агента состоит из Java-объекта, имеющего интерфейс, отражающий функциональные возможности агента, плюс множество политик. Политика представляет собой набор правил, связанных с одним или несколькими компонентами ЭДА. Таким образом, каждый компонент ЭДА может получить набор правил, как результат загрузки роли одного агента. Политика ЭДА может представлять обязательство или авторизацию. Обязательства представлены как цели, используя определение цели, описанное выше. Авторизации представлены с использованием того же синтаксиса, что и цели, но в формате шаблона, где некоторые элементы являются масками (соответствует любой константе, которая может быть на их месте).

Мультиагентное взаимодействие

Были разработаны несколько моделей бизнес-процессов, в нотации «Язык–Действие–Перспектива» в том числе диаграммы потоков данных. На рис. 3 изображена типичная диаграмма взаимодействий. Шаг 3 на этой схеме представляет собой этап выполнения действий, требующих доступа агента к деонтической компоненте ЭДА, где сервис агента хранится в процедурном представлении. Разработка взаимодействий, в нашем случае, выполняется с помощью синтаксис-направленной схемы перевода и приписанных грамматик [4].

Пример организационного процесса в университете

Этот исследовательский пример касается организационного процесса в университете, связанного с приобретением книги, и инициированного по запросу лектора, как показано на рис. 4.

Неформальное описание. Процесс приобретения книги инициируется лектором, который лишь констатирует желание получить определенную кни-

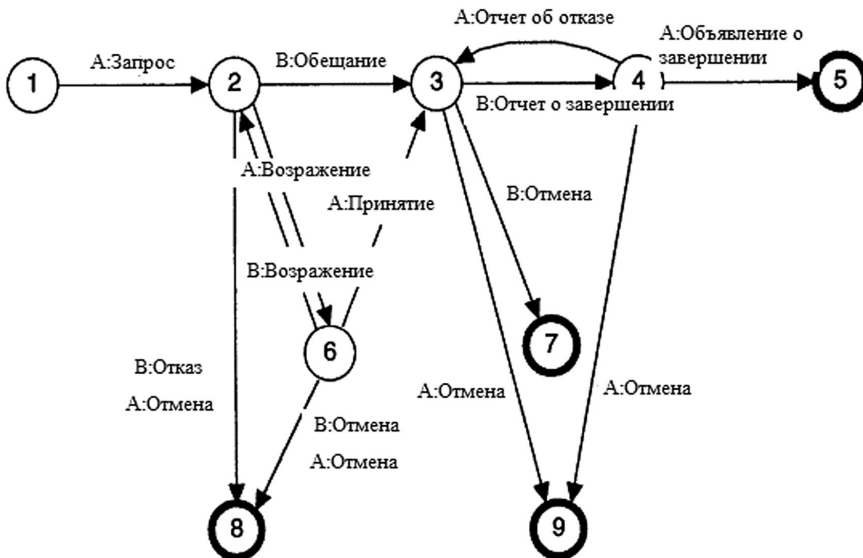


Рис. 3. Схема взаимодействий



Рис. 4. Упрощенная модель процесса заказа книги преподавателем

(assert (service (name book-Acquired) (client ICS-lecturer) (supplier Lib)))

Рис. 5

гу. Этот запрос передается человеку, который выполняет роль посредника между подразделением, к которому относится лектор и библиотекой. Этот посредник должен запросить, считает ли руководитель подразделения приобретение книги необходимым и имеется ли достаточно средств на ее приобретение. Если ответ положительный, то библиотека будет приобретать книгу и проинформирует лектора, как только книга станет доступной.

Онтология предметной области. Каждый организационный агент может быть человеком или искусственным агентом. Однако, искусственные агенты действуют в организационной сети от имени пользователей, которые в конечном счете ответственны за этих агентов. Для достижения общего понимания организационных процессов, независимо от того, какой агент выполняет какую роль, необходимо создать соответствующую онтологию предметной области. Это можно сделать с помощью онтологической схемы.

Роли. Есть четыре роли, определенные в примере. Каждая роль соответствует набору сервисов плюс набор политик, что представляется в виде

```
IF (request (from ?agent)(service book-acquired)(details ?book1)) and
  (lecturer (name ?agent) (type ICS-lecturer))
THEN
  (obligation (agent Rep) (stit book-acquired ?book1) (time now now+300)(utility 0)(cost 200)(sanction EST)))
```

Рис. 6

```
(assert (authorized (agent Rep) (stit book-Acquired :title "*" :author "*" :cost "<1000") (time *) (utility *) (cost *) (sanction *)))
```

Рис. 7

фактов и правил в базе знаний. Чтобы проиллюстрировать процесс построения роли, сосредоточим внимание на агенте представителе подразделения.

Услуги. Услуги рекламируются с помощью уверенностей, заявленных в базе знаний. Например, при использовании базы знаний (см. рис. 5).

Это означает, что этот агент выполняет сервис для достижения цели приобретения книги «book-Acquired», но только если клиент ICS-преподаватель; услуга может быть делегирована поставщиком агента библиотеки «Lib». Сервис может быть как атомарной задачей, представленной процедурно с помощью метода, так это может быть совокупность задач, представленных декларативно.

Политики. Политика это множество обязательств и авторизаций. Обязательства роли это по умолчанию нормы, введенные организацией, к которой принадлежит агент. В нашем примере, роль агента это «Rep», организация это «EST».

Пример в модуле «База знаний и машина логического вывода» (см. рис. 6).

Это означает, что если агент Rep получает запрос от ICS-лектора, то

он должен принять на себя обязательство следить за тем, что книга приобретена, хотя фактически удовлетворение обязательства может не возникнуть, например, если агент Кер не получит необходимое разрешение от начальника подразделения. В этом случае второе обязательство вступает в силу, что заставляет нарушить первое обязательство. В этом случае величина санкции будет равна нулю, потому что нарушение было «принято» организацией, которая управляет применением санкций. Каждая цель агента имеет неявное предусловие, которое должно быть удовлетворено до того, как цель будет поставлена в план исполнения агента: что агент авторизован на достижение этой цели. Авторизация представляет собой целевой шаблон, в котором некоторые пробелы заполняются с помощью масок, таких как в примере (см. рис. 7).

Взаимодействия. Взаимодействия на рис. 3 представлены двумя Java потоками: один связан с клиентским агентом (инициатором), другой подключается к сервисному провайдеру. Структура коммуникативного модуля обеспечивает прямое взаимодействие инфраструктуры на основе обмена сообщениями связи. Также имеется сервис доски объявлений, связанный с поисковым сервером, который используется в качестве вспомогательного агента, обеспечивающего механизмы для клиентов в соответствии с их запросами на обслуживание к зарегистрированным поставщикам услуг. Сеансы связи разработаны в соответствии с правилами переходов, определенными конечным автоматом, который декларативно описан в базе знаний множеством правил. Концептуальным эквивалентом конечного автомата в этом методе является динамическая модификация правил перехода, что делает доступным он-лайн машинное обучение.

Заключение

ЭДА модель основана на организационном семиотическом подходе, где спецификой являются нормативные знания и норма-основанные отношения. Основные компоненты модели (эпистемический, деонтический и аксиологический) отражают социально-психологическую классификацию норм, следовательно, обеспечивают принципиальную норма-основанную структуру для агентов внутренней архитектуры, которая также ориентирована на норма-ориентированные социальные взаимодействия в организациях. Архитектура ЭДА объединяет также ряд важных идей, взятых из области распределенного искусственного интеллекта и деонтической логики. Особенно важным для организационного моделирования является понятие «обязательство». Многие понятия обязательства были определены, как в распределенном искусственном интеллекте, так и в деонтической логике. Тем не менее, мы согласны с [3], где обязательства определяются как «договоренность между вовлеченными агентами о достижении и удержании заданной цели в течение определенного времени». Хотя мы формально не определили наше представление об обязательствах, мы представляем обязательства в терминах целей, проявляющихся как прагматический результат социального взаимодействия. Мы считаем, что многоагентные обязательства могут быть смоделированы как связанные множества деонтических утверждений, распределенных по связанным агентам, основанных на понятии единых целей, как предлагается в деонтической составляющей нашей модели. Аксиологический компонент, как представляется, является необходимой частью любого интеллектуального агента, для установления множества предпочтительных убеждений агентов и приоритетов для противоречивых целей. В многопользовательской агентной

среде взаимное обновление ЭДА-агентов модели является существенным как результат восприятия событий, таких как обмен сообщениями. Однако спецификация обновления ЭДА с использованием прагматической функции является предметом текущих исследований. Связанные с этим направлением исследования, которые проводятся в настоящее время, включают в себя

моделирование программного обеспечения моделей ЭДА. Решаются некоторые вопросы разработки программного обеспечения, связанные с осуществлением гетерогенной реализации многоагентных систем, где аспекты взаимодействия являются ключевыми проблемами, требующими прагматической интерпретации сообщений между агентами.

СПИСОК ЛИТЕРАТУРЫ

1. Austin J. *How To Do Things with Words*. Oxford University Press: Oxford, England, 1962.

2. Searle J. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press: Cambridge, England, 1969.

3. Hägg S. Commitment in Agent Cooperation, Applied to Agent-Based Simulation / Proceedings of the 5th International Conference on Intelligent Autonomous Systems (IAS-5), Sapporo, Japan, 1998. **ИАС**

КОРОТКО ОБ АВТОРАХ

Куляница Андрей Леонидович – доктор технических наук, профессор,
Фомичева Ольга Евгеньевна – кандидат технических наук, доцент,
e-mail: olga-e-fom@yandex.ru,
ИТАСУ НИТУ «МИСиС».

UDC 004.942, 004.853,721.021.23, 519.876.5

EDA MODEL SOFTWARE ARCHITECTURE

Kuljanica A.L.¹, Doctor of Technical Sciences, Professor,

Fomicheva O.E.¹, Candidate of Technical Sciences, Assistant Professor, e-mail: olga-e-fom@yandex.ru,

¹ Institute of Information Technologies and Automated Control Systems,

National University of Science and Technology «MISiS», 119049, Moscow, Russia

This article discusses the architecture of multi-agent systems and software that implements this architecture. The aim is to establish a framework for the development of agents of social orientation. As the "information" is difficult to formalize the term, preference is given to the concept of "semiotics", which uses a "sign" as a basic concept. The information provided in the form of a composition of characters is analyzed at various levels, including the syntax, semantics, pragmatics and social level. Based on the various properties of signs available in different semiotic levels, we propose a new agent-based model for representing information agent states and at the same time defining its conceptual environment interactions. Components of the model include an explicit representation of agents inherent institutional roles, where the role is defined as a set of services plus a variety of policies.

Key words: multi-agent modeling, organizational semiotics, distributed artificial intelligence, epistemic component, deontic component, axiological component, ontology.

ACKNOWLEDGEMENTS

The work was supported by the Russian Foundation for Basic Research, Project No. 12-07-00797.

REFERENCES

1. Austin J. *How To Do Things with Words*. Oxford University Press: Oxford, England, 1962.

2. Searle J. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press: Cambridge, England, 1969.

3. Hägg S. Commitment in Agent Cooperation, Applied to Agent-Based Simulation *Proceedings of the 5th International Conference on Intelligent Autonomous Systems (IAS-5)*, Sapporo, Japan, 1998.