

## Лекция 14

### Тема: Мультиагентные системы

Система, в которой несколько агентов могут общаться, передавать друг другу некоторую информацию, взаимодействовать между собой и решать поставленную задачу называется мультиагентной (МАС). В МАС задачи (или подзадачи) распределены между агентами, каждый из которых рассматривается как *член группы* или организации. Распределение задач предполагает назначение ролей каждому из членов группы, *определение* меры его "ответственности" и требований к "опыту".

Многоагентные системы зародились на пересечении теории систем и распределенного искусственного интеллекта. С одной стороны, речь идет об открытых, активных, развивающихся системах, в которых главное внимание уделяется процессам взаимодействия агентов как причинам возникновения системы с новыми качествами. С другой стороны, достаточно часто МАС строятся как *объединение* отдельных интеллектуальных систем, основанных на знаниях. МАС обычно состоит из следующих основных *компонент*:

- множество организационных единиц, в котором выделяются: подмножество агентов, манипулирующих подмножеством объектов;
- множество задач;
- среда, т. е. некоторое пространство, в котором существуют агенты и объекты;
- множество отношений между агентами;
- множество действий агентов (например, операций над объектами).

Основой формы организации взаимодействия между агентами, характеризующаяся объединением их усилий для достижения совместной цели при одновременном разделении между ними функций, ролей и обязанностей является *кооперация*. В общем случае это понятие можно определить формулой: *кооперация* = сотрудничество + координация действий + разрешение конфликтов. Под координацией обычно понимается управление зависимостями между действиями. Коммуникация между искусственными агентами зависит от выбранного протокола, который представляет собой множество правил, определяющих, как синтезировать значимые и правильные сообщения. Фундаментальными особенностями группы, составленной из агентов, сотрудничающих для достижения общей цели, являются социальная структура и распределение ролей между агентами.

Основой архитектуры агента является *контекст*, или серверная среда, в котором он исполняется. Каждый *агент* имеет постоянный *идентификатор* – имя. В серверной среде может исполняться не только исходный *агент*, но и его копия. Агенты способны самостоятельно создавать свои копии, рассылая их *по* различным серверам для исполнения работы. *По* прибытии агента на следующий *сервер* его код и данные переносятся в новый *контекст* и стираются на предыдущем местонахождении. В новом контексте *агент* может делать все, что там не запрещено. *По* окончании работы в контексте *агент* может переслать себя в другой *контекст* или *по* исходящему адресу отправителя. Агенты способны также выключаться ("умирать") сами или *по* команде сервера, который переносит их после этого из контекста в *место*, предназначенное для хранения.

На рис.1\_показана укрупненная структура типичного агента. Входами являются *внутренние параметры* агента и данные о состоянии среды. Выходы – параметры, воздействующие на среду и информирующие пользователя (или программу, выполняющую роль менеджера в системе) о состоянии среды и принятых решениях. Решатель – процедура *принятия решений*. Решатель может быть достаточно простым алгоритмом или элементом системы искусственного интеллекта.



Рис. 1. Укрупнённая структура агента



Рис. 2. Архитектура ядра мультиагентной системы

В архитектуре МАС основную часть составляет предметно-независимое *ядро*, в составе которого выделяются следующие базовые компоненты (рис. 2\_):

- служба прямого доступа обеспечивает непосредственный доступ к атрибутам агентов;
- служба сообщений отвечает за передачу сообщений между самим агентами, а также между агентами и дополнительными системами ядра;
- библиотека классов агентов (часть базы знаний) содержит информацию о классификации агентов в данной МАС.
- сообщество агентов – серверное "место", где размещаются агенты; этот блок, кроме жизнедеятельности агентов, обеспечивает еще функции по загрузке/записи агентов и их свойств и за оптимизацию работы агентов с ресурсами.
- онтология – предметная база знаний, содержащая конкретные знания об объектах и среде функционирования, представляемые в виде соответствующей семантической сети.

Общая методология восходящего эволюционного проектирования МАС может быть представлена цепочкой: <среда – функции МАС – роли агентов – отношения между агентами – базовые структуры МАС – модификации>, и включает следующие этапы:

- формулирование назначения (цели разработки) МАС;
- определение основных и вспомогательных функций агентов в МАС;
- уточнение состава агентов и распределение функций между агентами, выбор архитектуры агентов;
- выделение базовых взаимосвязей (отношений) между агентами в МАС ;
- определение возможных действий (операций) агентов;
- анализ реальных текущих или предполагаемых изменений внешней среды.

При проектировании организацию агентов можно рассматривать как набор ролей, находящихся между собой в определенном отношении, и взаимодействующих друг с другом. Таким образом, методология восходящего проектирования *MAC* требует предварительного задания исходных функций (ролей агентов), определения круга их обязательств *по* отношению друг к другу, формирования исходных и развивающихся структур на основе выделенных функций и исследования адекватности этих структур характеру решаемых задач в выделенных проблемных областях

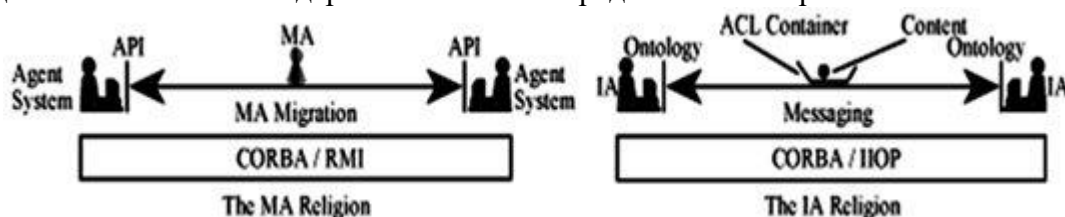
Главная идея нисходящего проектирования состоит в определении общих социальных характеристик *MAC* по некоторому набору критериев, построении базовых типов их организаций с последующим определением требований к архитектуре агентов. Когда речь идет о "выращивании" искусственных социальных систем и сообществ, на первый план выдвигается нисходящий подход к организационному проектированию.

В рассмотренных ниже примерах, безусловно, наиболее подходящим является проектирование на основе восходящего подхода.

### ***Современные международные стандарты создания агентов и платформы MAC***

Существует несколько международных подходов к созданию мультиагентных систем, наиболее известные из них – это *OMG MASIF*, созданный *Object Management Group*, в основе которого лежит понятие мобильный агент; спецификации *FIPA* (*Foundations for Intelligent Physical Agents*), основанные на предположении об интеллектуальности агента, а также стандарты, разработанные исследовательским подразделением Пентагона – Агентством Передовых Оборонных Научных Исследований (*Defense Advanced Research Projects Agency – DARPA*), в частности *Control of Agent Based Systems*.

Относительно мобильности и интеллектуальности агентов, большинство специалистов сходятся на том, что мобильность – центральная характеристика агента, интеллектуальность – желаемая, но не всегда строго требуемая. Различия подходов к определению агента в стандарте *FIPA* и *OMG* представлены на рис.3



**Рис. 3.** Сравнение мобильных и интеллектуальных агентов

Деятельность *FIPA* заключается в совместном исследовании и разработке членами организации международных согласованных спецификаций, которые позволят максимизировать взаимодействие между агентными приложениями, услугами и оборудованием. Членами *FIPA* являются такие высокотехнологичные компании как Alcatel, Boeing, British Telecom, Deutsche Telekom, France Telecom, Fujitsu, Hitachi, HP, IBM, Fujitsu, Hewlett Packard, IBM, Intel, Lucent, NEC, NHK, NTT, Nortel, Siemens, SUN, Telia, Toshiba, различные университеты, государственные организации.

Спецификации *FIPA* ориентируются на обеспечение возможности взаимодействия интеллектуальных агентов через стандартизованную коммуникацию агентов и языки контента. Наряду с общими основами коммуникации *FIPA* специализируется также на протоколах онтологии и переговоров для поддержки взаимодействия в конкретных прикладных сферах (*транспортная поддержка, производство, мультимедиа, поддержка сетевого взаимодействия*).

Стандарт *OMG MASIF* нацелен на создание условий для миграции мобильных агентов между мультиагентными системами посредством стандартизованных интерфейсов *CORBA IDL*.

Организация *DARPA* инициировала работу *по* распределению знаний (*Knowledge Sharing Effort*), в результате которой языки программирования агентов были разделены на синтакс (*syntax*), семантику (*semantics*) и прагматику (*pragmatics*):

- KIF – Knowledge Interchange Format (*syntax*);
- Ontolingua – a language for defining sharable ontologies (*semantics*);
- KQML (Knowledge Query and Manipulation Language) – a high-level interaction language (*pragmatics*).

Важным элементом при создании мультиагентных систем является язык коммуникации агентов – *Agent Communication Language*, который определяет типы сообщений, которыми могут обмениваться агенты. В рамках парадигмы коммуникации между агентами, *кооперация* между ними достигается за счет *ACL*, языка контента и онтологии, которые определяют набор базовых концепций, используемых в сообщениях кооперации. *Онтология* здесь выступает синонимом понятия *API* (*Application Programming Interface*), т.е. она определяет конкретный *интерфейс* интеллектуальных агентов.

На техническом уровне коммуникация между агентами происходит за счет передачи сообщений используя какой-либо транспортный протокол нижнего уровня (*SMTP, TCP/IP, HTTP, IIOP*). Альтернативами к использованию *ACL* является ряд других языков, таких как языки БД (*SQL*), *Distributed object systems* (*CORBA* и др.), *Service languages* (e-speak от Hewlett Packard, BizTalk от Microsoft и др.) и *Web languages* (*XML, RDF, DAML*).

Еще одной альтернативой *ACL* является *CORBA ORB*, разработанный уже упоминавшийся *Object Management Group*. Вся функциональность, предоставляемая *CORBA* доступна и на языке *JAVA*, путем комбинации *Java RMI, Java RMI servers, Jini, Java event servers* и других.

В настоящее время языки коммуникации агентов продолжают эволюционировать. Поскольку совместимость – определяющая характеристика агентов, при разработке *MAS* – очень важна именно стандартизированная коммуникативность. Основными объектами для стандартизации являются: *архитектура* агента, языки взаимодействия агентов, протоколы взаимодействия агентов, знания агентов, языки программирования агентов.

Как отмечают эксперты в области разработки агентов, для последующей эволюции технологий создания агентов необходимы следующие действия:

- развитие семантики языков коммуникации агентов (*ACL*) (общих языков контента и онтологии; языков для описания действий агентов, намерений и стремлений);
- развитие онтологии агентов (разделяемые онтологии для свойств агентов и их поведения);
- улучшение использования метаданных (абстрактное и совмещаемое со многими языками контента);
- декларативные и ясные протоколы (языки для определения протоколов высокого уровня, базирующиеся на более примитивных);
- практический обмен знаниями между агентами (социальные механизмы для обмена информацией и знаниями, рассмотрение обмена знаниями как мобильный код);
- развитие схем и методов для контроля за системами агентов (искусственные рынки, естественный отбор и т. п.).

Агентные платформы представляют собой один из способов построения распределенных систем и позволяют описать и предоставить *доступ* всех приложений, работающих на агентной платформе к необходимым им сервисам. Кроме того, в функции агентной платформы входит распределение агентов, *аудит* их функционирования и управление.

#### **Платформы для разработки MAS**

Перед тем, как приступить к созданию приложений на *Android*, необходимо выбрать подходящий *инструментарий* разработки и установить соответствующий *Android SDK*.

Наиболее популярные *средства разработки MAS* следующие:

- JADE (Java Agent Development Framework) — широко используемая программная среда для создания мультиагентных систем и приложений, поддерживающая FIPA-стандарты для интеллектуальных агентов. Включает в себя среду выполнения агентов (агенты регистрируются и работают под управлением среды), библиотеку классов, которые используются для разработки агентных систем, набор графических утилит для администрирования и наблюдения за жизнедеятельностью активных агентов. Программная среда JADE подключается к любому проекту на языке Java. Агенты JADE могут быть совершенно разными — от простых, только реагирующих, до сложных — ментальных.

- JACK Intelligent Agents — Java платформа для создания мультиагентных систем. Так же как и JADE, расширяет Java своими классами. JACK одна из немногих платформ, где используется модель логики агентов, основанная на убеждениях-желаниях-намерениях (Belief–desire–intention software model – BDI), и встроенные формально-логические средства планирования работы агентов.

- MadKIT — модульная и масштабируемая мультиагентная платформа, написанная на Java. Поддерживает агентов на разных языках: Java, Python, Jess, Scheme, BeanSchell. Красиво визуализирует и позволяет управлять этими агентами.

- AgentBuilder — большой коммерческий продукт, выпускаемый так же и в Academic Edition. Агенты достаточно интеллектуальны, и общаются на языке KQML (Knowledge Query and Manipulation Language) и обладают ментальной моделью. Платформа является Java-ориентированной.

- Cougaar (Cognitive Agent Architecture) — также Java-ориентированная платформа для построения распределенных мультиагентных систем. Включает не только исполняющую систему (run-time engine), но и некоторые средства для визуализации, управления данными и др.

- NetLogo — кроссплатформенное программируемое окружение для программирования мультиагентных систем.

- VisualBots — бесплатный мультиагентный симулятор в Microsoft Excel с Visual Basic синтаксисом.

- MASON — Java библиотека для моделирования мультиагентных систем.

- REPAST — набор инструментов для создания систем, основанных на агентах.

- CogniTAO — C++ платформа разработки автономных мультиагентных систем, ориентированная на реальных роботов и виртуальных существ (CGF).

### **Области применения**

На сегодняшний день мультиагентные системы используются для разработки широкого спектра информационных и промышленных систем. В промышленности МАС наиболее распространены применительно к решению задач автоматизации управления сложными системами, для сбора и обработки информации, в играх. Мультиагентные технологии применимы в управлении мобильными ресурсами, а также в таких сферах, как проектирование объектов, промышленное производство, финансовое планирование и анализ рисков, распознавание образов, извлечение знаний из данных, понимание текста и решение других сложных проблем.

Так, например, IBM использует агентов для производства полупроводниковых микросхем, датская судостроительная компания – для заварки отверстий в кораблях, а в Японии система на базе агентов выполняет функции интерфейса оператора сверхскоростного поезда

МАС могут применяться как для конструирования и моделирования гибких производственных систем, так и для управления реальными системами производства (логистика), продажи продукции различного назначения (e-коммерции), интеграции и управления знаниями и научной работы. Большое значение в мультиагентном подходе имеет социальный аспект решения современных задач как его концептуальная основа.. Такие системы должны постоянно "жить" на сервере предприятия и непрерывно участвовать в решении задач, а не быть запускаемыми от случая к случаю, а для этого –

обеспечивать пользователю возможность введения новых данных и компонентов. Наконец, такие системы должны накапливать информацию, извлекать из нее новые знания и в зависимости от этого изменять свое поведение с течением времени.

В настоящее время интеллектуальные агенты применяются в следующих областях бизнеса:

- управление распределенными или сетевыми предприятиями;
- сложная и многофункциональная логистика;
- виртуальные организации и Интернет-порталы по продаже продуктов и услуг;
- управление учебным процессом в системах дистанционного обучения;
- компании с развитыми дистрибьюторскими и транспортными сетями (например, в Procter&Gamble);
- управление каналами распределения;
- моделирование предпочтений пользователей.

Для крупных компаний преимущества мультиагентного подхода очевидны. Среди них можно отметить: сокращение сроков решения проблем, уменьшение объема передаваемых данных за счет передачи другим агентам высокоуровневых частичных решений; сокращение сроков согласования условий и формирования заказов.

Для распределенных компаний преимущества в первую очередь заключаются в возможности оптимального обеспечения продукцией, облегчении контроля удаленных подразделений и структур и взаимодействия с ними.

Для компаний с широким и быстро меняющимся ассортиментом – возможность гибко реагировать на изменения в предпочтениях клиентов и просчитывать периоды изменения. Для компаний оказывающих услуги – накопление опыта взаимодействия и решения проблем не только "в головах" сотрудников, но и в МАС.

Помимо этого, агенты могут быть уполномоченными представителями пользователя при общении с другими пользователями или их агентами, при решении порученных им задач.

Для решения задач автоматизации управления ресурсами предприятий в реальном времени в последнее время разрабатывается большое количество интеллектуальных программных систем нового поколения, построенных на основе мультиагентных технологий, которые позволяют автоматизировать полный цикл управления мобильными ресурсами в реальном времени, включая:

- оперативную реакцию на важные события;
- динамическое планирование и адаптивное перепланирование заказов/ресурсов;
- взаимодействие с клиентами, менеджерами и исполнителями для согласования принимаемых решений через Интернет или сотовый телефон;
- мониторинг исполнения построенных планов и бизнес-процессов заказчика;
- перепланирование расписаний в случае рассогласования между планом и фактом.

В таких системах агенты способны взаимодействовать друг с другом путем переговоров и демонстрировать коллективный *интеллект*, возникающий в системе в форме спонтанных цепочек согласованных изменений планов агентов.

Таким образом, в разработке программных систем используются фундаментальные принципы самоорганизации и эволюции, присущие живым системам, например, колонии муравьев или рою пчел, отличающихся способностью решать сложные задачи в реальном времени, открытостью к изменениям, высокой эффективностью, надежностью и живучестью.

Обычно такие системы легко интегрируются с существующими коммуникаторами, учетно-контрольными системами предприятия, электронными картами, средствами *GPS* навигации, *RFID*-чипами и т.п.

**МАС для распределения заказов такси.** Для одной из крупнейших в мире компаний корпоративного такси Addison Lee (Лондон) была разработана система, которая позволила распределять и планировать примерно 13 тысяч заказов в день при наличии нескольких

тысяч собственных машин (из них до 800 постоянно на линии), оснащенных средствами GPS-навигации. При появлении нового заказа система автоматически находит наилучшую машину, получая сведения о координатах ближайших машин на электронной карте Лондона, и предварительно бронирует заказ. Если эта машина была уже занята, то начинается цепочка переговоров, направленная на разрешении возникшего конфликта и достижение компромисса, что позволит перебросить старый заказ на другую машину, если это выгодно для всех. Но и после этого работа системы с новым заказом не останавливается.

В среднем на подачу машины требуется около 15 минут, при этом примерно половину этого времени система продолжает непрерывно искать возможности для улучшения перевозки с учетом поступающих заказов и появляющихся новых ресурсов и не принимает окончательного решения до момента, когда необходимо отправлять машину с учетом времени пути проезда до заказа. Когда уже пора отправлять автомобиль на заказ, система принимает окончательное решение, посылает водителю сообщение о параметрах заказа и ждет подтверждения о приеме заказа.

Внедрение такой системы дало возможность роста компании за счет повышения управляемости. Существенно увеличилась эффективность автопарка (10-15%) благодаря оптимальному распределению заказов за счет минимизации "холостого" пробега и времени простоя автомобилей. Также сократилось количество опозданий и время обслуживания заявок.

**МАС для управления группами интеллектуальных роботов.** Хорошим примером интеллектуальных агентов служат роботы. Роботы могут иметь широкий ассортимент искусственных органов чувств (сенсорные датчики) и искусственных эффекторов (манипуляторы, педипуляторы). Их мобильность достигается благодаря колесным, гусеничным, шагающим и прочим системам перемещения. *Активность* и автономность роботов тесно связаны с наличием средств целеполагания и планирования действий, систем поддержки решения задач, а *интеллектуализация*, помимо обладания системой обработки знаний, предполагает развитые средства коммуникации различных уровней, вплоть до средств естественного языкового общения.

Неотъемлемым атрибутом интеллектуальных роботов является наличие специальной подсистемы планирования, составляющей программу действий робота в реальных условиях окружающей среды, которые определяются рецепторами робота. Для планирования деятельности робота должен иметь знания о свойствах окружающей среды и путях достижения целей в этой среде.

**МАС для управления безопасностью автомобиля.** Примером инновационного продукта, обеспечивающего комплексную *безопасность* автомобиля на дороге может служить информационно-диагностическая подсистема "Автоматизированная помощь на дороге" (АПД) для on-line обслуживания водителей.

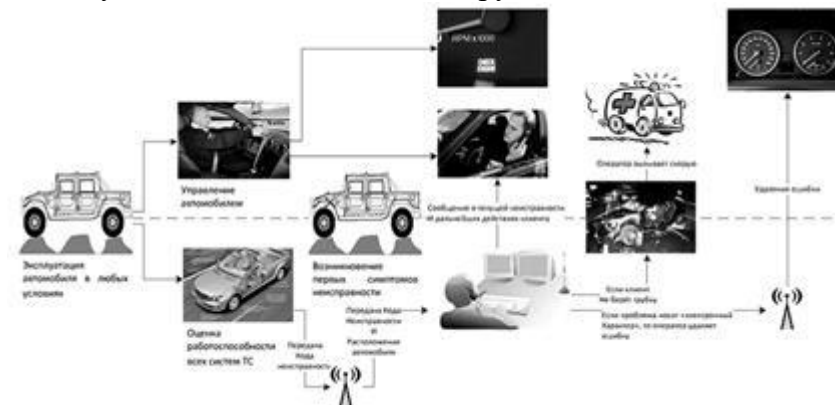
Подсистема предназначена для постоянного текущего мониторинга устройств и механизмов автомобиля клиента и базируется на современных достижениях в области сбора, передачи, и автоматизированного анализа сигналов, обработки сигналов в режиме on-line с использованием современных датчиков и контроллеров, а также контроля всей системы на базе мультиагентного подхода. Пользователем может стать любой человек, который пожелает установить систему АПД при покупке нового автомобиля распространенных марок Ford, Opel, Chevrolet, KIA, BMW и др.

Главным назначением АПД является повышение безопасности клиента. Во время движения автомобиля могут возникать нештатные ситуации, связанные с состоянием агрегатов автомобиля, дорожной обстановкой, состоянием покрытия, погодой и т.д. При этом водитель может не подозревать о надвигающейся аварийной ситуации, связанной с некорректной работой двигателя, тормозной системы, рулевого управления. Система АПД обязана минимизировать последствия критических поломок, так как она призвана при первых симптомах неисправности, отправить сведения о них в дилерский центр, оператор которого свяжется с клиентом и сообщит о возможных последствиях и путях их

предотвращения. Рассмотрим вариант, когда автомобиль не оборудован системой АПД (рис.4.) Клиент эксплуатирует автомобиль определённое количество времени, проводит регламентированные работы (ТО) в официальном автосервисе. В случае инцидента на дороге (отказ двигателя, съезд с полотна и переворот вследствие прокола колеса, наезд на препятствие, столкновение с другим транспортным средством) водитель обязан вызвать сотрудников ГИБДД, страхового агента и т.д. Но если водитель серьёзно пострадал, то он не может определить своё состояние, местоположение, вызвать себе скорую помощь и соответствующие службы. И если ДТП произошло на просёлочной дороге, то шансы быстро получить необходимую помощь резко уменьшаются.



**Рис. 4.** Эксплуатация автомобиля, не оборудованного системой АПД



**Рис. 5.** Эксплуатация автомобиля, оборудованного системой АПД

Если же автомобиль оснащен системой оказания комплексной услуги АПД (рис. 5), то при возникновении малейшей неисправности (например, нештатная вибрация кузова, рулевого управления, утечка масла или бензина, критический износ тормозных колодок и пр.) система отображает возникшую коллизию на панели приборов. Бортовой компьютер отправляет код неисправности оператору в дилерский центр или станцию технического обслуживания. Оператор немедленно связывается с дежурным мастером-консультантом, а тот, в свою очередь, удаленно тестирует неисправность, связывается с клиентом и даёт рекомендации по поводу дальнейшего эксплуатации автомобиля. Если клиент



попадает в ДТП, то *компьютер* автоматически отсылает информацию о том, что сработали ремни и подушки безопасности. Оператор сразу же пытается связаться с клиентом *по* сотовому телефону. Если клиент не снимает трубку, то оператор вызывает скорую помощь на *место*, которое было определено с помощью *GPS*-передатчика на автомобиле.

***МАС для управления группой БПЛА.*** Еще одним перспективным направлением применения *МАС* являются системы беспилотных летательных аппаратов. В современном мире "беспилотные летательные аппараты" (англ. *Unmanned Aerial Vehicles*, далее БПЛА) приобретают все большую популярность в качестве легких и недорогих инструментов для исследований территорий, разведки и воздушных съемок. В основной массе современная беспилотная техника не обладает автономностью (обычно дистанционно управляется оператором со специального пульта). Есть успешные разработки одиночных БПЛА, которые под управлением автопилота способны в автономном режиме облететь территорию *по* заданному маршруту, собирая ту или иную информацию или выполняя другие задания. *Анализ* большинства задач решаемых с использованием одиночных аппаратов показывает, что они могут более эффективно решаться группой, так как у группы эффективно взаимодействующих легких и маленьких самолетов появляются дополнительные полезные свойства. В следующей главе будет рассмотрен один из возможных вариантов реализации *МАС* для управления группой БПЛА.