

УДК 004.89

**ИНТЕЛЛЕКТУАЛЬНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ:
ИСТОРИЯ, КОНЦЕПЦИИ И МЕТОДИКА ПРОЕКТИРОВАНИЯ****А.И. ДУНЧЕНКО****(Представлено: Д.В. ПЯТКИН)**

Представлена история возникновения и развития интеллектуальных информационных систем. Показаны их разновидности, классы решаемых ими задач. Раскрываются особенности архитектуры, описаны основные принципы проектирования.

В действительности, термин *интеллектуальная информационная система* (ИИС) [1] может быть применен к довольно широкому кругу компьютерных программ, объединенных общей характерной особенностью – использованием базы знаний как главного критерия в вычислениях при решении различных типов сложных задач. Это означает, что вместо жесткого кодирования всех возможных ситуаций и их исходов при выполнении программы с помощью конструкций *if-then-else*, в ИИС применяется другой подход: они формализуются и с помощью какой-либо декларативной нотации представляются в виде фактов и правил. По сути ИИС представляет собой гибрид интеллектуальной и информационной систем [2].

Первыми ИИС были в основном базирующиеся на правилах *экспертные системы* (ЭС) [3]. Иногда по ошибке оба этих термина применяются как взаимозаменяемые. Но на самом деле между ними существует значительная семантическая разница. Интеллектуальная информационная система – это общий принцип построения архитектуры программных систем. Если выражаться в терминах объектно-ориентированного программирования, то ИИС следует рассматривать как абстрактный базовый класс. С этой точки зрения экспертная система будет классом-потомком, специфицирующим применяемый подход для решения уже более узкого класса проблем.

Каждая ЭС дублирует способности к принятию решений реального человека – эксперта в какой-либо конкретной области (отсюда и название таких систем) [4]. Другими словами, ЭС может самостоятельно выполнять анализ предложенных фактов и параметров и на их основании выдавать высокоточные заключения (давая, при необходимости, пояснения к своим выводам). Таким образом, одна ЭС способна заменить целую команду экспертов, обычно необходимую для выполнения этой задачи. Использование таких программных систем предоставляет (в теории) широкие возможности по оптимизации и автоматизации бизнес-процессов (проводить реинжиниринг).

История экспертных систем берет свое начало в 70-х годах прошлого столетия, когда они были предложены руководителем проекта Stanford Heuristic Programming Project Эдвардом Фейгенбаумом. В то время основной платформой для их реализации были Lisp-машины, выпускаемые компаниями Symbolics, Xerox и Texas Instruments. Экспертные системы стали одними из первых технически и коммерчески успешных форм *программного обеспечения* (ПО), использующего *искусственный интеллект* (ИИ) и спроектированного с учетом необходимости решения сложных задач путем оперирования знаниями.

В 80-х годах вместе с появлением на рынке IBM PC, всеобщим растущим интересом к более новой *клиент-серверной* парадигме организации вычислений, с увеличением финансирования для исследований в университетах, последние начали предлагать специализированные обучающие курсы, в результате чего ЭС получили необходимый толчок в развитии.

После 90-х, когда программисты уже полностью освоили все тонкости разработки подобных систем и основанные на правилах алгоритмы логического вывода стали еще одним инструментом в арсенале IT-специалистов, множество ведущих компаний, специализирующихся на разработке программных бизнес-комплексов, начали интегрировать в свои продукты возможности ЭС как один из способов описания бизнес-логики.

Не стоит, однако, делать поспешный вывод о том, что ЭС являются единственными представителями ИИС. На самом деле возможности по применению ИИС довольно широкие. В круг наиболее распространенных задач, с которыми они способны справляться, входят: интерпретация данных, диагностика, мониторинг, проектирование, прогнозирование, планирование, обучение, управление, принятие решений.

Примером самой популярной и успешной ИИС в наши дни служит Google Translate [5]. Этот сервис реализует концепцию *статистического машинного перевода* [6], где база знаний формируется на основе статистических моделей, полученных в результате анализа больших корпусов текстов.

Несмотря на кажущуюся сложность, ядро ИИС состоит всего из двух компонентов: *базы знаний* (БЗ) и *машины вывода* (МВ).

Говоря простыми словами, БЗ – это технология представления и хранения данных – фактов об окружающем мире [7]. Ранние системы имели довольно примитивную БЗ, которая представляла собой

обычные присваивания значений переменным. Само понятие возникло для того, чтобы подчеркнуть отличие таких хранилищ от традиционных *баз данных* (БД), поскольку ИИС оперируют структурированной информацией (иногда связанной с помощью указателей и перекрестных ссылок), а не просто таблицами с числами и строками. Тем не менее по мере развития информационных технологий БЗ стали приобретать черты, характерные для объектных БД (классы и экземпляры), в результате чего различия стали менее очевидными.

Вся хранящаяся в БЗ информация должна удовлетворять двум важным требованиям: достоверности и релевантности, т.е. она должна быть объективной и принципиально доказуемой [8].

Стоит отдельно отметить, что в связи с растущей значимостью сети Internet и количеством web-контента, понятие «база знаний» часто используется для обозначения большого репозитория с документами, которые, в отличие от ИИС, предназначены для использования людьми, а не какой-либо программной системой.

Машина вывода – второй важный элемент ИИС – служит для получения логических выводов путем применения правил к имеющимся в БЗ фактам [9]. Новые данные, полученные в результате этих вычислений, могут привести к срабатыванию других правил, что, в свою очередь, инициирует новую итерацию.

Обычно МВ представляет собой совокупность *if-then* правил. Исследователями в области ИИ была предпринята попытка использования вместо таких выражений более унифицированного исчисления предикатов (логики первого порядка), но эта идея не прижилась, так как любое обобщающее утверждение бесконечного множества вводит программу в нескончаемый цикл. Кроме того, очевидно, что психологически люди склонны к более простой и понятной *if-then* схеме для описания более сложной логики.

Раньше для написания МВ в основном применялись такие языки программирования (ЯП), как Lisp (Common Lisp, Scheme) и Prolog. Сейчас большой популярностью пользуются современные динамически интерпретируемые скриптовые ЯП (например, Python, Ruby), так как они позволяют вносить поправки в конфигурацию и менять логику работы приложения «на лету», без необходимости перекомпиляции модулей программы.

Машина вывода может работать в одном из двух возможных режимов: прямое связывание и обратное связывание.

Прямое связывание является более популярной и легко реализуемой стратегией логического вывода [10]. В этом режиме алгоритм начинает анализ с имеющихся данных и использует правила для выведения новых фактов. При этом выполняется просмотр всех правил и среди них ищется такое, для которого предикат (условие) принимает значение *истина*. Если поиск был успешным, то его следствие добавляется к уже известной информации и процесс повторяется до тех пор, пока не будет достигнута цель. Важным преимуществом этого метода является то, что он лучше всего подходит для работы в динамических системах, где в зависимости от ситуации могут меняться условия.

Обратное связывание, как и следует из названия, работает в строго противоположном направлении [11]. Для указанной цели осуществляется выборка правил, которые ведут к ней. Затем осуществляется анализ фактов, соответствующих этим правилам. В конечном счете, будет получено множество всех возможных комбинаций начальных условий, необходимых для достижения цели. Этот алгоритм часто применяется для доказательства теорем и для поиска решений в теории игр.

Как и для любого программного продукта, техника проектирования ИИС тесно связана с выбранной моделью разработки ПО. Но наиболее важным аспектом этого процесса является разработка БЗ, поскольку качество исполнения данного компонента (достоверность знаний) определяет функциональную пригодность получаемого на выходе программного продукта.

В процессе создания БЗ принимают непосредственное участие три вида специалистов: эксперты проблемной области, инженеры по знаниям и программисты, на которых возлагается практическая реализация алгоритмической составляющей.

Разработка БЗ включает несколько последовательных этапов. Вначале ставится цель и выявляется круг задач, подлежащих решению, определяется тип конечного пользователя. Затем следует этап извлечения знаний, когда с помощью экспертов выполняется тщательный анализ предметной области, который подразумевает выделение используемых понятий и связей между ними. Полученные таким образом знания после формализации подлежат дальнейшей структуризации с определением способов их представления и интерпретации, т.е. того, как система будет манипулировать ими. Когда методы решения задач ясны, можно приступить к моделированию работы системы и оценить ее адекватность. Затем следует наиболее трудоемкий этап, заключающийся в наполнении БЗ информацией, полученной инженерами по знаниям на основе анализа опыта экспертов, накопленного ими при решении реальных проблем. После этого создается один или несколько прототипов системы и осуществляется тестирование. По его результатам делается вывод о применимости выбранного способа представления знаний.

Проблемным местом при проектировании ИИС является то обстоятельство, что даже на сегодняшний день не найдено какое-либо универсальное логико-математическое и программное решение,

способное удовлетворить запросы разработчиков любой разновидности ИИС. В отличие от БД, фактически каждый раз приходится либо искать способы комбинирования и применения уже готовых наработок (технологий и алгоритмов), либо пытаться решить проблему самостоятельно.

Однако ИИС продолжают активно разрабатываться и успешно применяются в различных областях человеческой деятельности, начиная с автоматизации отдельных бизнес-процессов и заканчивая управлением деятельностью целых организаций (авиакомпания, железная дорога, NASA и т.п.). Если говорить об ЭС в отдельности, то, конечно же, они пока не способны полностью заменить реального человека. Потому что, во-первых, создание этих систем требует участие экспертов (значит, спрос на специалистов по-прежнему остается актуальным) и, во-вторых, в любом случае решающее слово всегда остается за человеком (директором, менеджером и т.п.). В то время как компьютерная программа, даже превосходя человека по скорости получения логического вывода, способна работать только с объективными данными и не обладает таким качеством, как *интуиция*, а значит, не может делать заключения с учетом эмпирической оценки ситуации. Поэтому данным системам обычно отводится второстепенная роль как вспомогательный инструмент для самопроверки.

ЛИТЕРАТУРА

1. Knowledge-based systems [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Knowledge-based_systems. – Date of access: 17.09.2016.
2. Интеллектуальная информационная система [Электронный ресурс] / Википедия – Свободная энциклопедия. – Режим доступа: https://ru.wikipedia.org/wiki/Интеллектуальная_информационная_система. – Дата доступа: 17.09.2016.
3. Expert systems [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Expert_system. – Date of access: 18.09.2016.
4. Экспертная система [Электронный ресурс] / Википедия – Свободная энциклопедия. – Режим доступа: https://ru.wikipedia.org/wiki/Экспертная_система. – Дата доступа: 18.09.2016.
5. Google Translate [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Google_Translate. – Date of access: 19.09.2016.
6. Statistical machine translation [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Statistical_machine_translation. – Date of access: 19.09.2016.
7. Knowledge base [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Knowledge_base. – Date of access: 20.09.2016.
8. База знаний [Электронный ресурс] / Википедия – Свободная энциклопедия. – Режим доступа: https://ru.wikipedia.org/wiki/База_знаний. – Дата доступа: 20.09.2016.
9. Inference engine [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Inference_engine. – Date of access: 20.09.2016.
10. Forward chaining [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Forward_chaining. – Date of access: 20.09.2016.
11. Backward chaining [Electronic resource] / Wikipedia – The Free Encyclopedia. – Mode of access: https://en.wikipedia.org/wiki/Backward_chaining. – Date of access: 20.09.2016.