

Лекция 12. Разработка экспертных систем

Цель: Изучение этапов разработки ЭС.

План:

1. Этапы разработки ЭС
2. Выбор подходящей проблемы
3. Разработка прототипа
4. Развитие прототипа до промышленной ЭС
5. Оценка системы
6. Стыковка системы
7. Поддержка системы

На сегодняшний день следует констатировать, что разработка программных комплексов экспертных систем как за рубежом, так и в нашей стране осталась скорее на уровне искусства, чем науки. Это связано с тем, что долгое время системы искусственного интеллекта внедрялись в основном во время фазы проектирования, а чаще всего разрабатывалось несколько прототип-ных версий программ, и на их основе уже создавался конечный продукт. Такой подход действует хорошо в исследовательских условиях, однако в коммерческих условиях он является слишком дорогим, чтобы оправдать затраты на разработку.

Процесс разработки промышленной экспертной системы, опираясь на традиционные технологии [Николов и др., 1990; Хейес-Рот и др., 1987; Tuthill, 1994], практически для любой предметной области можно разделить на шесть более или менее независимых этапов (рис. 13).

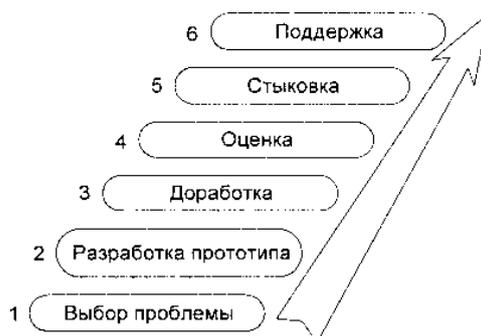


Рис. 13. Этапы разработки ЭС

Последовательность этапов дана только с целью получения общего представления о процессе создания идеального проекта, т. к. последовательность эта не вполне фиксирована. В действительности каждый последующий этап разработки может принести новые идеи, которые могут повлиять на предыдущие решения и даже привести к их переработке. Именно поэтому многие специалисты по информатике весьма критично относятся к методологии экспертных систем. Они считают, что расходы на разработку таких систем очень большие, время разработки слишком велико, а полученные в результате программы накладывают тяжелое бремя на вычислительные ресурсы.

В целом за разработку экспертных систем целесообразно браться организации, где накоплен опыт по автоматизации рутинных процедур обработки информации, таких как:

- формирование корпоративных информационных систем;
- организация сложных расчетов;
- работа с компьютерной графикой;
- обработка текстов и автоматизированный документооборот.

Решение таких задач, во-первых, подготавливает высококвалифицированных специалистов по информатике, необходимых для создания интеллектуальных систем, во-вторых, позволяет отделить от экспертных систем неэкспертные задачи.

1. Выбор подходящей проблемы

Этот этап определяет деятельность, предшествующую решению начать разрабатывать конкретную ЭС. Он включает [Николов и др., 1990]:

- определение проблемной области и задачи;
- нахождение эксперта, желающего сотрудничать при решении проблемы, и назначение коллектива разработчиков;
- определение предварительного подхода к решению проблемы;
- анализ расходов и прибылей от разработки;

- подготовку подробного плана разработки.

Правильный выбор проблемы представляет самую критическую часть разработки в целом. Если выбрать неподходящую проблему, можно очень быстро увязнуть в "болоте" проектирования задач, которые никто не знает, как решать. Неподходящая проблема может также привести к созданию экспертной системы, затраты на которую выше приносимой ею экономии. Дело будет обстоять еще хуже, если разработать систему, которая работает, но неприемлема для пользователей. Даже если разработка выполняется самой организацией для собственных целей, эта фаза является подходящим моментом для получения рекомендаций извне, чтобы гарантировать удачно выбранный и осуществимый с технической точки зрения первоначальный проект.

При выборе области применения следует учитывать, что если знание, необходимое для решения задач, постоянное, четко формулируемое, и связано с вычислительной обработкой, то обычные алгоритмические программы, по всей вероятности, будут самым целесообразным способом решения проблем в этой области.

Экспертная система ни в коем случае не устранит потребность в реляционных базах данных, статистическом программном обеспечении, электронных таблицах и системах текстовой обработки. Но если результативность задачи зависит от знания, которое является субъективным, изменяющимся, символьным или вытекающим частично из соображений здравого смысла, тогда область может обоснованно выступать претендентом на экспертную систему.

Обычно экспертные системы разрабатываются путем получения специфических знаний от эксперта и ввода их в систему. Некоторые системы могут содержать стратегии одного индивида. Следовательно, найти подходящего эксперта — это ключевой шаг в создании экспертных систем.

В процессе разработки и последующего расширения системы инженер по знаниям и эксперт обычно работают вместе. Инженер по знаниям помогает

эксперту структурировать знания, определять и формализовать понятия и правила, необходимые для решения проблемы.

Во время первоначальных бесед они должны решить, будет ли их сотрудничество успешным. Это немаловажно, поскольку обе стороны будут работать совместно, по меньшей мере, в течение одного года. Кроме них в коллектив разработчиков целесообразно включить потенциальных пользователей и профессиональных программистов. Подробно функции каждого члена коллектива описаны в следующем разделе.

Предварительный подход к программной реализации задачи определяется, исходя из характеристик задачи и ресурсов, выделенных на ее решение. Инженер по знаниям выдвигает обычно несколько вариантов, связанных с использованием имеющихся на рынке программных средств. Окончательный выбор возможен лишь на этапе разработки прототипа.

После того как задача определена, необходимо подсчитать расходы и прибыль от разработки экспертной системы. В расходы включаются затраты на оплату труда коллектива разработчиков. В дополнительные расходы будет включена стоимость приобретаемого программного инструментария, с помощью которого будет разработана экспертная система.

Прибыль может быть получена за счет снижения цены продукции, повышения производительности труда, расширения номенклатуры продукции или услуг или даже разработки новых видов продукции или услуг в области, в которой будет использоваться ЭС. Соответствующие расходы и прибыль от системы определяются относительно времени, в течение которого возвращаются средства, вложенные в разработку. На современном этапе большая часть фирм, развивающих крупные экспертные системы, предпочли разрабатывать дорогостоящие проекты, приносящие значительную прибыль.

Можно ожидать развития тенденции разработки менее дорогостоящих систем, хотя и с более длительным сроком окупаемости вложенных в них

средств, т. к. программные средства разработки экспертных систем непрерывно совершенствуются.

После того как инженер по знаниям убедился, что:

- данная задача может быть решена с помощью экспертной системы;
- экспертную систему можно создать предлагаемыми на рынке средствами;
- имеется подходящий эксперт;
- предложенные критерии производительности являются разумными;
- затраты и срок их окупаемости приемлемы для заказчика, он составляет план разработки. План определяет шаги процесса разработки и необходимые затраты, а также ожидаемые результаты.

2. Разработка прототипа

Прототипная система является усеченной версией экспертной системы, спроектированной для проверки правильности кодирования фактов, связей и стратегий рассуждения эксперта. Она также дает возможность инженеру по знаниям привлечь эксперта к активному участию в процессе разработки экспертной системы, и, следовательно, к принятию им обязательства приложить все усилия к созданию системы в полном объеме.

Объем прототипа — несколько десятков правил, фреймов или примеров. На рис. 14 изображены шесть стадий разработки прототипа и минимальный коллектив разработчиков, занятых на каждой из стадий (пять стадий заимствованы из [Хейес-Рот и др., 1987]). Приведем краткую характеристику всех стадий, хотя эта схема представляет собой грубое приближение к сложному, итеративному процессу.

Несмотря на то, что любое теоретическое разделение бывает часто условным, осознание коллективом разработчиков четких задач каждой стадии представляется целесообразным. Роли разработчиков (эксперт, програм-

мист, пользователь и аналитик) являются постоянными на протяжении всей разработки. Совмещение ролей нежелательно.

Сроки приведены условно, т. к, зависят от квалификации специалистов и особенностей задачи.

Идентификация проблемы

Уточняется формулировка (спецификация) задачи, планируется ход разработки прототипа экспертной системы, определяются:

- необходимые ресурсы (время, люди, компьютеры, деньги и т. д.);
- источники знаний (книги, дополнительные эксперты, методики);
- имеющиеся аналогичные экспертные системы;
- цели (распространение опыта, автоматизация рутинных действий и др.);
- классы решаемых задач и т. д.

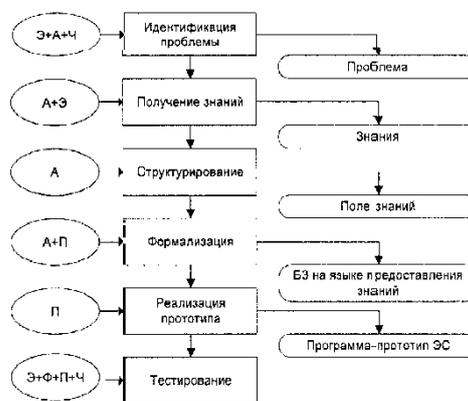


Рис. 14. Стадии разработки прототипа ЭС: Э — эксперт; А — аналитик (инженер по знаниям); П — программист; Ч — пользователь ("чайник")

Определение

Идентификация проблемы — знакомство и обучение членов коллектива разработчиков, а также создание неформальной спецификации задачи.

Средняя продолжительность 1—2 недели.

Извлечение знаний

На этой стадии происходит перенос компетентности от эксперта к инженеру по знаниям с использованием различных методов :

- анализ текстов;
- диалоги;
- экспертные игры;
- лекции;
- дискуссии;
- интервью;
- наблюдение и др.

Определение

Извлечение знаний— получение инженером по знаниям наиболее полного из возможных представлений о предметной области и способах принятия решения в ней.

Средняя продолжительность 1—3 месяца.

Структурирование или концептуализация знаний

Через выявление структуры полученных знаний о предметной области определяются:

- терминология;
- список основных понятий и их атрибутов;
- отношения между понятиями;
- структура входной и выходной информации;
- стратегия принятия решений;
- ограничения стратегий и т. д.

Определение

Структурирование (или концептуализация) знаний — разработка неформального наглядного описания знаний о предметной области в виде графа, таблицы, диаграммы или текста, которое отражает основные концепции и взаимосвязи между понятиями предметной области.

Такое описание называется *полем знаний*. Средняя продолжительность этапа 2—4 недели.

Формализация знаний

Строится формализованное представление концепций предметной области на основе выбранного языка представления знаний (ЯПЗ). Традиционно на этом этапе используются:

- логические методы (исчисления предикатов 1-го порядка и др.);
- продукционные модели (с прямым и обратным выводом);
- семантические сети;
- фреймы;
- объектно-ориентированные языки, основанные на иерархии

классов, объектов.

Определение

Формализация знаний — разработка базы знаний на языке представления знаний, который, с одной стороны, соответствует структуре поля знаний, а с другой — позволяет реализовать прототип системы на следующей стадии программной реализации.

Все чаще на этой стадии используется симбиоз языков представления знаний, например, в системе ОМЕГА [Справочник по ИИ, 1990] — фреймы + семантические сети + полный набор возможностей языка исчисления предикатов. Средняя продолжительность 1—2 месяца.

Программная реализация

Создается прототип экспертной системы, включающий базу знаний и остальные блоки, при помощи одного из следующих способов:

- программирование на традиционных языках типа Pascal, C++ и др.;
- программирование на специализированных языках, применяемых в задачах искусственного интеллекта: LISP [Хювянен, Сеппянен, 1991], FRL [Байдун, Бунин, 1990], SMALLTALK [Справочник по ИИ, 1990] и др.;
- использование инструментальных средств разработки ЭС PIES

[Хорошевский, 1993]; G2 [Попов, Фоминых, Кисель, 1996];

➤ использование "пустых" ЭС или "оболочек" типа ЭКСПЕРТ [Кирсанов, Попов, 1990], ФИАКР [Соловьев, Соловьева, 1989] и др.

Определение

Программная реализация— разработка программного комплекса, демонстрирующего жизнеспособность подхода в целом.

Чаще всего первый прототип отбрасывается на этапе реализации действующей ЭС.

Средняя продолжительность 1—2 месяца. Более подробно эти вопросы рассматриваются в следующих главах.

Тестирование

Оценивается и проверяется работа программ прототипа с целью приведения в соответствие с реальными запросами пользователей. Прототип проверяется:

- на удобство и адекватность интерфейсов ввода/вывода (характер вопросов в диалоге, связность выводимого текста результата и др.);
- на эффективность стратегии управления (порядок перебора, использование нечеткого вывода и др.);
- на качество проверочных примеров;
- на корректность базы знаний (полнота и непротиворечивость правил).

Определение

Тестирование — выявление ошибок в подходе и реализации прототипа и выработка рекомендаций по доводке системы до промышленного варианта.

Средняя продолжительность 1—2 недели.

3. Развитие прототипа до промышленной ЭС

При неудовлетворительном функционировании прототипа эксперт и инженер по знаниям имеют возможность оценить, что именно будет включено в разработку окончательного варианта системы.

Если первоначально выбранные объекты или свойства оказываются неподходящими, их необходимо изменить. Можно сделать оценку общего числа эвристических правил, необходимых для создания окончательного варианта экспертной системы. Иногда [Хювянен, Сеппянен, 1991] при разработке промышленной и/или коммерческой системы выделяют следующие дополнительные этапы для перехода (табл. 1.2):

- демонстрационный прототип;
- исследовательский прототип;
- действующий прототип;
- промышленная система;
- коммерческая система.

Однако чаще реализуется плавный переход от демонстрационного прототипа к промышленной системе, при этом, если программный инструментарий был выбран удачно, не обязательно даже переписывать окончательный вариант другими программными средствами.

Понятие же коммерческой системы в нашей стране входит в понятие "промышленный программный продукт", или "промышленная ЭС" (в этой работе).

Таблица 1.2. Переход от прототипа к промышленной экспертной системе

Этапы развития прототипа	Функциональность прототипа
Демонстрационный прототип ЭС	Система решает часть задач, демонстрируя жизнеспособность подхода (несколько десятков правил или понятий)
Исследовательский прототип ЭС	Система решает большинство задач, но неустойчива в работе и не полностью проверена (несколько сотен правил или понятий)
Действующий прототип ЭС	Система надежно решает все задачи на реальных примерах, но для сложной задачи требует много

	времени и памяти
Промышленная система	Система обеспечивает высокое качество решений при минимизации требуемого времени и памяти; переписывается с использованием более эффективных средств представления знаний
Коммерческая система	Промышленная система, пригодная к продаже, хорошо документирована и снабжена сервисом

Основная работа на данном этапе заключается в существенном расширении базы знаний, т. е. в добавлении большого числа дополнительных правил, фреймов, узлов семантической сети или других элементов знаний. Эти элементы знаний обычно увеличивают *глубину системы*, обеспечивая большее число правил для трудноуловимых аспектов отдельных случаев. В то же время эксперт и инженер по знаниям могут увеличить базу знаний системы, включая правила, управляющие дополнительными подзадачами или дополнительными аспектами экспертной задачи (метазнания).

После установления основной структуры ЭС знаний инженер по знаниям приступает к разработке и адаптации интерфейсов, с помощью которых система будет общаться с пользователем и экспертом. Необходимо обратить особое внимание на языковые возможности интерфейсов, их простоту и удобство для управления работой ЭС. Система должна обеспечивать пользователю возможность легким и естественным образом уточнять непонятные моменты, приостанавливать работу и т. д. В частности, могут оказаться полезными графические представления.

На этом этапе разработки большинство экспертов узнают достаточно о вводе правил и могут сами вводить в систему новые правила. Таким образом, начинается процесс, во время которого инженер по знаниям передает право собственности и контроля системы эксперту для уточнения, детальной разработки и обслуживания.

4. Оценка системы

После завершения этапа разработки промышленной экспертной системы необходимо провести ее тестирование в отношении критериев эффективности. К тестированию широко привлекаются другие эксперты с целью апробирования работоспособности системы на различных примерах. Экспертные системы оцениваются главным образом для того, чтобы проверить точность работы программы и ее полезность. Оценку можно проводить, исходя из различных критериев, которые сгруппируем следующим образом:

- критерии пользователей (понятность и "прозрачность" работы системы, удобство интерфейсов и др.);
- критерии приглашенных экспертов (оценка советов-решений, предлагаемых системой, сравнение ее с собственными решениями, оценка подсистемы объяснений и др.);
- критерии коллектива разработчиков (эффективность реализации, производительность, время отклика, дизайн, широта охвата предметной области, непротиворечивость БЗ, количество тупиковых ситуаций, когда система не может принять решение, анализ чувствительности программы к незначительным изменениям в представлении знаний, весовых коэффициентах, применяемых в механизмах логического вывода, данных и т. п.).

5. Стыковка системы

На этом этапе осуществляется стыковка экспертной системы с другими программными средствами в среде, в которой она будет работать, и обучение людей, которых она будет обслуживать. Иногда это означает внесение существенных изменений. Такие изменения требуют непрямого вмешательства инженера по знаниям или какого-либо другого специалиста, который сможет модифицировать систему. Под стыковкой подразумевается также разработка связей между экспертной системой и средой, в которой она действует.

Когда экспертная система уже готова, инженер по знаниям должен убедиться в том, что эксперты, пользователи и персонал знают, как

эксплуатировать и обслуживать ее. После передачи им своего опыта в области информационной технологии инженер по знаниям может полностью предоставить ее в распоряжение пользователей.

Для подтверждения полезности системы важно предоставить каждому из пользователей возможность поставить перед ЭС реальные задачи, а затем проследить, как она выполняет эти задачи. Для того чтобы система была одобрена, необходимо представить ее как помощника, освобождающего пользователей от обременительных задач, а не как средство их замещения.

Стыковка включает обеспечение связи ЭС с существующими базами данных и другими системами на предприятии, а также улучшение системных факторов, зависящих от времени, чтобы можно было обеспечить ее более эффективную работу и улучшить характеристики ее технических средств, если система работает в необычной среде (например, связь с измерительными устройствами).

Так успешно была состыкована со своим окружением система PUFF — экспертная система для диагностики заболеваний легких [Хейес-Рот и др., 1987]. После того как PUFF была закончена и все были удовлетворены ее работой, систему перекодировали с LISP на Бейсик. Затем систему перенесли на ПЭВМ, которая уже работала в больнице. В свою очередь, эта ПЭВМ была связана с измерительными приборами. Данные с измерительных приборов сразу поступают в ПЭВМ. PUFF обрабатывает эти данные и печатает рекомендации для врача. Врач в принципе не взаимодействует с PUFF. Система полностью интегрирована со своим окружением — она представляет собой интеллектуальное расширение аппарата исследования легких, который врачи давно используют.

Другой системой, которая хорошо функционирует в своем окружении, являлась CAT-1 [Уотермен, 1990] — экспертная система для диагностики неисправностей дизелей локомотивов.

Эта система была разработана также на языке LISP, а затем была переведена на FORTH с тем, чтобы ее можно было более эффективно

использовать в различных локомотивных цехах. Мастер по ремонту запрашивает систему о возможных причинах неисправности дизеля. Система связана с видеодиском, с помощью которого мастеру показывают визуальные объяснения и подсказки, касающиеся более подробных проверок, которые он должен сделать.

Кроме того, если оператор не уверен в том, как устранить неисправность, система предоставляет ему обучающие материалы, которые фирма подготовила предварительно, и показывает ему их на видеотерминале. Таким образом, мастер по ремонту может с помощью экспертной системы диагностировать проблему, найти тестовую процедуру, которую он должен использовать, получить на дисплее объяснение, как провести тест, или получить инструкции о том, как справиться с возникшей проблемой.

6. Поддержка системы

При перекодировании системы на язык, подобный Си, повышается ее быстроедействие и увеличивается переносимость, однако гибкость при этом уменьшается. Это приемлемо лишь в том случае, если система сохраняет все знания проблемной области, и это знание не будет изменяться в ближайшем будущем. Однако если экспертная система создана именно из-за того, что проблемная область изменяется, то необходимо поддерживать систему в ее инструментальной среде разработки.

Удачным и ставшим уже хрестоматийным примером ЭС, внедренной таким образом, является XCON (R1) — ЭС, которую фирма DEC использовала для комплектации ЭВМ семейства VAX. Одной из ключевых проблем, с которой столкнулась фирма DEC, являлась необходимость постоянного внесения изменений для новых версий оборудования, новых спецификаций и т. д. Для этой цели XCON поддерживается в программной среде OPS5.

Контрольные вопросы:

1. Перечислите этапы разработки ЭС.
2. Опишите этап «разработка прототипа ЭС»

3. Определение термина «структурирование знаний».

Литература:

1. А. П. Частиков Т. А. Гаврилова Д. Л.Белов. РАЗРАБОТКА
ЭКСПЕРТНЫХ СИСТЕМ. СРЕДА CLIPS. СПб: БХВ-Петербург,
2003

Ключевые слова:

Этапы разработки ЭС, прототип, оценка системы, стыковка
системы, поддержка системы