

Лекция 8

Тема: Экспертные системы: Определения и классификация

Одним из наиболее значительных достижений искусственного интеллекта стала разработка мощных компьютерных систем, получивших название "экспертных" или основанных на "знаниях" систем. В современном обществе при решении задач управления сложными многопараметрическими и сильносвязанными системами, объектами, производственными и технологическими процессами приходится сталкиваться с решением неформализуемых либо трудноформализуемых задач. Такие задачи часто возникают в следующих областях: авиация, космос и оборона, нефтеперерабатывающая промышленность и *транспортировка* нефтепродуктов, химия, энергетика, металлургия, целлюлозно-бумажная промышленность, *телекоммуникации* и *связь*, пищевая промышленность, машиностроение, производство цемента, бетона и т.п. транспорт, медицина и фармацевтическое производство, административное управление, прогнозирование и *мониторинг*. Наиболее значительными достижениями в этой области стало создание систем, которые ставят диагноз заболевания, предсказывают месторождения полезных ископаемых, помогают в проектировании электронных устройств, машин и механизмов, решают задачи управления реакторами и другие задачи. Примеры *экспертных систем* в различных предметных областях приводятся в конце лекции.

Под *экспертной системой (ЭС)* будем понимать программу, которая использует знания специалистов (*экспертов*) о некоторой конкретной узко специализированной предметной области и в пределах этой области способна принимать решения на уровне *эксперта* - профессионала.

Осознание полезности систем, которые могут копировать дорогостоящие или редко встречающиеся человеческие знания, привело к широкому внедрению и расцвету этой технологии в 80-е, 90-е годы прошлого века. Основу успеха ЭС составили два важных свойства, отмечаемые рядом исследователей:

- в ЭС знания отделены от данных, и мощность *экспертной системы* обусловлена в первую очередь мощностью базы знаний и только во вторую очередь используемыми методами решения задач;
- решаемые ЭС задачи являются неформализованными или слабоформализованными и используют эвристические, экспериментальные, субъективные знания *экспертов* в определенной предметной области.

Основными категориями решаемых ЭС задач являются: *диагностика*, управление (в том числе технологическими процессами), *интерпретация*, прогнозирование, проектирование, *отладка* и ремонт, планирование, наблюдение (*мониторинг*), обучение.

Обобщенная схема ЭС приведена на рис. 1. Основу ЭС составляет подсистема *логического вывода*, которая использует информацию из базы знаний (БЗ), генерирует рекомендации по решению искомой задачи. Чаще всего для *представления знаний* в ЭС используются *системы продукции* и *семантические сети*. Допустим, БЗ состоит из фактов и правил (если <посылка> то <заключение>). Если ЭС определяет, что посылка верна, то правило признается подходящим для данной консультации и оно запускается в действие. *Запуск* правила означает принятие заключения данного правила в качестве составной части процесса консультации.

Обязательными частями любой ЭС являются также *модуль* приобретения знаний и *модуль* отображения и объяснения решений. В большинстве случаев, реальные ЭС в промышленной эксплуатации работают также на основе баз данных (БД). Только одновременная работа со знаниями и большими объемами информации из БД позволяет ЭС получить неординарные результаты, например, поставить сложный диагноз (медицинский или технический), открыть месторождение полезных ископаемых, управлять ядерным реактором в реальном времени.

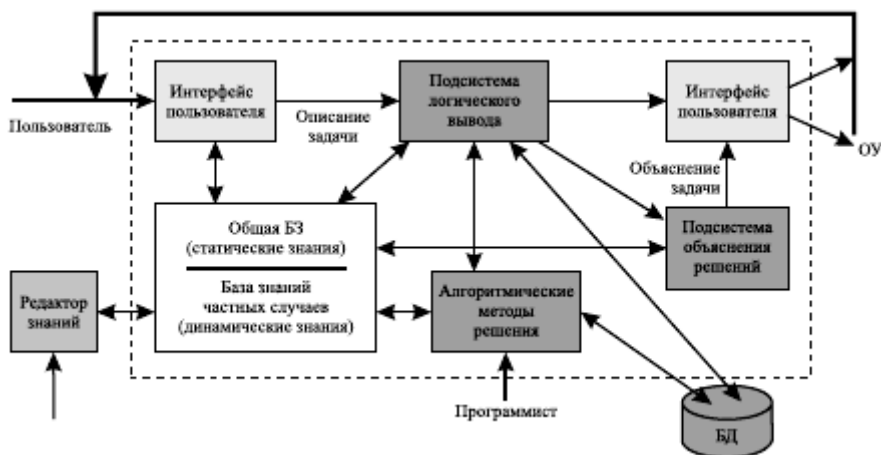


Рис. 1. Структура экспертной системы

Важную роль при создании ЭС играют инструментальные средства. Среди инструментальных средств для создания ЭС наиболее популярны такие языки программирования, как *LISP* и *PROLOG*, а также *экспертные системы-оболочки* (ЭСО): *KEE*, *CENTAUR*, *G2* и *GDA*, *CLIPS*, *AT_ТЕХНОЛОГИЯ*, предоставляющие в распоряжение разработчика - инженера *по* знаниям широкий набор для комбинирования систем *представления знаний*, языков программирования, объектов и процедур.

Рассмотрим различные способы *классификации* ЭС.

По назначению ЭС делятся на:

- ЭС общего назначения.
- Специализированные ЭС:
 1. проблемно-ориентированные для задач диагностики, проектирования, прогнозирования
 2. предметно-ориентированные для специфических задач, например, контроля ситуаций на атомных электростанциях.

По степени зависимости от внешней среды выделяют:

- Статические ЭС, не зависящие от внешней среды.
- Динамические, учитывающие динамику внешней среды и предназначенные для решения задач в реальном времени. Время реакции в таких системах может задаваться в миллисекундах, и эти системы реализуются, как правило, на языке C++.

По типу использования различают:

- Изолированные ЭС.

- ЭС на входе/выходе других систем.
- Гибридные ЭС или, иначе говоря, ЭС интегрированные с базами данных и другими программными продуктами (приложениями).

По сложности решаемых задач различают:

- Простые ЭС - до 1000 простых правил.
- Средние ЭС - от 1000 до 10000 структурированных правил.
- Сложные ЭС - более 10000 структурированных правил.

По стадии создания выделяют:

- Исследовательский образец ЭС, разработанный за 1-2 месяца с минимальной БЗ.
- Демонстрационный образец ЭС, разработанный за 2-4 месяца, например, на языке типа *LISP, PROLOG, CLIPS*
- Промышленный образец ЭС, разработанный за 4-8 месяцев, например, на языке типа *CLIPS* с полной БЗ.
- Коммерческий образец ЭС, разработанный за 1,5-2 года, например, на языке типа C++, Java с полной БЗ.

Трудности при разработке экспертных систем

Разработка ЭС связана с определенными трудностями, которые необходимо хорошо знать, так же как и спос обы их преодоления. Рассмотрим подробнее эти проблемы.

1. Проблема *извлечения знаний экспертов*. Ни один специалист никогда просто так не раскроет секреты своего профессионального мастерства, свои сокровенные знания в профессиональной области. Он должен быть заинтересован материально или морально, причем хорошо заинтересован. Никто не хочет рубить сук, на котором сидит. Часто такой специалист опасается, что, раскрыв все свои секреты, он будет не нужен компании. Вместо него будет работать *экспертная система*. Избежать эту проблему поможет выбор высококвалифицированного *эксперта*, заинтересованного в сотрудничестве.
2. Проблема *формализации знаний экспертов*. *Эксперты* -специалисты в определенной области, как правило, не в состоянии формализовать свои знания. Часто они принимают правильные решения на интуитивном уровне и не могут аргументированно объяснить, почему принято то или иное решение. Иногда *эксперты* не могут прийти к взаимопониманию (фраза "встретились два геолога, у них было три мнения" - не шутка, а реальная жизнь). В таких ситуациях поможет выбор *эксперта*, умеющего ясно формулировать свои мысли и легко объяснять другим свои идеи.
3. Проблема нехватки времени у *эксперта*. Выбранный для разработки *эксперт* не может найти достаточно времени для выполнения проекта. Он слишком занят. Он всем нужен. У него есть проблемы. Чтобы избежать этой ситуации, необходимо получить от *эксперта*, прежде чем начнется проект, согласие тратить на проект время в определенном фиксированном объеме.
4. Правила, формализованные *экспертом*, не дают необходимой точности. Проблему можно избежать, если решать вместе с *экспертом* реальные задачи. Не надо придумывать "игрушечных" ситуаций или задач. В условиях задач нужно использовать реальные данные, такие как лабораторные данные, отчеты, дневники и другую информацию, взятую из практических задач. Постарайтесь говорить с *экспертом* на одном языке, используя единую терминологию. *Эксперт*, как правило,

легче понимает правила, записанные на языке, близком к естественному, а не на языке типа *LISP* или *PROLOG*.

5. Недостаток ресурсов. В качестве ресурсов выступают персонал (*инженеры знаний*, разработчики инструментальных средств, *эксперты*) и средства построения ЭС (средства разработки и средства поддержки). Недостаток благожелательных и грамотных администраторов порождает скептицизм и нетерпение у руководителей. Повышенное внимание в прессе и преувеличения вызвали нереалистические ожидания, которые приводят к разочарованию в отношении *экспертных систем*. ЭС могут давать не самые лучшие решения на границе их применимости, при работе с противоречивыми знаниями и в рассуждениях на основе *здорового смысла*. Могут потребоваться значительные усилия, чтобы добиться небольшого увеличения качества работы ЭС. *Экспертные системы* требуют много времени на разработку. Так, создание системы PUFF для интерпретации функциональных тестов легковых автомобилей потребовало 5 человеко-лет, на разработку системы PROSPECTOR для разведки рудных месторождений ушло 30 человеко-лет, система XCON для расчета конфигурации компьютерных систем на основе VAX 11/780 потребовала 8 человеко-лет. ЭС последних лет разрабатываются более быстрыми темпами за счет развития технологий ЭС, но проблемы остались. Удвоение персонала не сокращает время разработки наполовину, потому что процесс создания ЭС - это процесс со множеством обратных связей. Все это необходимо учитывать при планировании создания ЭС.
6. Неадекватность инструментальных средств решаемой задаче. Часто определенные типы знаний (например, временные или пространственные) не могут быть легко представлены на одном ЯПЗ, так же как и разные схемы представления (например, фреймы и продукции) не могут быть достаточно эффективно реализованы на одном ЯПЗ. Некоторые задачи могут быть непригодными для решения по технологии ЭС (например, отдельные задачи анализа сцен). Необходим тщательный анализ решаемых задач, чтобы определить пригодность предлагаемых инструментальных средств и сделать правильный выбор.

О других трудностях и ловушках при создании ЭС более подробно можно прочитать в книге Д. Уотермана.

Методология построения экспертных систем

Рассмотрим методику *формализации экспертных знаний* на примере создания *экспертных диагностических систем* (ЭДС).

Целью создания ЭДС является *определение* состояния объекта диагностирования (ОД) и имеющихся в нем неисправностей.

Состояниями ОД могут быть: исправно, неисправно, работоспособно. Неисправностями, например, радиоэлектронных ОД являются обрыв связи, *замыкание* проводников, неправильное функционирование элементов и т.д.

Число неисправностей может быть достаточно велико (несколько тысяч). В ОД может быть одновременно несколько неисправностей. В этом случае говорят, что неисправности кратные.

Введем следующие определения. Разные неисправности ОД проявляются во внешней среде информационными параметрами. Совокупность значений информационных параметров определяет "информационный образ" (ИО) неисправности ОД. ИО может

быть полным, то есть содержать всю необходимую информацию для постановки диагноза, или, соответственно, неполным. В случае неполного ИО постановка диагноза носит вероятностный характер.

Основой для построения эффективных ЭДС являются знания *эксперта* для постановки диагноза, записанные в виде информационных образов, и *система представления знаний*, встраиваемая в *информационные системы* обеспечения функционирования и контроля ОД, интегрируемые с соответствующей технической аппаратурой.

Для описания своих знаний *эксперт* с помощью инженера *по* знаниям должен выполнить следующее.

1. Выделить множество всех неисправностей ОД, которые должна различать ЭДС.
2. Выделить множество *информативных (существенных) параметров*, значения которых позволяют различить каждую неисправность ОД и поставить диагноз с некоторой вероятностью.
3. Для выбранных параметров следует выделить информативные значения или информативные диапазоны значений, которые могут быть как количественными, так и качественными. Например, точные количественные значения могут быть записаны: задержка 25 нсек, задержка 30 нсек и т.д. Количественный диапазон значений может быть записан: задержка 25--40 нсек, 40--50 нсек, 50 нсек и выше. Качественный диапазон значений может быть записан: индикаторная лампа светится ярко, светится слабо, не светится.

Для более удобного дальнейшего использования качественный *диапазон* значений может быть закодирован, например, следующим образом:

- светится ярко $P1 = +++$ (или $P1 = 3$),
- светится слабо $P1 = ++$ (или $P1 = 2$),
- не светится $P1 = +$ (или $P1 = 1$).

Процедура получения информации *по* каждому из параметров определяется индивидуально в каждой конкретной системе диагностирования. Эта процедура может заключаться в автоматическом измерении параметров в ЭДС, в ручном измерении параметра с помощью приборов, качественном определении параметра, например, светится слабо, и т.д.

4. Процедура создания полных или неполных ИО каждой неисправности в алфавите значений информационных параметров может быть определена следующим образом. Составляются диагностические правила, определяющие вероятный диагноз на основе различных сочетаний диапазонов значений выбранных параметров ОД. Правила могут быть записаны в различной форме. Ниже приведена форма записи правил в виде таблицы.

Таблица 1. Диагностические правила

Номер	P1	P2	P3	Диагноз	Вероятность диагноза	Примечания
-------	----	----	----	---------	----------------------	------------

1		+++		Неисправен блок А1	0.95	
2	12-15	+		Неисправен блок А2	0.80	

Для записи правил с учетом изменений *по* времени следует ввести еще один *параметр* P0 - время (еще один столбец в таблице). В этом случае диагноз может ставиться на основе нескольких строк таблицы, а в графе Примечания могут быть указаны использованные тесты. Диагностическая *таблица* в этом случае представлена в таблице 1.

Таблица 2. Динамические диагностические правила

Номер	P0	P1	P2	P3	Диагноз	Вероятность диагноза	Примечания
1	12:00	+	+	+			тест T1
2	12:15	++	++	+	Неисправен блок А3	0.90	

Для записи последовательности проведения *тестовых процедур* и задания ограничений (если они есть) на их проведение может быть предложен аналогичный механизм. Механизм записи последовательности проведения *тестовых процедур* в виде правил реализуется, например, следующим образом:

ЕСЛИ: P2 = 1

ТО: тест = T1, T3, T7

где T1, T3, T7 - тестовые процедуры, подаваемые на ОД при активизации (срабатывании) соответствующей продукции.

В современных ЭДС применяются различные стратегии поиска решения и постановки диагноза, которые позволяют определить необходимые последовательности *тестовых процедур*. Однако приоритет в ЭС отдается прежде всего знаниям и опыту, а лишь затем *логическому выводу*.

Данная методика будет применена в следующей лекции при создании *экспертной системы* управления технологическим процессом.

Примеры экспертных систем

Для начала совершим краткий экскурс в историю создания ранних и наиболее известных ЭС. В большинстве этих ЭС в качестве СПЗ использовались *системы продукции* (правила) и прямая цепочка рассуждений. Медицинская ЭС MYCIN разработана в Стэнфордском университете в середине 70-х годов для диагностики и лечения инфекционных заболеваний крови. MYCIN в настоящее время используется для обучения врачей.

ЭС DENDRAL разработана в Стэнфордском университете в середине 60-х годов для определения топологических структур органических молекул. Система выводит молекулярную структуру химических веществ *по* данным масс-спектрометрии и ядерного магнитного резонанса.

ЭС PROSPECTOR разработана в Стэнфордском университете в 1974--1983 годах для оценки геологами потенциальной рудоносности района. Система содержит более 1000 правил и реализована на INTERLISP. *Программа* сравнивает наблюдения геологов с моделями разного рода залежей руд. *Программа* вовлекает геолога в диалог для извлечения дополнительной информации. В 1984 году она точно предсказала существование молибденового месторождения, оцененного в многомиллионную сумму.

Рассмотрим *экспертную систему* диагностирования (ЭСД) цифровых и цифроаналоговых устройств, в которой использовались системы продукции и фреймы, а также прямая и обратная цепочка рассуждений одновременно. В качестве объекта диагностирования (ОД) в ЭСД могут использоваться цифровые устройства (ЦУ), БИС, цифро-аналоговые устройства. На рис. 2 показано, что такая ЭСД работает совместно с автоматизированной системой контроля и диагностирования (АКД), которая подает в динамике воздействия на ОД (десятки, сотни и тысячи воздействий в секунду), анализирует выходные реакции и дает заключение: годен или не годен. В случае, если *реакция* проверяемого ОД не соответствует эталонным значениям, то подключается основанная на знаниях подсистема диагностирования. ЭСД запрашивает значения сигналов в определенных контрольных точках и ведет оператора *по* схеме ОД, рекомендуя ему произвести измерения в определенных контрольных точках или подтвердить промежуточный диагноз, и в результате приводит его к месту неисправности. Исходными данными для работы ЭСД являются результаты машинного моделирования ОД на этапе проектирования. Эти результаты моделирования передаются в ЭСД на магнитных носителях в виде тысяч производственных правил. Движение *по* контрольным точкам осуществляется на основе модели, записанной в виде сети фреймов для ОД.

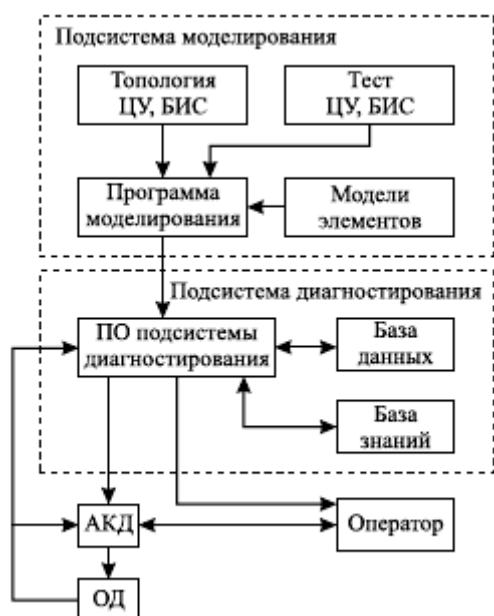


Рис. 2. Общая структура экспертной системы диагностирования

Такая ЭСД не была бы *интеллектуальной системой*, если бы она не накапливала *опыт*. Она запоминает найденную неисправность для данного типа ОД. В следующий раз при

диагностике неисправности ОД этого типа она предлагает проверить сразу же эту неисправность, если *реакция* ОД говорит о том, что такая неисправность возможна. Так поступают опытные мастера радиоэлектронной аппаратуры (РЭА), знающие "слабые" места в конкретных типах РЭА и проверяющие их в первую *очередь*. ЭСД накапливает вероятностные знания о конкретных неисправностях с целью их использования при логическом выводе. При движении *подереву* поиска решений на очередном шаге используется критерий - *максимум* отношения вероятности (коэффициента уверенности) постановки диагноза к трудоемкости распознавания неисправности. *Коэффициенты* уверенности автоматически корректируются во время работы ЭСД при каждом подтверждении или не подтверждении диагноза для конкретных ситуаций диагностирования. Трудоемкости элементарных проверок первоначально задаются *экспертом*, а затем автоматически корректируются в процессе работы ЭСД.

ЭСД не была реализована в виде ИРС *по* экономическим соображениям. Небольшая серийность проверяемой аппаратуры, недостаточная *унификация* и дешевая рабочая сила (последний фактор и в наше время играет в России немаловажную роль) помешали реализовать полностью автоматическое диагностирование.

Среди современных коммерческих систем хочется выделить *экспертную систему* - оболочку *G2* американской фирмы *Gensym (USA)* как непревзойденную экспертную коммерческую *систему* для работы с *динамическими объектами*. Работа в реальном времени с малыми временами ответа часто необходима при анализе ситуаций в корпоративных информационных сетях, на атомных реакторах, в космических полетах и множестве других задач. В этих задачах необходимо принимать решения в течение миллисекунд с момента возникновения критической ситуации. ЭС *G2*, предназначенная для решения таких задач, отличается от большинства динамических ЭС такими характерными свойствами, как:

- работа в реальном времени с распараллеливанием процессов рассуждений;
- структурированный естественно-языковой интерфейс с управлением по меню и автоматической проверкой синтаксиса;
- обратный и прямой вывод, использование *метазнаний*, сканирование и фокусирование;
- интеграция подсистемы моделирования с *динамическими моделями* для различных классов объектов;
- структурирование *БЗ*, наследование свойств, понимание связей между объектами;
- библиотеки знаний являются ASCII-файлами и легко переносятся на любые платформы и типы ЭВМ;
- развитый редактор для сопровождения *БЗ* без программирования, средства трассировки и отладки *БЗ*;
- управление доступом с помощью механизмов авторизации пользователя и обеспечения желаемого взгляда на приложение;
- гибкий интерфейс оператора, включающий графики, диаграммы, кнопки, пиктограммы и т.п.;
- интеграция с другими приложениями (по TCP/IP) и базами данных, возможность удаленной и многопользовательской работы.

В качестве примера быстродействующей системы для отслеживания состояния корпоративной информационной сети (КИС) можно привести основанную на знаниях *систему мониторинга OMEGAMON* фирмы *Candle (IBM с 2004 г.)*. *OMEGAMON* - типичный представитель современных экспертных мультиагентных динамических *систем*, работающих в реальном времени. *OMEGAMON* позволяет за считанные минуты

вести и отладить правила *мониторинга* внештатных ситуаций для объектов КИС. Правило (*situation*) записывается как продукция. *Логический вывод* в такой ЭС реализован при помощи механизма *policy*, обеспечивающего построение цепочек *логического вывода* на основе *situations*. На рис. 3 приведен один из интерфейсов для заполнения БЗ в ЭС *OMEGAMONM*. На этом рисунке показана ситуация, определяющая критическое количество сообщений в очередях транспортной системы *IBM WebSphere MQ (MQSeries)*.

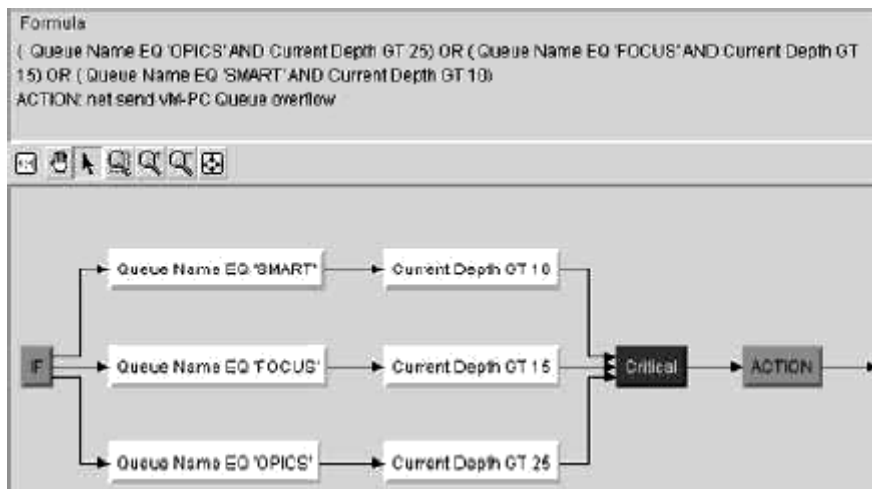


Рис. 3. Интерфейс *OMEGAMON* для заполнения БЗ

На рис. 4 показаны основные компоненты системы *OMEGAMON*:

- сервер сбора информации от агентов *CandleManagementServer (CMS)*;
- сервер отображения результатов, оповещения пользователей и *настройки мониторинга* КИС *CandleNetPortal Server (CNP)* со своими клиентами;
- *Candle Management Workstation (CMW)* - рабочая станция администратора *OMEGAMON*;
- *Managed Systems* - компьютеры КИС, на которых работают агенты.

Агенты *OMEGAMON* работают на контролируемых системах (*Managed Systems*), как первоклассные шпионы: они незаметны с точки зрения использования *CPU* и оперативны при *мониторинге* с точки зрения времени поставки своих донесений в центр (*CMS*). Они фиксируют критическую ситуацию и обеспечивают реакцию (*ACTION*) менее чем за 1 секунду. Все определяется тем интервалом *мониторинга*, который задается *экспертом* на основе своих интуитивных знаний. В качестве *ACTION* при определении ситуаций можно использовать различные типы действий: посылку почтовых сообщений и *sms* специалистам сопровождения, посылку информации в другие системы, выполнение системных команд и т.д. Количество объектов *мониторинга* (компьютеров КИС) может достигать нескольких сотен, и на каждом объекте может быть несколько сотен контролируемых параметров. Количество платформ (типов операционных систем), на которых работают агенты, превышает 30, начиная от *OS/390*, *OS/400*, далее различные *UNIX*-платформы (*HP_UX*, *AIX*, *Solaris*) и заканчивая *Windows*. На одном сервере может работать несколько агентов, например, для *мониторинга WebSphere MQ (MQSeries)*, *WebSphere Application Server*, *DB-2* и *HP_UNIX* одновременно.

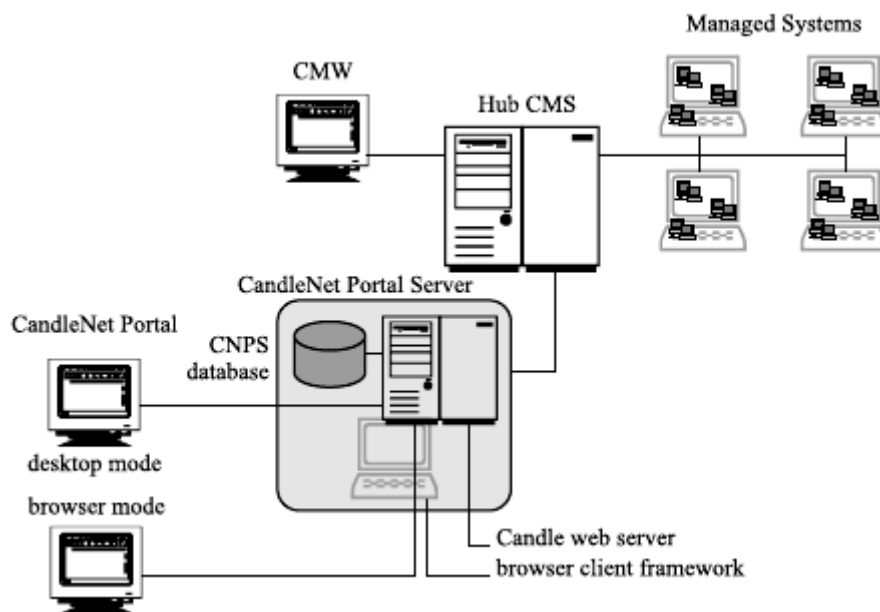


Рис. 4. Структурная схема ЭС OMEGAMON

Серверы *CMS* и *CNP*-servers могут работать на одном выделенном сервере, как правило, на базе операционной системы *Windows*. Настройка ситуаций (*situations*) и механизмов логического вывода (*policy*) производится на рабочем компьютере администратора через *CNP*-client. Для только что созданной ситуации вы нажимаете кнопку *Apply* и моментально видите *отображение ACTION* через *CNP*-client, через почту и т.д.

Следует подчеркнуть, что основанная на знаниях система мониторинга *OMEGAMON* - это весьма эффективная система управления вычислительными ресурсами, надежный и незаменимый помощник в поисках решений по оперативному устранению критических и трудных для диагностирования ситуаций, при анализе информационных потоков, анализе производительности и настройке КИС.