

ИНСТРУМЕНТАЛЬНАЯ ОБОЛОЧКА ДЛЯ СОЗДАНИЯ ГИБРИДНЫХ ЭКСПЕРТНЫХ САПР СЛОЖНЫХ ТЕХНИЧЕСКИХ ОБЪЕКТОВ

Кузьмичев В.С., Солодченко О.В.

Самарский государственный аэрокосмический университет г. Самара

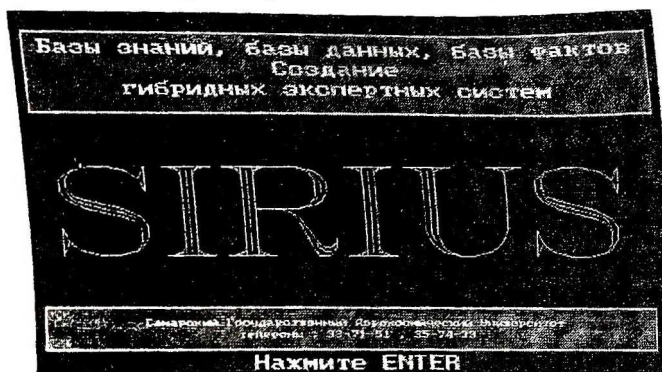
Возрастающая из года в год сложность задач, решаемых при проектировании авиационной техники требует применения более тализированных математических моделей, программного обеспечения, способного легко и оперативно модифицироваться для решения конкретных задач проектирования. Для принятия обоснованных проектных решений приходится оперировать большими объемами информации. Вместе с тем в существующих САПР достаточно эффективно используются накопленный коллективный опыт и знания в рассматриваемой области. Все это требует перехода на новую технологию проектирования, основанную на комплексном применении математического моделирования и методов искусственного интеллекта. Системы, реализующие такую технологию получили название «гибридные экспертные системы». Они характеризуются способностью формировать и пояснять свои действия и сообщения, работать с нечеткой информацией, обобщать свойства проектных решений и делать из этого обобщения выводы о том, какие действия целесообразно выполнять, расширяться путем добавления новых знаний и использовать описательное представление знаний, которое легко взаимодействует с естественным языком.

Однако разработка гибридных экспертных САПР является чрезвычайно трудоемкой и достаточно длительной задачей. Ускорить и облегчить разработку таких систем позволяет использование специально разработанных комплексов инструментальных программных средств - так называемых интегрированных оболочек.

В данной статье приводится краткое описание интегрированной оболочки для создания гибридных экспертных САПР (ГЭСАПР), разработанной в СГАУ.

Исходя из области применения ГЭСАПР проектирование сложных технических объектов, например авиационных ГТД, при

разработке системы естественно используется накопленный опыт создания традиционных систем автоматизированного проектирования, использующих для проектирования объектов соответствующие матема-



тические модели и, кроме этого добавляются новые средства обработки информации, неподдающейся формализованному описанию в виде канонических знаний, а именно логико-лингвистические модели (модели описания эвристических, неформализованных знаний и опыта в рассматриваемой предметной области). Таким образом, главным требованием к ГЭСАПР является возможность интеграции математических и логико-лингвистических моделей в едином комплексе программно-информационных средств. Для обеспечения такого рода интеграции концептуально система была разделена на математическую часть (куда входят модули расчета проектируемого объекта и его узлов, имеющие в большинстве случаев свои базы данных функциональных подсистем) и интеллектуальную компоненту (являющуюся надстройкой, берущей на себя управление функциональными модулями и процессом проектирования в целом). В информационную часть интеллектуальной надстройки входят базы знаний и базы данных. Базы знаний, содержащие декларативные и процедурные знания, обеспечивают решение следующих задач:

- хранение и накопление неформализованных знаний (декларативных и процедурных) об объекте проектирования, методах и рекомендациях по принятию решений в процессе проектирования (локальные базы знаний);
- использование эвристических знаний в процессе решения проектных задач (базы правил).

Базы данных обеспечивают:

- решение задач информационно-справочного характера (получение справочной информации из ГОСТов, ОСТов, нормативных документов, по типовым проектным решениям, патентам и т.п.);
- хранение исходной, промежуточной и результирующей информации по объекту проектирования;
- решение задач организации информационной связи функциональных подсистем САПР.

Кроме этого разработанная система удовлетворяет и общим требованиям, предъявляемым к САПР, в частности:

- обеспечивает дружелюбный интерфейс с пользователем путем общения на ограниченных конструкциях русского языка;
- ориентирована на конечного пользователя (не программиста);
- инструментальная оболочка инвариантна к предметной области;
- имеются средства оперативной модификации баз данных и баз знаний;
- имеется возможность обеспечения эволюционности развития системы;
- ядро инструментального комплекса занимает небольшой объем оперативной памяти.

В качестве языка программной реализации системы были выбраны СУБД FoxPro и Turbo Pascal 7.0

Достаточно большое количество выполнимых программ и баз данных в составе системы, требование обеспечения ее целостности, открытости и гибкости обусловили необходимость ее четкой структуризации.

Основными компонентами системы являются :

- ◆ монитор системы;
- ◆ каталог задач;
- ◆ регистратор задач;
- ◆ разработанные в системе задачи;
- ◆ машина логического вывода;
- ◆ редактор баз знаний;
- ◆ редактор баз фактов;

- ◆ редактор баз правил:
 - ◆ редактор текстов правил;
 - ◆ транслятор правил:
 - ◆ машина логического вывода ;
 - ◆ база знаний транслятора (база фактов и база правил);
 - ◆ функциональные модули транслятора;
- ◆ редактор баз данных.

Структурная схема системы представлена на рис.1. Отметим, что под задачей здесь понимается в общем случае задача, решаемая гибридной экспертной системой. В частном случае это может быть чисто экспертная система (если в процессе решения (логического вывода) не требуется выполнение функциональных модулей, основанных на математических моделях), либо чисто функциональная система (если модуль, реализующий математические модели не использует машину логического вывода).

Особенностью данной инструментальной оболочки является то, что транслятор правил реализован в виде гибридной экспертной системы и является продуктом ее работы, доказавшем на практике собственную работоспособность и работоспособность всей системы в целом.

Транслятор состоит из четырех основных компонент:

- машина логического вывода (инвариантная к предметной области);
- база фактов транслятора;
- база правил транслятора;
- модуль связи с функциональными модулями.

Трансляция русскоязычного текста правила производится путем запуска транслятора из монитора. При этом текст правила извлекается из текстового поля базы правил и записывается в текстовый файл рабочего каталога транслятора.

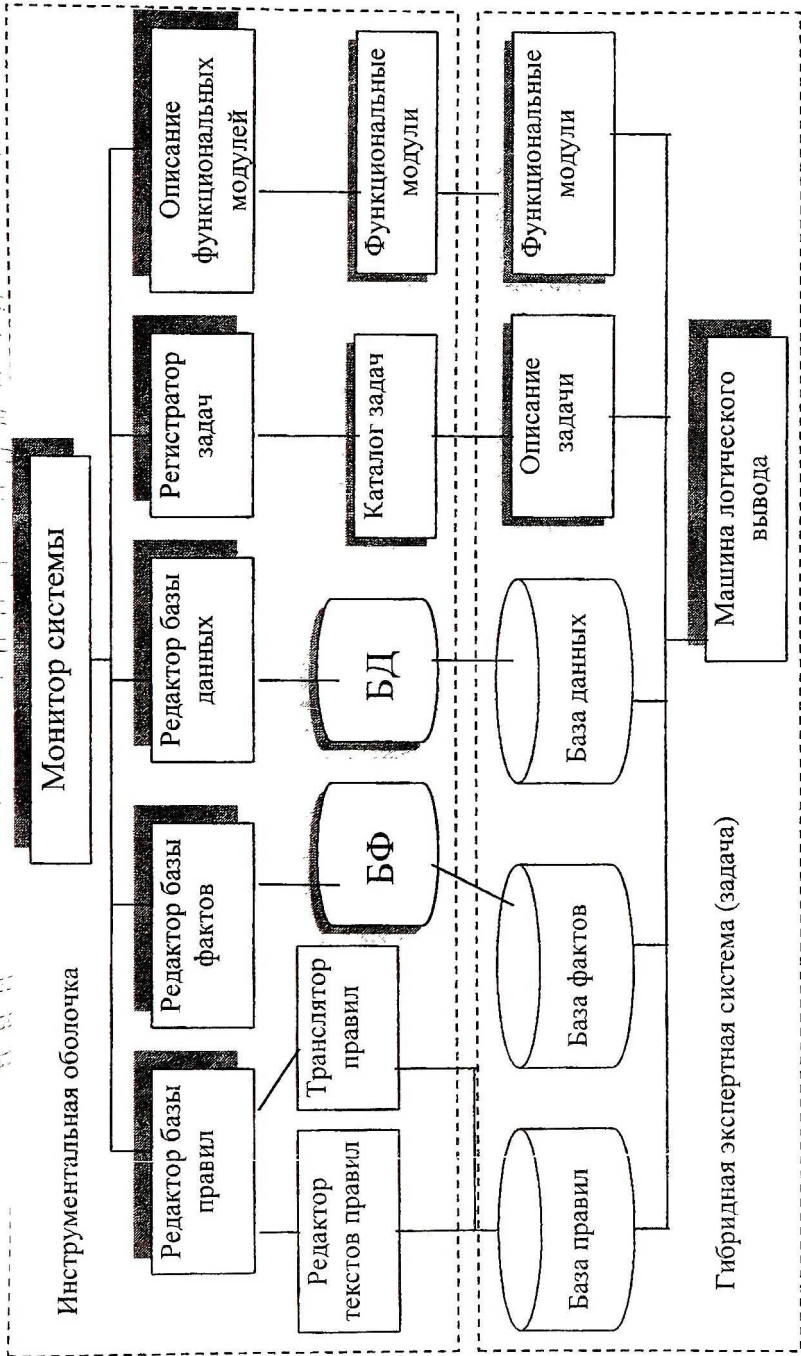


Рис. 1. Структура гибридной экспертной системы «СИРИУС»

В качестве примера в таблицах 1 и 2 приведены фрагменты базы фактов и базы правил транслятора.

Таблица 1

Фрагмент базы фактов транслятора

№ зап.	№ атриб	Имя атрибута	Значение атрибута	Тип	Форма группировки	Счетчик знач.
1	1	количество слов предложения	1 .. 1000	целый	интервал	1
2	2	номер текущего слова	1 .. 1000	целый	интервал	1
3	3	номер первого слова условия	1 .. 1000	целый	интервал	1
4	4	номер первого слова следствия	2 .. 1000	целый	интервал	1
5	5	номер первого слова альтернативы	3 .. 1000	целый	интервал	1
6	6	счетчик лексем	1 .. 1000	целый	интервал	1
7	7	статус текущей лексемы	левый атрибут	строковый	список	17
8	7	статус текущей лексемы	правый атрибут	строковый	список	17
9	7	статус текущей лексемы	значение атрибута	строковый	список	17
10	7	статус текущей лексемы	сравнение	строковый	список	17
11	7	статус текущей лексемы	связка	строковый	список	17
12	7	статус текущей лексемы	присваивание	строковый	список	17
13	7	статус текущей лексемы	знак операции	строковый	список	17
14	7	статус текущей лексемы	имя	строковый	список	17
15	7	статус текущей лексемы	выполнение	строковый	список	17
16	7	статус текущей лексемы	просмотр	строковый	список	17
17	7	статус текущей лексемы	запрос	строковый	список	17
18	7	статус текущей лексемы	имя правила	строковый	список	17
19	7	статус текущей лексемы	следствие	строковый	список	17
20	7	статус текущей лексемы	условие	строковый	список	17
21	7	статус текущей лексемы	альтернатива	строковый	список	17
22	7	статус текущей лексемы	точка	строковый	список	17
23	7	статус текущей лексемы	парамстр запроса	строковый	список	17
24	8	статус предыдущей лексемы	левый атрибут	строковый	список	17

Фрагмент Базы правил транслятора

Имя правил	Текст правила
r1	Если статус предыдущей лексемы = условие или статус предыдущей лексемы = следствие или статус предыдущей лексемы = альтернатива и статус предыдущей лексемы = связка То статус текущей лексемы=левый атрибут и выполнить распознавание лексемы .
m7	Ели номер текущего слова > количества слов предложения То выполнить трансляцию текста правила на Паскаль и выполнить компиляцию правила иначе выполнить возврат правила на редактирование.
r0	Если номер текущего слова = 1 То статус текущей лексемы = условие и выполнить распознавание лексемы .
r2	Ели статус предыдущей лексемы = сравнение или статус предыдущей лексемы = знак операции или статус предыдущей лексемы = присваивание То статус текущей лексемы = правый атрибут и выполнить распознавание лексемы .
r3	Если статус предыдущей лексемы = сравнение или статус предыдущей лексемы = знак операции или статус предыдущей лексемы присваивание То статус текущей лексемы = значение атрибута и выполнить распознавание лексемы.

Для повышения эффективности отладки базы правил введена возможность трансляции помеченной группы правил (в частном случае- одного правила) или всей базы правил целиком. Кроме того, транслятор содержит модуль диагностики трансляции правил, который выдает информацию об успешной трансляции правил или о опущенных ошибках при создании правил. Результат работы транслятора выполнимый файл с именем, соответствующим имени транслируемого правила, который записывается в соответствующий аталог.

Описываемый вариант инструментальной оболочки предполагает написание модулей функционального наполнения на языке Pascal, каждый модуль должен иметь файл с исходным текстом и

собственную базу данных, содержащую информацию о входных данных модуля, позволяющую производить расчет с помощью тех данных, которые не используются в базе фактов системы.

Интеграция математических и логико-лингвистических моделей происходит путем вызова из правил определенных функциональных модулей, требующихся на данном этапе для выполнения; обработки выходных данных модуля и отправки этих данных обратно в правила для их последующей обработки. Одним из главных требований корректной работы функциональных модулей и транслятора является однородное описание входных и выходных данных, использующихся в процессе работы модуля, как в базе фактов системы так и в собственной базе данных модуля.

Суть интеграции функциональных модулей с базой знаний заключается в следующем:

- в базе фактов задачи описываются данные функционального модуля, которые участвуют в работе базы правил системы. Эти данные должны быть описаны и в базе данных модуля;
- для получения результатов расчета, функциональный модуль вызывается на выполнение из соответствующего правила базы правил в виде "Если <условие > То выполнить < имя модуля >" В результате трансляции этого правила происходит настройка функционального модуля на данную экспертную задачу (установка связей базы фактов задачи и базы данных модуля, ретрансляция модуля и т.д.). Естественно, что в процессе решения задачи может вызываться несколько модулей или один и тот же модуль несколько раз.

Разработанный вариант системы, получившей название «СИРИУС» прошел апробацию на целом ряде тестовых задач в области САПР ГТД, а также в течение нескольких лет используется в учебном процессе СГАУ

Принципиальной особенностью разработанной системы является его ориентация на конечного пользователя, не обладающего глубокими знаниями в области вычислительной техники и программирования.