

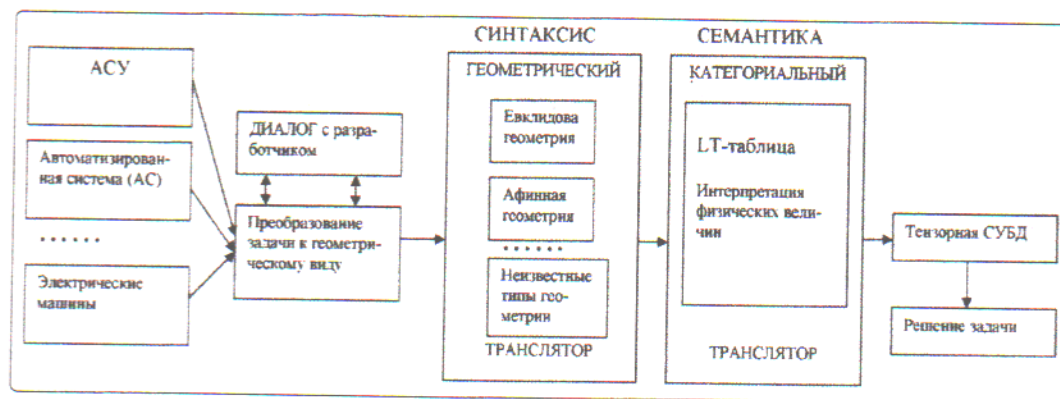


Рис. 6. Тензорная технология проектирования ИС

цедурой в этом методе является определение размера подзадач.

Проецирование постановки задачи в пространство программных модулей осуществляется путём функционального деления исходной задачи на подзадачи. При таком делении обеспечивается непротиворечивость разрабатываемой системы и всех её подсистем, так как функциональное деление обеспечивает набор аксиом для построения на основе математических теорий программных модулей. Проблемы покрытия множества алгоритмов множеством программных модулей в значительной степени зависят от конкретной ИС. При этом выбор программного модуля (или его создание) осуществляется на основе инвариантного (геометрического) объекта и исходной и результирующей систем координат.

Информационное пространство, в которое также необходимо осуществить проецирование ИС (постановки задачи), при тензорном подходе позволяет принципиально обеспечить автоматическое построение концептуальных подсхем данных. Это позволяет избежать регенерации БД при модификациях системы. Тензорная СУБД позволяет работать с динамическими описаниями (фреймами).

Следующей процедурой при разработке информационного обеспечения является выбор минимального (линейно независимого) количества параметров. Метод проецирования ИС в различные системы координат обладает тем преимуществом, что имеется инвариантный объект (постановка задачи) и он подвергается различным видам проецирования, оставаясь при этом неизменным.

Технология построения ИС тензорным методом включает: систему общения, геометрический и категориальный трансляторы, тензорную СУБД (рис.6).

На первом этапе, на естественном языке формулируется постановка задачи, стоящей перед ИС. Выделяется инвариантный объект. Например, имеются некоторые исходные данные $a, c, \dots, g$ и инвариантный объект $ab + g/c = \text{const}$. Необходимо определить $u = f(a, b, c, \dots, g)$. Величины $a, b, c, \dots, g$ имеют

конкретную интерпретацию в заданной предметной области.

Так как постановка задачи в геометрическом (тензорном) виде сложна, то на входе используется препроцессор с некоторого искусственного языка,

например профессионального языка пользователя. На основе диалога с разработчиком (или без такового) осуществляется преобразование исходной задачи к тензорному виду. При этом по определённым правилам строятся ВП1 в виде системы координат, аналогично ВП2, и определяется геометрический объект. Определяется преобразование, соответствующее ВП1, ВП2 и геометрическому объекту (тензору). Определение ВП1, ВП2, тензора и соответствующего преобразования осуществляется геометрическим транслятором. Этот транслятор производит чисто геометрические преобразования, поскольку при преобразовании задачи к геометрическому виду происходит отвлечение от сущности объектов из предметной области. Этот переход на высший уровень абстракции необходим для того, чтобы отождествлять алгоритмы, имеющие разные синтаксические конструкции, но выполняющие эквивалентные по существу преобразования. Производится синтаксический контроль решения задачи, стоящей перед ИС, т.е. определяется, правильно ли представлен геометрический объект в исходной (ВП1) и результирующей (ВП2) системах координат. Если представление неверное (неполное), осуществляется коррекция поставленной задачи.

Совокупность средств, осуществляющих семантический контроль решения задачи, стоящей перед ИС, называется категориальным транслятором. Все исходные данные задачи имеют размерность. Известно, что все размерности могут быть выражены с помощью LT -таблицы через $L^r T^s$, где r, s – целые ($r = 0, s = 0$ для безразмерных величин). Семантический контроль заключается в контроле преобразования с точки зрения размерности. Например, если в элементарном акте преобразования выполняется действие $a = bc - d/e$, то величины $a, bc, d/e$ должны иметь одинаковую размерность. Категориальный транслятор переводит размерности исходных величин в результирующие. При наличии ошибки происходит возврат к постановке задачи.

Если стадии синтаксического и семантического контроля выполнены без ошибок, то на последнем этапе происходит обращение к тензорной СУБД, подставляются значения исходных данных и задача считается решённой.