

Институт Систем Информатики им. А.П. Ершова СО РАН

На правах рукописи



Владимир Николаевич Малюх

**Методы организации функционального взаимодействия
приложений в расширяемых графических САПР**

Специальность: 05.13.11 - Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных
сетей

Автореферат диссертации
на соискание ученой степени
кандидата физико-математических наук

Новосибирск-2006

Работа выполнена в Институте систем информатики СО РАН

Научный руководитель: Доктор технических наук, профессор
Балаганский Игорь Андреевич

Официальные оппоненты: Доктор физико-математических наук,
Мушер Семен Львович
Кандидат физико-математических наук
Ушаков Дмитрий Михайлович

Ведущее предприятие: Институт Автоматики и
Электрометрии СО РАН.

Защита состоится 23 июня 2006 г. в 16:30 на заседании
диссертационного совета К003.032.01 в Институте Систем
Информатики им. А.П. Ершова Сибирского Отделения РАН по
адресу:

630090, пр. ак. Лаврентьева, 6.

С диссертацией можно ознакомиться в читальном зале библиотеки
ИСИ СО РАН (пр. ак. Лаврентьева, 6).

Автореферат разослан 22 мая 2006 г.

Ученый секретарь диссертационного
совета К003.032.01

к.ф.-м.н.



Мурзин Ф.А.

Общая характеристика работы

Актуальность темы. Системы автоматизированного проектирования (САПР) – одна из наиболее динамично развивающихся областей применения компьютеров. Непрерывное повышение производительности компьютеров сопровождающееся одновременным расширением возможностей средств разработки программного обеспечения являются ключевыми тенденциями, вызывающими практический интерес широкого круга пользователей средствам компьютерной графики и, в частности, системам геометрического моделирования и САПР. К настоящему времени созданы и находятся в эксплуатации десятки профессиональных систем геометрического моделирования, однако, их архитектура, структуры данных и методы организация взаимодействия данных, традиционно применяемые в них, имеют ряд узких мест.

Каждая новая прикладная задача, как правило, решается путем разработки специализированных модулей, для работы которых требуются структуры данных, не предусмотренные для обработки ядром системы, вследствие чего возникают коллизии в использовании ранее или параллельно разработанных модулей. Проблема усугубляется тем, что на этапе разработки базовой САПР достоверно не известны все задачи, для которых она будет применяться в будущем.

Диссертационная работа посвящена разработке архитектуры и выбору базовых алгоритмов инструментальной САПР общего назначения для создания смешанных функционально-геометрических моделей, с возможностью инкапсуляции разнородной, в том числе динамической, информации негеометрического характера в модели проектируемых объектов и организации автоматического взаимодействия таких моделей.

Целью работы является создание архитектуры инструментального ядра САПР и выбор базовых алгоритмических решений, обеспечивающих автоматическое функциональное взаимодействие независимо разработанных приложений, разработка методов и средств для построения программных систем для синтеза, хранения, отображения и анализа функционально-геометрических моделей объектов, ориентированных на использование в расширяемых системах автоматизированного проектирования промышленных изделий.

Методы исследований. Для достижения поставленной цели используются методы системного анализа существующих решений, сопоставление известных архитектур систем, экспериментальная проверка выработанных решений путем тестовых программных реализаций и сравнения их с аналогами.

Основные задачи работы:

1. Построение архитектуры базовой САПР, обеспечивающей разработчикам проблемно-ориентированных приложений возможность моделировать проектируемые объекты как в виде геометрических, так и в виде аналитических и эмпирических описаний и комбинаций этих описаний.
2. Выбор набора базовых алгоритмов, обеспечивающего расширяемость проектной среды, возможность добавления описаний новых объектов в уже существующие модели и обеспечение взаимодействия этих объектов.
3. Разработать эффективные методы представления, хранения, внедрения в проектную базу данных и управления разнородными данными, статическими ресурсами и активных компонентами

проблемно-ориентированных систем автоматизированного проектирования.

4. Разработать программный и пользовательский интерфейс для доступа к интегрированным функционально-геометрическим моделям.
5. Выработать требования и апробировать инструментальные средства проектирования и разработки активных компонент функционально-геометрических моделей.

Научная новизна работы заключается в том, что, на основе анализа формальных моделей объектов, применяемых в САПР, сформулирован новый системный набор принципов построения архитектуры базовой САПР, обеспечивающей автоматическое взаимодействие расширяемые функционально-геометрических описаний моделей:

1. Инкапсуляция в проектные данные геометрических, литеральных и функциональных описаний объектов.
2. Минимизация и унификация набора базовых геометрических представлений и используемых алгоритмов головной системы.
3. Обеспечение возможности программно переопределять в приложениях базовые методы и классы, реализующие модель проектируемого объекта.

На основе этих принципов разработана архитектура базовой САПР и осуществлена ее программная реализация.

Практическая ценность работы заключается в том, что:

1. Предложены и реализованы схемы совмещенного описания и хранения разнородных данных в системах функционально-геометрического моделирования, позволяющие создавать, совместно хранить и анализировать элементы моделей объектов как в виде

аппроксимации геометрии, так и в виде аналитических или эмпирических описаний.

2. Предложенная технология обеспечивает внедрение активных компонент в статическую геометрическую модель.
3. Созданы методы разработки интегрированных проблемно-ориентированных программно-информационных комплексов.
4. Реализованы базовые и прикладные алгоритмы в виде законченных программных компонент, позволяющих разрабатывать прикладные системы автоматизированного проектирования.

Достоверность результатов подтверждается многолетним опытом создания, внедрения и использования прикладных САПР, построенных в соответствии с изложенными в работе принципами и методами.

Личный вклад автора заключается в разработке архитектуры, выборе базовых алгоритмов инструментальной САПР, разработке технического задания на программную реализацию системы.

На защиту выносятся основные положения и результаты, сформулированные в Заключении.

Апробация работы. Результаты работы докладывались и обсуждались на Международной научно-технической конференции "Авангардные технологии, оборудование, инструмент и компьютеризация производства оптико-электронных приборов в машиностроении" (г. Новосибирск, НГТУ 1995), выставках "Компьютерная графика в архитектуре и дизайне" (г. Новосибирск, НАрХИ 1995), ежегодной международной выставке "Сибкомпьютер" (г. Новосибирск, ежегодно с 1995 г), международной выставке СеВIT'96 (Ганновер, ФРГ, 1996г), международной выставке SMAU'97 (Милан, Италия, 1997г), научно-технических семинарах ИСИ

СО РАН и кафедры газодинамических импульсных устройств НГТУ, международных конференциях isiCAD-2004 и Graphicon-2005.

Внедрение результатов работы. На основе результатов исследовательских работ разработана расширяемая среда для создания проблемно-ориентированных систем автоматизации выполнения проектных работ и, на ее базе, широкий набор специализированных прикладных систем для различных отраслей. Создана универсальная система автоматизированного проектирования bCAD и, на ее базе, ряд прикладных систем, которые используются в сотнях предприятий и организаций Российской Федерации, стран СНГ и дальнего зарубежья в частности:

Новосибирский филиал ОКБ им. Сухого

Институт Прикладной Физики, г. Новосибирск

Концерн БФК, г. Новосибирск

Завод Торгового Оборудования, г. Новосибирск

ГУП СК "Краевая техническая инвентаризация", г. Ставрополь

ЗАО «Первая мебельная фабрика», г. Москва.

Новосибирский Государственный Технический Университет

Iowa State University, США

University of Maryland, США

Altendorf GmbH, ФРГ.

Структура и объем работы

Диссертационная работа состоит из введения, четырех глав, заключения и приложения, содержит 93 страницы машинописного текста, 39 рисунков, список литературы из 61 наименования.

Содержание работы

Во введении дается краткая характеристика диссертационной работы.

Первая глава посвящена обзору литературы и решениям, применяемым в программном обеспечении автоматизации проектирования, рассматриваются принципы построения различных САПР. В программном обеспечении САПР выделяются подсистемы построения и хранения расширяемых баз данных моделей и инструментальные средства их расширения. Подробно рассматриваются типичные архитектурные решения САПР на примере таких распространенных систем как AutoCAD, SolidWorks и ArchiCAD. При анализе принципов организации баз данных существующих САПР, отмечается многообразие подходов в построении САПР различного целевого назначения и выявляются узкие моменты и ограничения в известных архитектурных решениях САПР. Делается вывод о целесообразности организации взаимодействия приложений САПР на уровне архитектуры базовой системы.

Во второй главе изложены разработанные принципы построения архитектуры САПР, обеспечивающей автоматизированное взаимодействие независимо разработанных прикладных программ САПР в общей среде. Прежде всего, необходимо определение понятия *объекта* САПР. Ниже приведены требования, которыми должны обладать объекты, являющиеся основой построения САПР.

- Объект является разновидностью другого объекта. Соответственно, имеется базовый объект, а также каталог предопределенных объектов, из которых можно выводить их разновидности.

- Объект может инкапсулировать в себе другие объекты, называемые далее *атрибутами*.
- Некоторые атрибуты являются всего лишь ссылками на другие объекты и их атрибуты; такие атрибуты называются *параметрами*. Параметр может быть помечен как входной и/или выходной.
- Каждый объект имеет ассоциированный с ним метод Build (построить) для пересчета выходных параметров по текущим значениям входных. Этот метод также может осуществлять побочные действия, например, отображение объекта на устройстве вывода.
- Каждый объект имеет ассоциированный с ним набор методов, в частности: Build, (построить), Transform (трансформировать), Display, Write/Load для пересчета и отображения выходных параметров по текущим значениям входных. Эти методы также могут осуществлять побочные действия, например, отображение объекта на устройстве вывода (дисплей, файл, печатное устройство и др.).

Для построения подсистемы геометрического моделирования существенным является наличие в САД-системе таких объектов как

- Двухмерные геометрические объекты (точка, отрезок, кривая)
- Трехмерные геометрические объекты (точка, отрезок, кривая, плоскость, поверхность, тело)
- Ограничения
- Литеральные объекты (числа, строки, стили и т.п.)

Основная задача на пути расширения геометрической модели до построения функционально-геометрической – это создание новой для

САПР совокупности методов и данных объектов, механизмов их взаимодействия, с помощью которых разработчики прикладных САПР смогут создавать геометрические и поведенческие модели.

Геометрическое представление объектов. В описываемой архитектуре САПР предлагается использовать максимально унифицированное описание геометрии объектов, изначально накладывающее как можно меньше ограничений на возможные варианты описываемых данных.

В качестве базового элемента геометрии используется **пространственный композитный контур** (рис. 1.) представляющий собой сложный цельный объект, состоящий из произвольной последовательности прямых, эллиптических дуговых и сплайновых сегментов. Сегменты, следующие друг за другом, имеют одну и только одну общую точку. Контур может иметь свойство замкнутости. Такое представление достаточно традиционное для САПР, в описываемой реализации дополнено существенным расширением: в узлах контура могут быть определены условия сопряжения сегментов – угол и радиус. На базе такого универсального описания формируются частные случаи контурных объектов – окружности, эллипсы, правильные многоугольники и т.п.

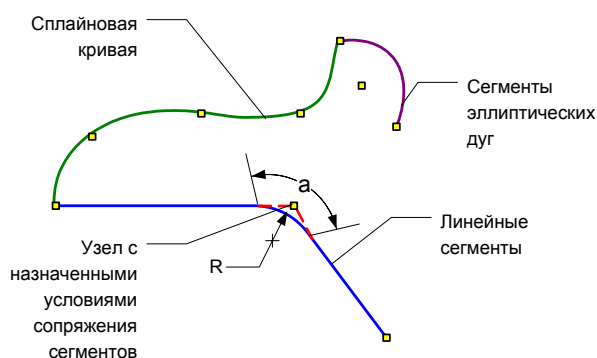


Рис.1 Композитный контур.

В качестве *описания поверхностей* выбран метод **кинематического описания**, заключающийся в том, что поверхности определяются набором образующих и направляющих контуров и способом их применения. В структуре данных, описывающих поверхность при таком описании, в отличие от традиционно принятых методов, разделены контекст построения, используемый для хранения и модификации информации о поверхности и ее аппроксимация, используемая для визуализации и, частично, для промежуточных вычислений. В качестве унифицированного набора избраны поверхности четырех основных кинематических классов (рис 2): поверхности протяжки (поз. а), поверхности Кунса (поз. б) поверхности вращения (поз. с) и плоские поверхности (поз d).

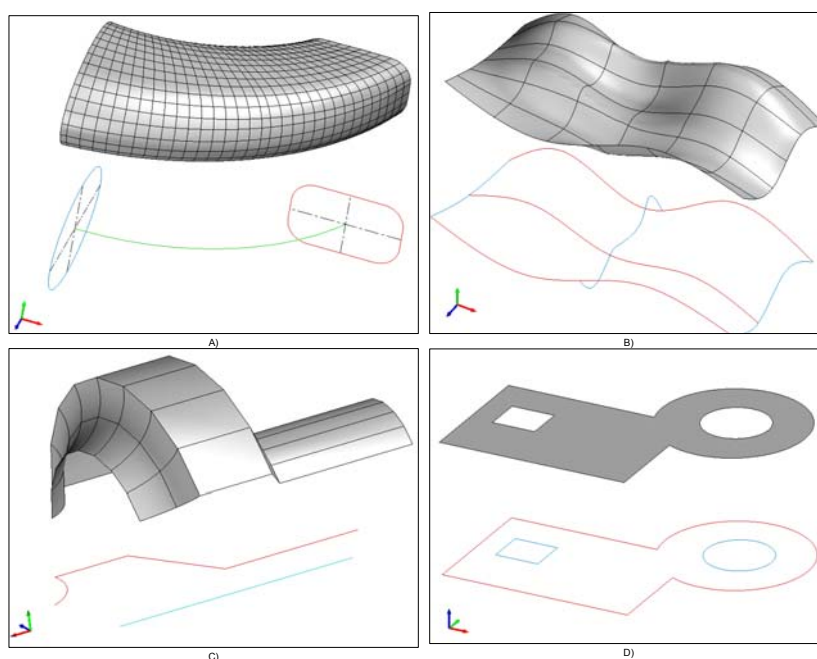


Рис.2 Классы поверхностей.

Существенным обстоятельством, предопределившим выбор унифицированных классов поверхностей, является то, что поверхности одного класса могут быть преобразованы без искажения формы в поверхность или композицию из поверхностей других классов.

Способ аппроксимации поверхностей вынесен за рамки инструментального ядра и может быть выбран разработчиком приложений. На практике наиболее часто используются либо полигональная аппроксимация, либо NURBS (Non-rational Uniform Bezier Spline).

Твердые тела описываются в рамках представляемой архитектуры комбинацией из граничного представления элементарных односвязных тел, т.н. BREP (Boundary REPresentation) и последовательностью булевых операций над ними, т.н. CSG-tree (Constrictive Solid Geometry Tree).

Суть BREP-представления (рис.3) заключается в том, что твердое тело описывается замкнутая пространственная область, ограниченная набором элементарных тонких поверхностей (граней), с общими образующими контурами (ребрами) на границе поверхностей и признаком внешней или внутренней стороны поверхности, а также рядом обязательных операций, определенных над телами.

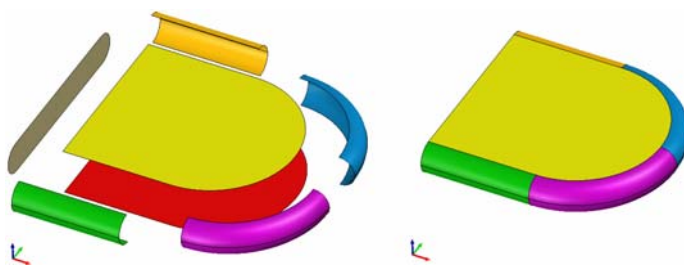


Рис.3 BREP-представление твердого тела.

Для описания сложных тел, моделирующих объекты, получаемые обработкой материала или неразъемной сборкой, используется

иерархическая структура, описывающая последовательность применения булевых операций над набором элементарных твердых тел – так называемое CSG-дерево (Constructive Solid Geometry tree). Для составных твердых тел определены операции, дополняющие набор операций, обязательных для элементарных тел: вычитание, объединение, пересечение.

Ограничение есть разновидность базового объекта, которая связывает определенным образом другие объекты базы данных проекта, а также их атрибуты (при этом, само ограничение также может иметь атрибуты). Связанные ограничением объекты являются его параметрами. Таким образом, ограничение – это всего лишь новый идентификатор для уже имеющейся в системе функциональности. Ограничения по своей природе естественно делятся на: геометрические, топологические, инженерные (прикладные).

Для надлежащей работы с ограничением нам в первую очередь необходимо развить понятие **литерального объекта**. С точки зрения структуры данных проекта литеральные объекты представляют собой шаблоны (template) ограничений, хранящиеся как самостоятельные элементы базы данных. Основные литеральные объекты по типам делятся на: символьные, целочисленные, вещественно-значные.

Кроме типа, с каждым литералом также связано *текущее значение* и *диапазон* возможных значений (заданный перечислением или интервалом). Диапазон необходим для указания области «разумных» значений параметра в случае, когда это значение вычисляется посредством автоматического удовлетворения ограничений.

Логическая структура данных. Традиционно в САПР используется одна из двух систем группирования и разделения объектов проекта. Первая, с использованием т.н. *слов* сложилась исторически по аналогии с

чертежами, элементы которых вычерчены на нескольких листах прозрачного носителя. Такой подход позволяет эффективно и интуитивно ясно управлять видимостью элементов в сложных проектах. Следует оговорить, что под видимостью понимается набор динамически настраиваемых групповых прав доступа для произвольного набора объектов. Второй подход представляет собой помещение объектов в *иерархическое дерево*, аналогично тому, как это устроено в файловой системе с директориями и файлами. Такой подход чрезвычайно удобен для описания сборок, особенно в случае использования внешних ссылок на элементы сборок, описанные в отдельных проектах.

В полной мере достоинства обоих методов сочетает предлагаемый в настоящей работе комплексный подход, в котором в качестве основной структуры используется сборочная иерархия, а в качестве способа управления видимостью частей проекта как документа - модифицированная иерархия слоев (рис. 4).

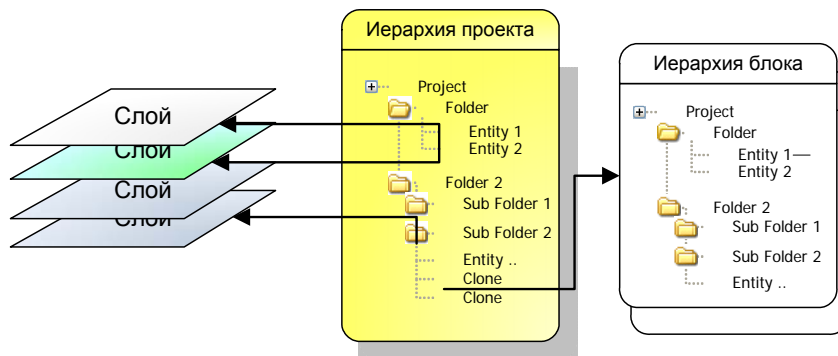


Рис.4 Логическая структура данных проекта.

Суть модификации послойного подхода заключается в том, что любой из объектов иерархии (как геометрический, литеральный, так и собственно узел дерева) может ссылаться не на один слой, а на произвольное

количество слоев одновременно. Таким образом, расширяется количество возможных комбинаций выставляемых признаков видимости объектов.

Внешние ссылки в предлагаемой архитектуре реализуются достаточно традиционным способом, путем сохранения в объекте вставки ссылки на расположение внешнего проекта с типовой внутренней структурой и, в целях достижения переносимости проектов и быстродействия базы данных проекта, кэширования экземпляра содержимого блока.

Инкапсуляция произвольных прикладных данных. Для создания прикладных систем ключевым моментом является наличие возможности расширять структуру базы данных проекта и их взаимодействие, насыщая ее прикладной информацией, заранее (на этапе разработки инструментальной системы) неизвестной структуры и поведения. В качестве решения этой задачи предлагается присоединять к стандартным объектам головной системы контейнеры, представляющие собой обычную байтовую последовательность произвольного размера, снабженную идентификатором. С этой точки зрения он подобен файлу на диске. Роль головной системы сводится к автоматическому сохранению контейнеров при записи и чтении проекта из файла, копировании и переносе объектов внутри проекта и при обмене между проектами.

Количество контейнеров, прикрепленных к одному объекту в общем случае неограниченно, хотя возможны частные решения, с фиксированным числом контейнеров. Для предотвращения использования контейнеров с данными другими приложениями каждый контейнер предлагается снабдить уникальным идентификатором, создаваемым инструментальными средствами. В простейшем случае применение вышеописанного метода внедрения прикладных данных в базу данных проекта позволяют создать *пассивную* расширяемую систему, в которой вся обработка прикладных данных производится

путем исполнения прикладных программ, лишь функции хранения данных приложений обеспечиваются головной системой.

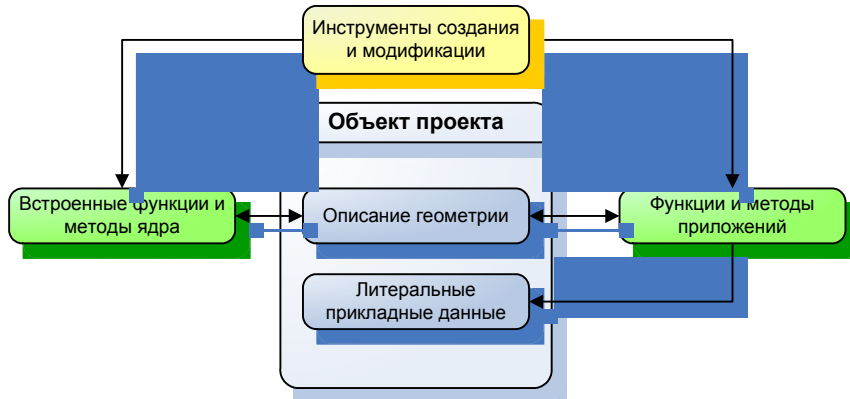


Рис.5 Структура работы с прикладными данными в пассивно расширяемой системе.

Расширение за счет инкапсуляции пассивных данных позволяет при небольших трудозатратах на разработку решать часть задач, в первую очередь связанных с постобработкой данных проекта, хранение в проекте специализированных прикладных данных, не предусмотренных разработчиками базовой САПР.

Инкапсуляция геометрических трансформируемых данных. В ряде случаев необходимо обеспечить модификацию прикладных геометрических данных синхронно с изменением базовых объектов встроенными средствами и инструментами головной системы. Наиболее типичным случаем таких модификаций являются аффинные геометрические преобразования: смещение, поворот, масштабирование, зеркальное или осесимметричное отражение. В практике трехмерной графики применяемые к объектам геометрические трансформации выражаются обобщенной матрицей аффинных преобразований в однородных координатах, содержащей в своих элементах результат

смещение, поворот и масштабирование. Для обеспечения синхронизации геометрических трансформаций предлагается кроме независимых контейнеров, данные в которых не изменяются при модификации объектов встроенными средствами головной системы предусмотреть также контейнеры специального типа, содержащие структурированную геометрическую информацию в виде последовательности трехмерных координат опорных точек. Так как встроенные данные головной системы модифицируются аналогичным образом то прилагать матрицу трансформаций к контейнерам с прикладной геометрией достаточно автоматически при каждом применении общесистемной трансформации.

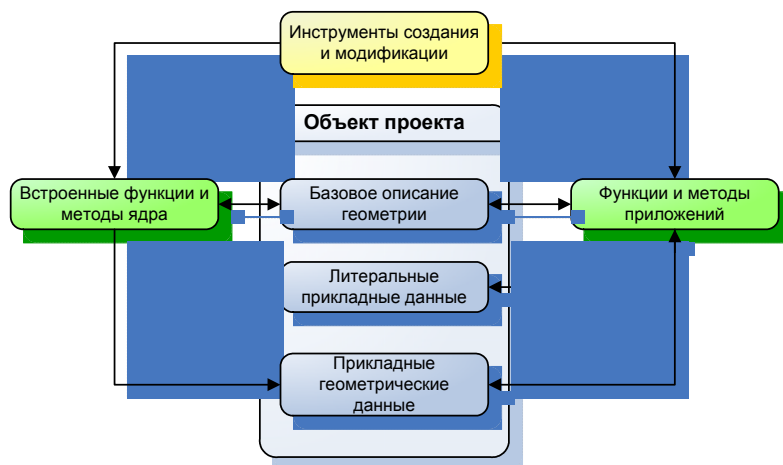


Рис.6 Структура работы с прикладными данными в полуактивной расширяемой системе.

Таким образом, изменения геометрической информации, описанной в прикладных данных, гарантированно происходят синхронно с преобразованиями геометрии основных объектов. Применение этого метода внедрения прикладных данных в базу данных проекта позволяют создать *полуактивную* расширяемую систему (рис. 6.) в которой вся

специфическая обработка литеральных прикладных данных, связанных с целевым назначением производится путем исполнения прикладных программ, а геометрические трансформации – могут быть выполнены автоматически головной системой.

Активные компоненты функциональной модели. Описанных выше ограничений лишен предлагаемый в настоящей работе способ расширения данных, который заключается не только в хранении в объекте прикладных данных, но и таблицы переопределенных базовых методов, автоматически вызываемых системой при совершении операций над объектом (рис.7). В том случае, если при разработке прикладного объекта метод не переопределен, то базовая система использует встроенные методы модификации, аналогично способам описанным ранее.

Таким образом, при всяком интерактивном или программном изменении, вносимом в базу данных объекта, их перевычисление может производиться в соответствии с поведенческим алгоритмом, заложенным разработчиком приложений, что обеспечивает расширяемость функциональности базовой системы и обеспечивает корректную модификацию всех компонент объекта. В таблице подмены методов, реализующих объект, могут быть переопределены не только методы изменяющие объект, но и, например, методы его отображения в зависимости от требуемого контекста, что принципиально расширяет возможности, предоставляемые разработчику прикладных систем.

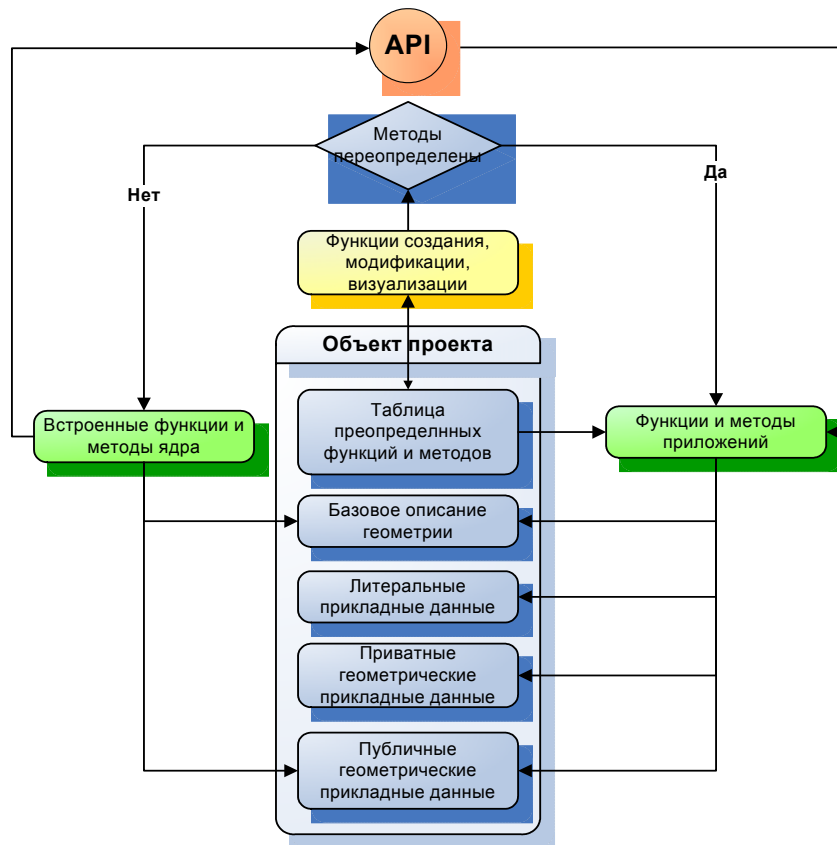


Рис.7 Структура работы с прикладными данными в активной расширяемой системе.

Наличие активные компонент в архитектуре прикладных систем позволяют сформировать наборы литеральных объектов, описывающих *ограничения и связи*, установленных между литералами и параметрами геометрических объектов. Каждый экземпляр установленной связи также содержит интервал возможных значений параметра.

В главе 3 приводится техническое описание программной реализации системы, основанной на предложенной архитектуре. Проектирование и

разработка системы велась исходя из следующих общих технических требований к внутренней организации программного кода:

- независимое изменение исходных кодов и отладку частей системы несколькими программистами одновременно
- возможность совместного использования значительной части исходных кодов, как самой системой, так и какими-либо иными уже существующими или будущими программными системами.
- возможность расширения функционала системы, без каких-либо изменений исходных текстов базовой системы и соответственно перекомпиляции базового исполняемого модуля.
- возможность простой установки и регистрации новых расширений в уже установленной у пользователя системы.
- как система, так и расширения должны быть защищены от возможных повреждений своих внутренних данных или нештатного поведения, в результате их несовместимости.

Учитывая достаточно большие объемы данных, хранимые системой в оперативной памяти, и на основе имеющейся информации по способам работы пользователей с системой (по опыту предыдущей версии системы), категорически неприемлемо аварийное завершение работы системы с потерей всех текущих данных, например, при возникновении нехватки памяти для завершения текущей операции. Следовательно, реализация системы спроектирована таким образом, что бы максимально защитить пользователя не только от потери, но и от нарушения целостности его данных при возникновении *не фатальных* исключительных ситуаций.

Система проектировалась с использованием необходимой декомпозиции проекта в целом на набор *слабо связанных* библиотек. Для поддержания

возможности расширения функционала и обеспечения гибкости конфигурирования системы под нужды различных категорий пользователей в системе объявлено несколько основных интерфейсов для *системных* и *прикладных* расширений. Реализовано несколько таких расширений в основном ориентированных на добавление чтения/записи дополнительных форматов хранения изображений, импорта/экспорта объектной модели в различные графические форматы, а также реализации специализированных прикладных объектов. Реализация системных расширений в виде динамически подгружаемых библиотек (DLL) обеспечивает возможность независимой их разработки, отладки и распространения при условии неизменности интерфейса "система-расширение". Базовая библиотека поддержки интерфейса пользователя содержит механизм проверки на совместимость версий системы и расширения, основанный на проверке номеров версии и номеров сборки (build) системы и загружаемого расширения, и базовые механизмы настройки графических элементов управления под нужды пользователя.

В главе 4 приводятся технические описания ряда прикладных систем, созданных на базе предлагаемой архитектуры и организации модели данных. Описаны прикладные системы для создания проектов архитектурных сооружений, конструирования мебели и торгового оборудования, моделирования кинематики, сложных трубопроводных коммуникаций. Дано подробное описание прототипа системы ассоциативного параметрического геометрического моделирования, созданной на основе внешнего геометрического решателя.

В **Заключении** сформулированы основные результаты работы.

1. На основе системного анализа определены принципы построения архитектуры САПР, обеспечивающей разработку автоматически взаимодействующих прикладных расширений базовой системы.
2. На основе этих принципов разработана архитектура САПР, обеспечивающая создание проблемно-ориентированных внешних приложений способных моделировать проектируемые объекты как в виде геометрических, так и в виде функциональных моделей и комбинаций этих описаний.
3. Определен набор базовых алгоритмов, обеспечивающий расширяемость проектной среды, возможность добавления описаний новых объектов в уже существующие модели и обеспечение взаимодействия этих объектов.
4. Разработаны эффективные методы представления, хранения, внедрения в проектную базу данных и управления разнородными данными, статическими ресурсами и активных компонентами проблемно-ориентированных САПР.
5. Выработаны требования и апробированы инструментальные средства проектирования и разработки активных компонент функционально-геометрических моделей.
6. На основе предложенной архитектуры реализована САПР bCAD и, на ее базе, функционально-геометрические модели для ряда прикладных областей.

Публикации: Материалы диссертации изложены в работах:

1. Малюх В.Н. Программируем САПР на Java. «САПР и Графика», М. Изд-во Компьютер Пресс, № 12, 1998, с 65-67
2. Малюх В.Н. Разработка специализированных приложений в САПР bCAD. «Автоматизация проектирования», М., изд. РАН, № 2, 2000 с15-18,
3. Бахтин И.Н., Малюх В.Н., Архитектура интерфейса прикладного программирования в САПР bCAD. "САПР и Графика" М. Изд-во Компьютер Пресс, №3, 2000, с 20-26.
4. Malukh V.N. Intelligent capabilities for small CAD systems: problems and perspectives. Сборник докладов на международной конференции isiCAD-2004, Новосибирск, июнь 2004, с. 311-312.
5. Malukh V.N., Nickitin A.G. Modern Architecture of light-weight CAD. Сборник докладов на международной конференции Graphicon -2005, Новосибирск, июнь 2005, с. 111-113.