

**Конспект лекций по курсу
ОСНОВЫ САПР**

Содержание

1	Основы теории проектирования	3
1.1	Методы проектирования.	3
1.2	Стадии проектирования	3
1.3	Аспекты проектирования	4
1.4	Блочнo - иерархический подход к проектированию	4
1.5	Стратегии проектирования	5
1.6	Алгоритм процесса проектирования	7
1.7	Типовые проектные процедуры	7
2	Обеспечения САПР	8
2.1	Программные обеспечения САПР	9
3	Математическое обеспечение САПР	12
3.1	Требования к математическому обеспечению	12
	Универсальность	12
	Алгоритмическая надежность	12
	Точность	13
	Затраты машинного времени	13
	Используемая память	13
3.2	Математическое моделирование объектов и устройств автоматизации в САПР	14
3.2.1.	Классификация математических моделей	14
3.2.2.	Математические модели на микро-, макро- и метауровнях	15
3.2.3.	Методика получения математических моделей объектов	19
4	Основы работы с системой КОМПАС.	19
4.1	Основные сведения о системе	19
4.2	Типы документов системы КОМПАС	20
4.3	Интерфейс системы	20
4.4	Базовые приемы работы	21
	Курсор и управление им	22
	Использование контекстных меню	22
4.5	Приемы создания объектов	22
	Параметры объектов	22
	Автоматическое и ручное создание объектов	26
	Прерывание команды	26
	Привязка	27
	Клавиатурная привязка	29
	Ортогональное черчение	29
4.6	Техмерное моделирование в КОМПАС	29
	Ограничения двумерного проектирования	29
	Переход на трехмерное моделирование	30

1 Основы теории проектирования

1.1 Методы проектирования.

Существует 2 основных метода проектирования:

1. Ремесленный
2. Чертежный

1. Сам процесс проектирования происходит в процессе изготовления. Детали и узлы проектируются по месту. Непосредственно проектирование из всего процесса вычленишь (выделить) практически невозможно. При таком методе проектирования время получения готового изделия достаточно высоко. В настоящее время этот метод применяется при изготовлении единичных (уникальных) изделий, а также в опытно – конструкторских работах.

2. Представляет собой предварительную проработку конструкции изделия, а также подготовку конструкторской и технологической документации. Этот метод наиболее широко распространен. Проектирование отделено от изготовления. За счет этого сокращается время на изготовление изделия, но увеличивается время на проектирование и подготовку производства. Данный метод применяется при производстве как серийных, крупносерийных и массовых изделий, так и может применяться и для производства единичных изделий.

При использовании чертежного метода резко сокращается время необходимое для изготовления изделий, но, в то же время, увеличивается срок подготовки производства. Очевидно, что сокращение сроков выпуска новой продукции ведет к уменьшению ее себестоимости. И если на технологических операциях возможно сокращение на секунды или максимум минуты, при использовании более совершенных технологий, то на подготовке производства (проектировании), при использовании средств автоматизации, можно сэкономить недели и даже месяцы. Особенно это важно при мелкосерийном, часто обновляющемся, гибком производстве.

1.2 Стадии проектирования

Прежде чем поступить к автоматизации проектирования необходимо выяснить, что есть такое проектирование, каковы его задачи и принципы. Этим как раз и занимается наука называемая теорией проектирования.

Проектирование технического объекта связано с созданием, преобразованием и представлением в принятой форме образа этого объекта. Образ объекта может создаваться в воображении человека в результате творческого процесса.

Любое проектирование начинается с возникновения осознанной потребности общества в изделии. Образ его постепенно преобразовываясь и претерпевая ряд существенных изменений в конце концов преобразуется в готовый материальный объект. Преобразование это проходит в несколько "этапов" (стадий). Возможно наличие следующих стадий:

- Выявление потребности общества в изделии (МАРКЕТИНГ)
- НИОКР и ОКР
- Техническое задание
- Техническое предложение
- Эскизный проект
- Технический проект
- Рабочий проект
- Технологический проект

1. Техническое задание.

В ходе работы на этой стадии готовится документ – «Техническое задание», где описываются основные требования к изделию, его общие характеристики, назначение и область использования.

2. На основании технического задания разрабатывается техническое предложение.

На этой стадии разрабатываются общие принципы функционирования изделий. Строится функциональные, структурные и кинематические схемы. Определяются различные математические зависимости. Производятся начальные проектировочные расчеты. Результатом этой стадии является подготовка плакатов, чертежей (ВО) изделия с возможными вариантами конструкций узлов, прорабатывается общая компоновка изделия.

3. Эскизный проект (в нек сл стадия м Мб опущена)

На этой стадии более подробно прорабатывается конструкция узлов изделия с возможными вариантами конфигурации топологии и размеров деталей. Результатом являются чертеж общего вида изделия, а также чертежи общих видов узлов. По рез-м эксплуатационного проекта выбирается окончательное использование всех узлов и деталей изделия.

4. Технический проект

Это окончательная проработка конструкций изделия. Готовятся наиболее полные чертежи общего вида, на которых ясна конфигурация каждой детали, производятся окончательные проектировочные и проверочные расчеты, определяются возможности изделия. Результат оформляется в виде чертежей общего вида стадии «технический проект» на изделие в целом и на узлы. Такие чертежи часто используются как сборочные чертежи рабочего проекта.

5. Рабочий проект

Данная стадия носит вспомогательный характер, т.к. сам процесс разработки конструкции уже окончен. На этой стадии производится детализировка узлов и изделия в целом, готовятся рабочие чертежи деталей и сборочные чертежи.

6. Технологический проект

Подготовка техпроцесса изготовления деталей и сборки изделия, а также подготовка технических заданий на технологическую оснастку.

Очевидно, что каждый этап представляет собой какую-либо информацию об объекте с соответствующей детализацией. В ходе проектирования от этапа к этапу меняется объем информации об изделии, Таким образом "проектирование" - это процесс преобразования информации об объекте. Здесь же можно отметить, что коль скоро проектирование - преобразование информации, то можно применить науку информатику. Т.е. данный процесс может быть автоматизирован.

1.3 Аспекты проектирования

На каждом этапе проектирования м существовать 3 аспекта:

1. Функциональный. Включает в себя разработку функциональных схем и проведение расчетов, касающихся функционирования изделия.

2. Конструкторский. Включает в себя изготовление чертежей и проведение расчетов, касающихся формы и размеров изделия, а также топология элементов (местоположение элементов в узле).

3. Технологический. Включает в себя возможность сборки и изготовление изделия.

Таким образом, процесс проектирования представляет собой многомерную сеть, координаты которой являются стадии, стратегии, аспекты и время.

1.4 Блочно - иерархический подход к проектированию

При проектировании сложных изделий зачастую одному человеку не под силу охватить объект целиком, кроме того это может быть связано с необходимостью иметь различные специальные знания. Для упрощения процесса проектирования и вовлечения большего числа проектировщиков в этот процесс необходимо разбиение объекта на части, снижая сложность до уровня, при котором этот объект может быть воспринят человеком. Такое разбиение называется "декомпозицией"

Декомпозиция – это разбиение объекта на некоторые части (блоки) (различные объекты по разным признакам), которые строятся в иерархическое дерево. Данный подход к проектированию получил название – «блочно - иерархический подход». (рисунок 1).

Блочно - иерархический подход к проектированию

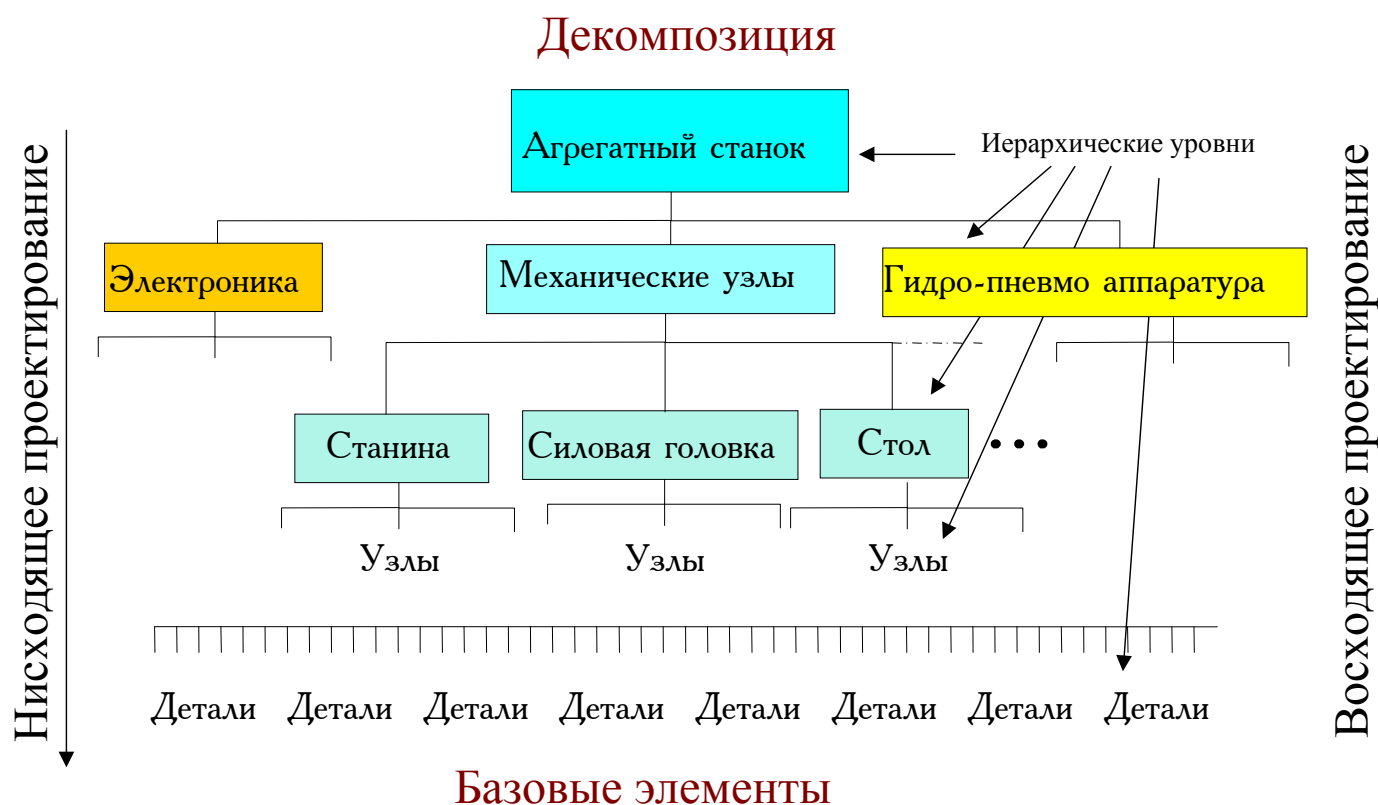


Рисунок 1 – Пример декомпозиции агрегатного станка

Процесс декомпозиции происходит в несколько этапов

На I этапе – проектируемый объект разбивается на части по функциональному признаку. Чаще всего это определяет предметное направление специалистов – проектировщиков. Каждый элемент разбивается на отдельные составляющие. Это происходит до тех пор, пока отдельный элемент не будет ясен целиком одному человеку (проектировщику). Элементы не подлежащие дальнейшему разбиению, называются «базовыми элементами».

Уровни разбиения называются «иерархическими уровнями».

Из схеме видно, что возможны два подхода к проектированию:

Проектирование сверху – вниз – «нисходящее» проектирование;

Проектирование снизу – вверх – «восходящее» проектирование.

При нисходящем проектировании не определены возможности элементов низших иерархических уровней. При восходящем – не определены требования к элементам низших иерархических уровней.

Пример нисходящего – проектирование уникальных изделий, которые чаще всего могут не иметь аналогов.

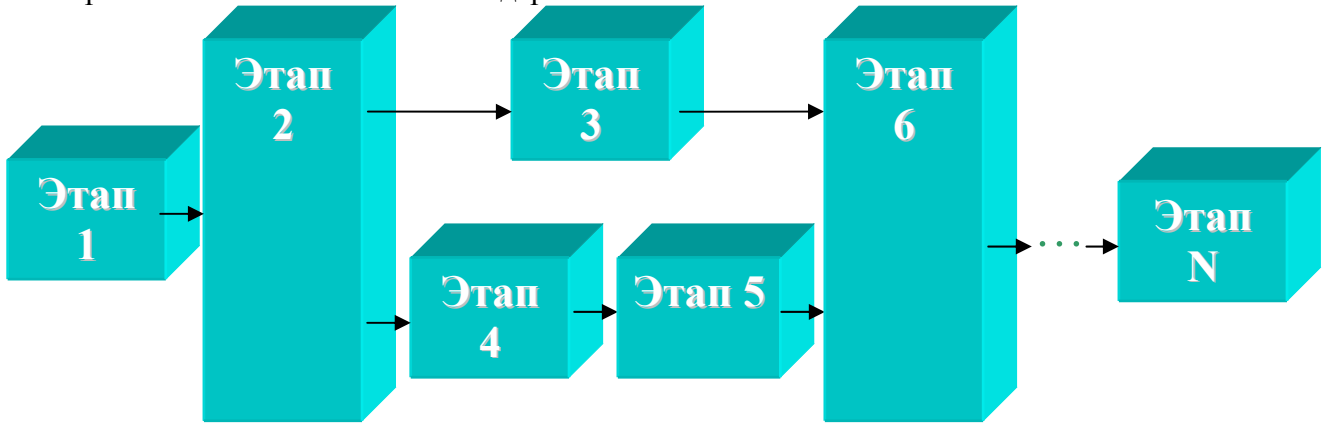
Пример восходящего – универсальное сборное приспособление.

1.5 Стратегии проектирования

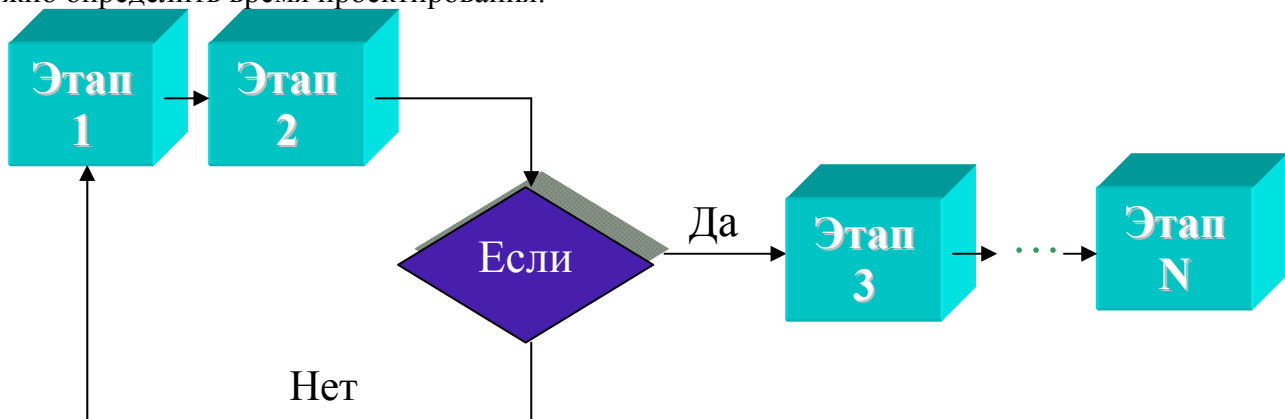
1. Линейная. Этапы следуют один за другим. Однако на каждом следующем этапе известны все результаты предыдущего этапа. Такую стратегию называют «идеальной». Недостаток – велико время на проектирование.



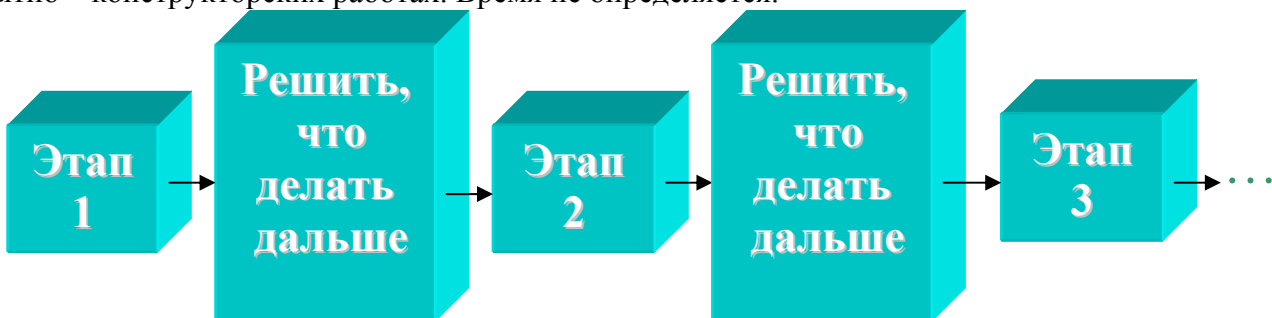
2.Разветвленная. Подобна линейной. Этапы следуют один за другим, но есть такие этапы, которые можно производить параллельно (//). Благодаря такой стратегии можно сократить время на проектирование за счет вовлечения большего числа проектировщиков. Недостаток – необходимо заранее правильно выявить наличие и содержание // этапов.



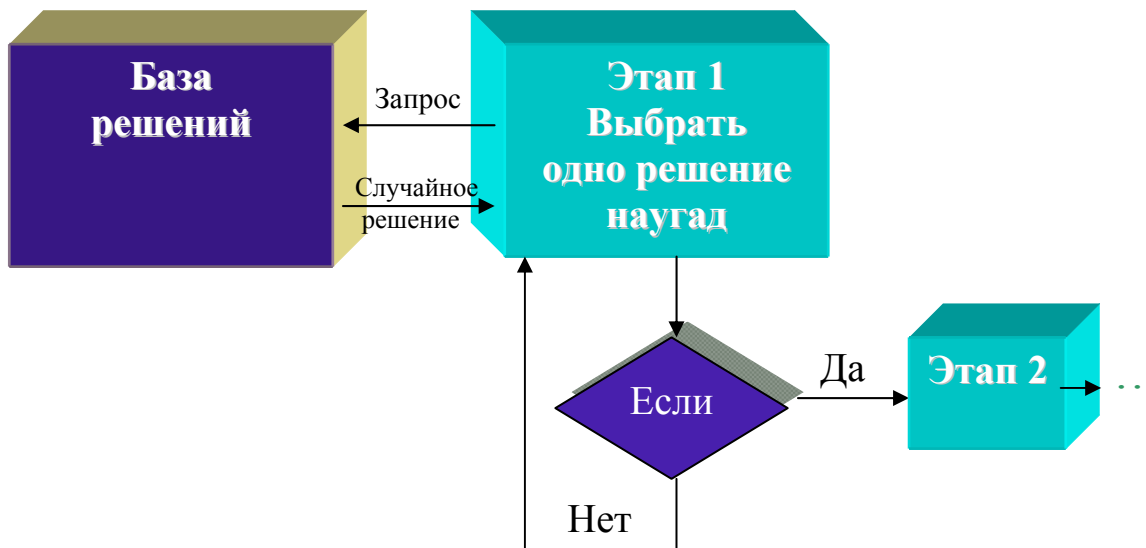
3.Циклическая. Имеет смысл, когда необходимо после выполнения какого-либо этапа оценить результат, выяснив корректность выполнения требований к изделию. Такая стратегия дает возможность оптимизировать. Недостаток – опасность заикливания. Для того, чтобы не происходило заикливание проектирования необходимо ввести предельное количество интеграций. Можно определить время проектирования.



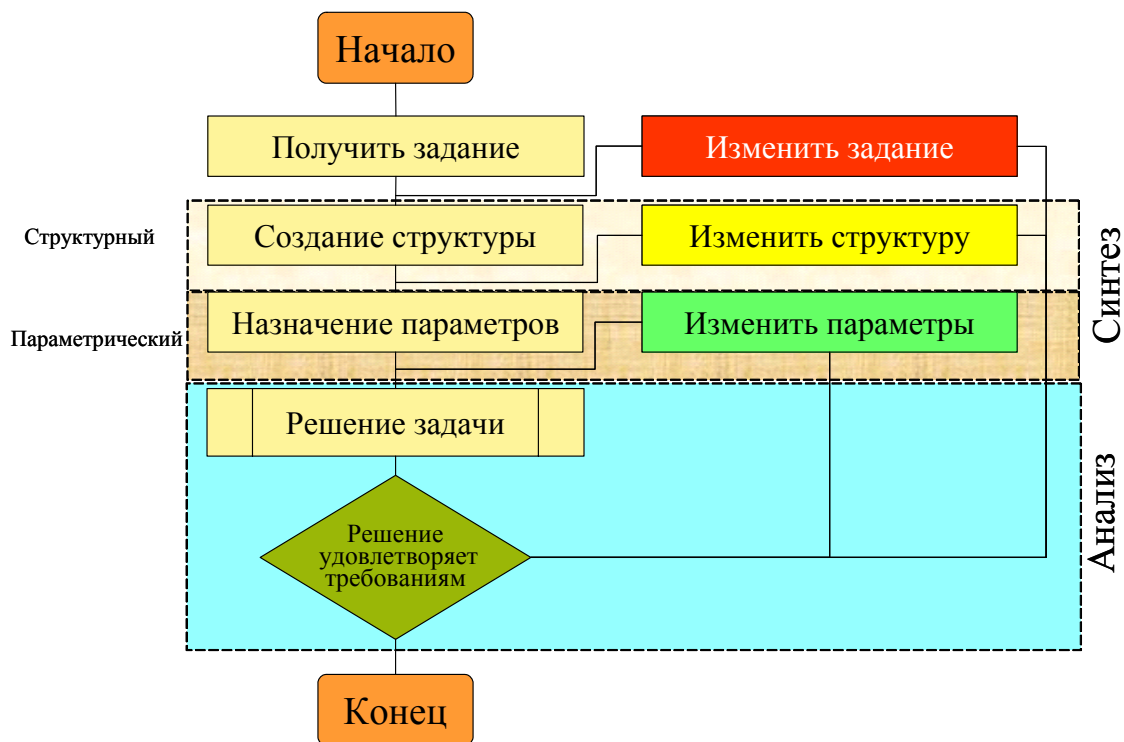
4.Адаптивная. Характеризуется тем, что после выполнения каждого этапа проектирования принимается решение, что делать дальше. Такая стратегия приносит положительные этапы при НИ и опытно – конструкторских работах. Время не определяется.



5.«Случайный поиск». Существует некая база решений и на определенном этапе проектирования выбирается решение из базы. Если решение удовлетворяет требованиям, процесс проектирования переходит в след этап. В противном случае выбирается след случай – вершина. Такая стратегия лучше прочих поддается автоматизации. Недостаток – нет уверенности в том, что в базе решений найдется нужное решение. М определить мах время проектирования.



1.6 Алгоритм процесса проектирования



1.7 Типовые проектные процедуры

Рассмотрим сначала ряд понятий, которые используются при разработке какого-либо объекта: «Проектное решение» – это промежуточное или конечное описание объекта проектирования необходимое и достаточное для рассмотрения и определения дальнейшего направления или окончания проектирования.

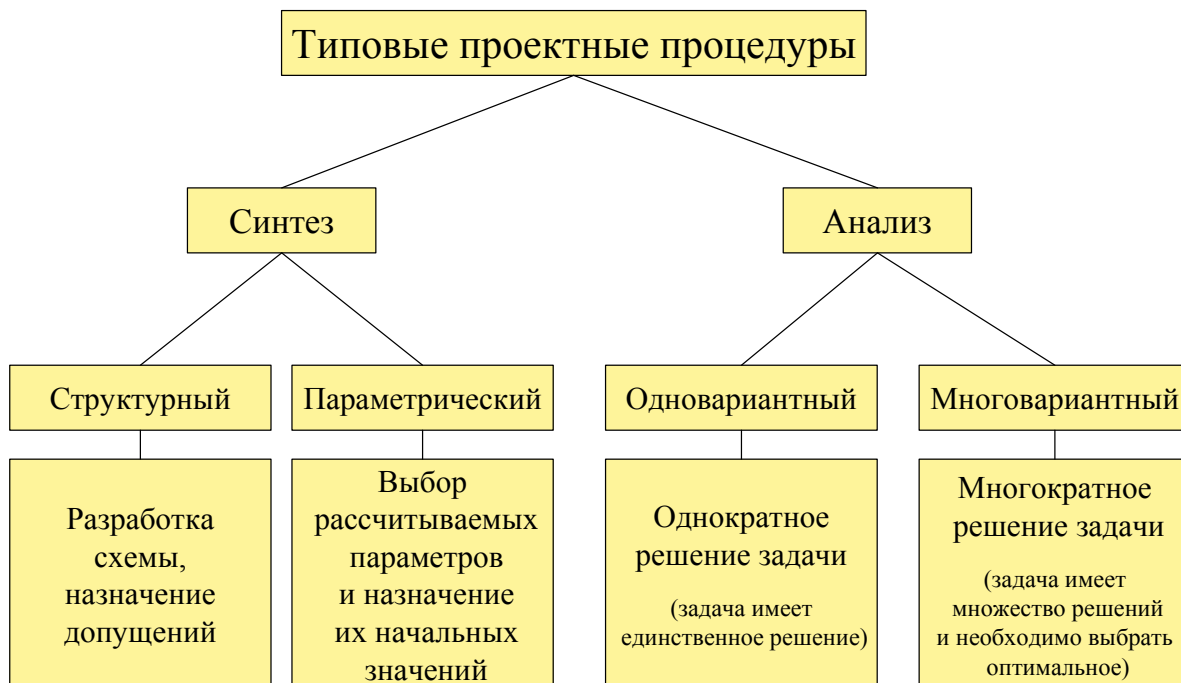
В свою очередь проектное решение или их совокупность представленные в заданной форме, в которых указаны все требования необходимые для создания объекта, называется "проектом".

«Проектной процедурой» называют формализованную совокупность действий, выполнение которых оканчивается проектным решением.

Например: поиск решения, корректировка, компоновка, оптимизация и т.п.

Проектная процедура, алгоритм которой остается неизменным для различных объектов проектирования или различных стадий проектирования одного и того же объекта, называется типовой или унифицированной

Представим классификацию типовых процедур в виде схемы.



Под синтезом понимается построение описания объекта показывающее его функционирование. Анализ - это возможностей функционирования объекта по заданному описанию.

2 Обеспечения САПР

Для функционирования САПР необходимо наличие 8 обеспечений:

1. **Техническое обеспечение.** Включает в себя все аппаратные средства, необходимые для выполнения задач автоматизированного проектирования.

2. **Математическое обеспечение.** Включает в себя математическое моделирование, методы создания мат моделей, методы решения дифференциальных и интегральных уравнений, методы обработки результатов экспериментов.

3. **Алгоритмическое обеспечение.** Тесно связано с математическим и включает в себя принципы построения алгоритмов, алгоритмы решения не типовых задач.

4. **Лингвистическое обеспечение.** Включает в себя все языковые средства автоматизированного проектирования, т е языки программирования (средства разработчика САПР) и языки проектирования (средства пользователя САПР), способы взаимодействия человека с ЭВМ.

5. **Программное обеспечение.** Включает в себя все программные средства обработки информации, начиная от общесистемных, заканчивая пакетами прикладных программ.

6. **Информационное обеспечение.** Включает в себя принципы структурирования и хранения информации, способы обмена информацией, организацию к доступа к информации, а также программные средства для выполнения указанных выше функций.

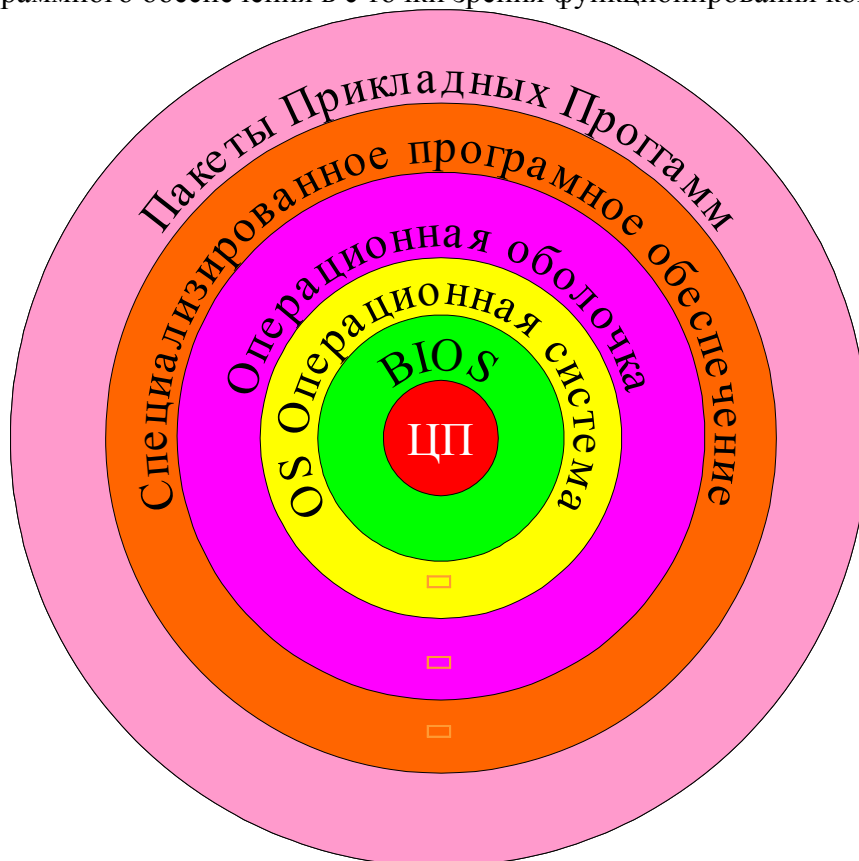
7. **Методическое обеспечение.** Включает в себя документацию о САПР, описывающую работу с конкретной системой проектирования, а также государственные, отраслевые стандарты и стандарты предприятий, регламентирующие работу с системами автоматизированного проектирования.

8. **Организационное обеспечение:** Включает в себя штатное расписание сотрудников, обеспечивающих работу систем автоматизированного проектирования, должностные инструкции, пароли и условия доступа к информации, распределение времени работы на автоматизированных рабочих местах.

2.1 Программные обеспечения САПР

Структура программного обеспечения

Классификацию программ дает соответствующий ГОСТ, который основан на направлениях работы и предметных областях САПР. Предлагаемая классификация не характеризует место программного обеспечения в с точки зрения функционирования компьютера и пользователя.



➤ **ЦП**- центральный процессор

➤ **BIOS** – базовая система ввода, вывода. Данная программа осуществляет связь центрального процессора со всеми периферийными устройствами. Определяет место положения устройств, их начальные настройки. Назначает адреса и номера запросов прерывания. Данная программа реализована в виде ППЗУ, т е перезаписываемое запоминающее устройство. Программа является уникальной для каждой группы системных плат.

➤ **OS** – операционная система – загружается на компьютере первой после инициализации BIOS и позволяет общаться с компьютером на языке символьных команд. Основная работа системы – распределение файлов в оперативной внешней памяти, структурирование информации на внешних носителях, запуск программы на выполнение, обеспечивает одноразовое хранение информации.

Существуют следующие виды ОС:

- MS DOS работает с оперативной памятью, объем 640 kb. Текущая выполняемая программа находится в опер памяти полностью. С помощью специальных драйверов ОС MSDOS может использовать больший объем памяти, но лишь для хранения временных данных. Процесс обработки в верхней памяти не происходит. Имеет адресацию (длину машинного слова) - 8 бит

- MS Windows 95/98/ME. ОС, которая используется для домашних компьютеров и характеризуется тем, использует всю имеющуюся на компьютере оперативную память. При загрузке Windows первоначально используются данные, полученные от BIOS, а затем система переназначает

адреса и прерывания для всех периферийных устройств. Это обеспечивает более быструю работу компьютера в целом, т.е. обращение к периферийным устройствам происходит минуя BIOS. Указанные ОС используют файловую систему FAT (File Allocation Table).

Файловая система FAT работает с единицами дискового пространства, называемыми кластер. Каждый кластер может включать один или несколько секторов жесткого диска (диск обычно разбит на сектора по 512 байт). Из чего следует, что минимальный размер кластера - 512 байт. Для хранения одного файла можно использовать один или несколько кластеров. Каждому кластеру диска в таблице FAT соответствует отдельная запись, которая либо указывает на следующий кластер файла, либо содержит метку конца файла. В составе каждого каталога хранятся имена входящих в него файлов. Вместе с именем файла хранится указатель на первый кластер этого файла. Помимо этого в каталоге хранится дата создания файла, его размер и атрибуты. Атрибуты могут указывать на то, что файл является скрытым, зарезервированным для использования операционной системой, требует архивирования (резервного копирования) или предназначен только для чтения.

FAT имеет следующие недостатки:

1. FAT16 идентифицирует записи, соответствующие дисковым кластерам, при помощи 16-разрядных чисел. Таким образом, в таблице можно разместить не более 65 536 записей ($=2^{16}$). А если учитывать то, что максимальный размер кластера - 32 Кбайта, то выходит, что максимальный раздел дискового тома - 2 Гбайта. FAT32 идентифицирует записи при помощи 32-разрядных чисел, что в значительной степени снижает этот недостаток (Формально максимальный объем для FAT32 - 4 Тбайта).

2. Для хранения ВСЕХ файловых атрибутов система FAT использует всего 1 байт. По этой причине в FAT нельзя хранить ни сведения о праве доступа к файлу, ни о его владельце.

3. При использовании FAT большой размер дискового тома означает большой размер кластера. Но один файл в FAT - это, как минимум один кластер (или целое число кластеров). Т.е. достаточно большое количество пространства попросту не используется.

4. Сведения о физическом расположении файлов хранятся в одном месте - таблице размещения файлов FAT, что: а) увеличивает вероятность повреждения и потери всей информации; б) снижает скорость поиска, т.к. для поиска определенного файла нужно обработать всю таблицу.

Нужно признать, что FAT16 создавалась давно, во времена MS-DOS и требованиям того времени вполне удовлетворяла.

- MS Windows NT4/2000/XP. Отличается от предыдущей тем, что используются так называемые NT - технологии, которые предназначены для обеспечения надежности хранения и поддержания целостности информации. ОС относится к профессиональным. Наряду с таблицей FAT, ОС Windows NT могут использовать файловую систему NTFS (New Technology File System). В отличие от FAT, NTFS система обеспечивает более высокую надежность защиты информации, жестко ограничивает разрешение на доступ к информации, что очень важно при работе глобальных и локальных сетях.

По сравнению с FAT файловая система NTFS обладает куда более сложной структурой и куда более широкими возможностями. В отличие от FAT, файловая система NTFS не хранит всю информацию о расположении файлов в одном месте. Вместо этого сведения о распределении дискового пространства между файлами сохраняются в составе специальных пакетов, которые могут располагаться в любом месте раздела. Структура каталогов NTFS также отличается от структуры каталогов FAT. Дисковые каталоги NTFS лучше приспособлены для осуществления поиска файлов, так как записи о файлах сохраняются с использованием бинарного дерева, а не простого линейного списка (как это было в FAT). Это означает, что для того, чтобы обнаружить файл, требуется проанализировать меньшее количество записей. Файловая система NTFS обладает встроенной поддержкой длинных имен и расширяемых файловых атрибутов. Благодаря этому разделы NTFS могут хранить информацию, связанную с защитой файлов, аудитом доступа к файлам, а также сведения, связанные с правами на владение файлами

Еще одна возможность NTFS- задание дисковой квоты. Она связанная с возможностью сохранения расширенного количества атрибутов файла и заключается она в том, что определенному пользователю можно назначить определенный размер дискового пространства, который он может

использовать для хранения своих файлов. При попытке сохранения файла, система анализирует размер всех файлов, которые уже принадлежат текущему пользователю по атрибуту "владелец" и сравнивает с назначенной дисковой квотой. Если остаток квоты достаточен для размещения файла, то будет выполнено сохранение, в противном случае система откажет в сохранении и выжаст сообщение "превышена дисковая квота".

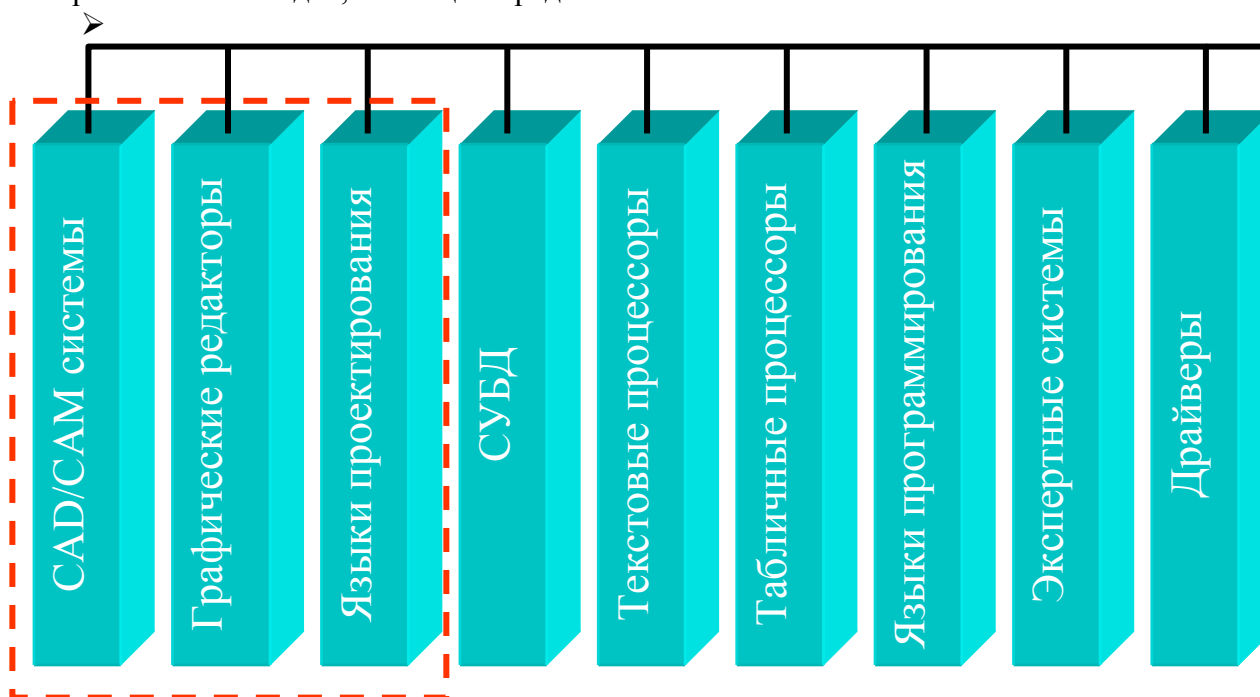
- OS/2 – ОС работает также со всей оперативной памятью, так же как и Windows поддерживает многозадачность, в отличии от ОС MS DOS. Работает на файловой системе FAT, не нашла широкого применения, т.к. набор программ под эту операционную систему довольно ограничен.

- UNIX – предназначен для обработки больших объемов информации, поэтому используется в компьютерах серверного исполнения, работающих с большим объемом информации. Сервера под управлением UNIX работают как сервера internet.

- LINUX – характеризуется тем, что является многопоточной, т.е. каждая запущенная программа работает как бы в своем пр-ве, т.е. данная ОС обеспечивает наличие многопоточных процессов.

- **Операционные (Файловые) оболочки** повышают удобство работы с файловыми системами (просмотр файлов и их атрибутов, запуск, редактирование и поиск файлов, выделение групп файлов, систематизирование размещения файлов и др.). К ним относятся такие программы как Norton Commander, Dos Navigator, Volkov Commander (для DOS). В операционных системах WINDOWS наряду со встроенной оболочкой Explorer (Проводник) используются такие программы как FAR, Total Commander, Directory Opus и некоторые другие.

- **Специализированные программы** – программы, предназначенные для выполнения конкретных типов задач, имеющих предметное назначение.



- CAD/CAM системы – предназначены для конструкции и тех проекта.
- Графические редакторы – программы, предназначенные для создания и редактирования векторной и растровой графики.

- Языки проектирования – спец языковые средства, предназначенные для обслуживания CAD/CAM систем. Могут быть визуальными и текстовыми.

- СУБД – система управления базами данных (БД) – предназначены для разработки и хранения БД.

- Текстовые процессоры систем – предназначены для набора редактирования и верстки текстов.

- Табличный процессор – система для разработки и обработки табличных алгоритмов.

- Языки программирования – это средства, предназначенные для разработки приложений программ. Различаются – языки высокого и широкого уровня. Высокого – более удобны для программирования. Языки низкого уровня менее удобны для программирования, но полученный машинный код работает быстрее (более экономично).
- Экспертные системы – системы, ориентируются на конкретную предметную область и позволяет принимать решения на уровне специалиста – эксперта, оперируя при этом базой знаний разработанной на основе знаний и правил описанных специалистами в предметной области и инженерии знаний .
- Драйверы – программы, предназначенные для сопровождения периферийных устройств.
- Пакеты прикладных программ – это программы, разработанные с помощью специализированных программ – обеспечения. Какой-либо классификации не подлежат.
- Математическое обеспечение автоматизации проектирования

3 Математическое обеспечение САПР

Математическое обеспечение САПР состоит из математических моделей объектов проектирования, методов и алгоритмов выполнения проектных операций и процедур. В математическом обеспечении САПР можно выделить специальную часть, в значительной мере отражающую специфику объекта проектирования, физические и информационные особенности его функционирования и тесно привязанную к конкретным иерархическим уровням (эта часть охватывает математические модели, методы и алгоритмы их получения, методы и алгоритмы одновариантного анализа, а также большую часть используемых алгоритмов синтеза), и инвариантную часть, включающую в себя методы и алгоритмы, слабо связанные с особенностями математических моделей и используемые на многих иерархических уровнях (это методы и алгоритмы многовариантного анализа и параметрической оптимизации).

3.1 Требования к математическому обеспечению

Свойства математического обеспечения (МО) оказывают существенное, а иногда и определяющее влияние на возможности и показатели САПР. При выборке и разработке моделей, методов и алгоритмов необходимо учитывать требования, предъявляемые к МО в САПР. Рассмотрим основные из них.

Универсальность

Под универсальностью МО понимается его применимость к широкому классу проектируемых объектов. Одно из отличий расчетных методов в САПР от ручных расчетных методов - высокая степень универсальности. Например, в подсистеме схемотехнического проектирования САПР ИЭТ используются математические модели транзистора, справедливые для любой области работы (активной, насыщения, отсечки, инверсной активной), а методы получения и анализа моделей применимы к любой аналоговой или переключательной схеме на элементах из разрешенного списка; в подсистеме структурного проектирования САПР ЭВМ используются модели и алгоритмы, позволяющие исследовать стационарные и нестационарные процессы переработки информации при произвольных законах обслуживания в устройствах ВС и при произвольных входных потоках. Высокая степень универсальности МО нужна для того, чтобы САПР была применима к любым или большинству объектов, проектируемых на предприятии.

Алгоритмическая надежность (адекватность)

Методы и алгоритмы, не имеющие строгого обоснования, называют эвристическими. Отсутствие четко сформулированных условий применимости приводит к тому, что эвристические методы могут использоваться некорректно. В результате либо вообще не будет получено решение (например, из-за отсутствия сходимости), либо оно будет далеким от истинного. Главная неприятность заключается в том, что в распоряжении инженера может не оказаться данных, позволяющих определить, корректны или нет полученные результаты. Следовательно, возможна ситуация, когда неверное решение будет использоваться в дальнейшем как правильное.

Свойство компонента МО давать при его применении в этих условиях правильные результаты называется алгоритмической надежностью (адекватностью). Степень универсальности характеризуется заранее оговоренными ограничениями, а алгоритмическая надежность - ограничениями, заранее не выявленными и, следовательно, не оговоренными.

Количественной оценкой алгоритмической надежности служит вероятность получения правильных результатов при соблюдении оговоренных ограничений на применение метода. Если эта вероятность равна единице или близка к ней, то говорят, что метод алгоритмически надежен.

Применение алгоритмичности ненадежных методов в САПР нежелательно, хотя и допустимо в случаях, когда неправильные результаты легко распознаются.

С проблемой алгоритмической надежности тесно связана проблема обусловленности математических моделей и задач. О плохой обусловленности говорят в тех случаях, когда малые погрешности исходных данных приводят к большим погрешностям результатов. На каждом этапе вычислений имеются свои промежуточные исходные данные и результаты, свои источники погрешностей. При плохой обусловленности погрешности могут резко возрасти, что может привести как к снижению точности, так и к росту затрат машинного времени.

Точность

Для большинства компонентов МО важным свойством является точность, определяемая по степени совпадения расчетных и истинных результатов. Алгоритмически надежные методы могут давать различную точность. И лишь в тех случаях, когда точность оказывается хуже предельно допустимых значений или решение вообще невозможно получить, говорят не о точности, а об алгоритмической надежности.

В большинстве случаев решение проектных задач характеризуется: совместным использованием многих компонентов МО, что затрудняет определение вклада в общую погрешность каждого из компонентов;

векторным характером результатов (например, при анализе находят вектор выходных параметров, при оптимизации - координаты экстремальной точки), т.е. результатом решения является значение не отдельного параметра, а многих параметров.

В связи с этим оценка точности производится с помощью специальных вычислительных экспериментов. В этих экспериментах используются специальные задачи, называемые тестовыми. Количественная оценка погрешности результата решения тестовой задачи есть одна из норм вектора относительных погрешностей: m -норма или l -норма, где l - относительная погрешность определения j -го элемента вектора результатов; m - размерность этого вектора.

Затраты машинного времени

Универсальные модели и методы характеризуются сравнительно большим объемом вычислений, растущим с увеличением размерности задач. Поэтому при решении большинства задач в САПР *затраты машинного времени* значительны. Обычно именно затраты машинного времени являются главным ограничивающим фактором при попытках повысить сложность проектируемых на ЭВМ объектов и тщательность их исследования. Поэтому требование экономичности по затратам машинного времени - одно из основных требований к МО САПР.

При использовании в САПР многопроцессорных ВС уменьшить время счета можно с помощью параллельных вычислений. В связи с этим один из показателей экономичности МО - его приспособленность к распараллеливанию вычислительного процесса.

В САПР целесообразно иметь библиотеки с наборами моделей и методов, перекрывающими потребности всех пользователей САПР.

Используемая память

Затраты памяти являются вторым после затрат машинного времени показателем экономичности МО. Они определяются длиной программы и объемом используемых массивов данных. Несмотря на значительное увеличение емкости оперативной памяти в современных ЭВМ, требование экономичности по затратам памяти остается актуальным. Это связано с тем, что в

мультипрограммном режиме функционирования ЭВМ задача с запросом большого объема памяти получает более низкий приоритет и в результате время ее пребывания в системе увеличивается.

Улучшить экономичность по затратам оперативной памяти можно путем использования внешней памяти. Однако частые обмены данными между оперативной памятью и внешней могут привести к недопустимому росту затрат машинного времени. Поэтому при больших объемах программ и массивов обрабатываемой информации целесообразно использовать МО, допускающее построение оверлейных программных структур и реализующее принципы диакоптической обработки информации.

3.2 Математическое моделирование объектов в САПР

3.2.1. Классификация математических моделей

Рассмотрим основные признаки, классификации и типы ММ, применяемые в САПР.

По характеру отображаемых свойств объекта ММ делятся на структурные и функциональные.

Структурные ММ предназначены для отображения структурных свойств объекта. Различают структурные ММ топологические и геометрические.

В топологических ММ отображаются состав и взаимосвязи элементов. Их чаще всего применяют для описания объектов, состоящих из большого числа элементов, при решении задач привязки конструктивных элементов к определенным пространственным позициям (например, задачи компоновки оборудования, размещения деталей, трассировки соединений) или к относительным моментам времени (например, при разработке расписаний, технологических процессов). Топологические модели могут иметь форму графов, таблиц (матриц), списков и т.п.

В геометрических ММ отображаются свойства объектов, в них дополнительно к сведениям о взаимном расположении элементов содержатся сведения о форме деталей. Геометрические ММ могут выражаться совокупностью уравнений линий и поверхностей; совокупностью алгебраических соотношений, описывающих области, составляющие тело объекта; графами и списками, отображающими конструкции из типовых конструктивных элементов, и т.п. Геометрические ММ применяют при решении задач конструирования в машиностроении, приборостроении, радиоэлектронике, для оформления конструкторской документации, при задании исходных данных на разработку технологических процессов изготовления деталей. Используют несколько типов геометрических ММ.

Функциональные ММ предназначены для отображения физических или информационных процессов, протекающих в объекте при его функционировании или изготовлении. Обычно функциональные ММ представляют собой системы уравнений, связывающих фазовые переменные, внутренние, внешние и выходные параметры.

По степени детализации описания в пределах каждого иерархического уровня выделяют полные ММ и макромоделли.

Полная модель - эта модель, в которой фигурируют фазовые переменные, характеризующие состояния всех имеющихся межэлементных связей (т.е. состояние всех элементов проектируемого объекта).

Макромодель - ММ, в которой отображаются состояния значительно меньшего числа межэлементных связей, что соответствует описанию объекта при укрупненном выделении элементов.

По способу представления свойств объекта функциональные ММ делятся на аналитические и алгоритмические.

Аналитические ММ представляют собой явные выражения выходных параметров как функций входных и внутренних параметров.

Алгоритмические ММ выражают связи выходных параметров с параметрами внутренними и внешними в форме алгоритма.

Имитационная ММ - это алгоритмическая модель, отражающая поведение исследуемого объекта во времени при задании внешних воздействий на объект.

3.2.2. Математические модели на микро-, макро- и метауровнях

Описания технических объектов должны быть по сложности согласованы с возможностями восприятия человеком и с возможностями ЭВМ оперировать описаниями моделей в процессе их преобразования при проектировании. Однако выполнить это требование в рамках некоторого единого описания, не расчлняя его на отдельные составные части, удастся лишь для простых изделий. Как правило, требуется структурирование описаний и соответствующее расчленение представлений о проектируемых объектах на иерархические уровни и аспекты. Это позволяет распределять работы по проектированию сложных объектов между подразделениями проектной организации, что способствует эффективности и производительности труда проектировщиков.

Использование принципов блочно-иерархического подхода к проектированию структур математических моделей проектируемых объектов позволяет формализовать процесс их написания. Количество иерархических уровней при моделировании определяется сложностью проектируемых объектов и возможностью средств проектирования. Однако иерархические уровни большинства предметных областей можно отнести к одному из трех обобщенных уровней, называемых далее микро-, макро- и метауровнями.

В зависимости от места в иерархии описания математические модели делятся на ММ, относящиеся к микро-, макро- и метауровням.

Особенностью ММ на микроуровне является отражение физических процессов, протекающих в непрерывном пространстве и времени. Типичные ММ на микроуровне - дифференциальные уравнения в частных производных (ДУЧП). В них независимыми переменными являются пространственные координаты и время. С помощью этих уравнений рассчитываются поля механических напряжений и деформаций, электрические потенциалы и напряжения, давления и температуры и т.п. Возможности применения ММ в ДУЧП ограничены отдельными деталями, попытки анализировать с их помощью процессы в многокомпонентных средах, сборочных единицах, электронных схемах не могут быть успешными из-за чрезмерного роста затрат машинного времени и памяти.

На макроуровне используют укрупненную дискретизацию пространства по функциональному признаку, что приводит к представлению ММ на этом уровне в виде систем обыкновенных дифференциальных уравнений (ОДУ). В этих уравнениях независимой переменной является время t , а вектор зависимых переменных составляют фазовые переменные, характеризующие состояние укрупненных элементов дискретизированного пространства. Такими переменными являются силы и скорости в механических системах, напряжения и токи в электрических системах, давления и расходы жидкостей и газов в гидравлических и пневматических системах и т.п. Системы ОДУ являются универсальными моделями на макроуровне, пригодными для анализа как динамических, так и установившихся состояний объектов. Модели для установившихся режимов можно также представить в виде систем алгебраических уравнений. Порядок системы уравнений зависит от числа выделенных элементов объекта. Если порядок системы приближается к 10000, то оперирование моделью становится затруднительным и поэтому необходимо переходить к представлениям на метауровне.

На метауровне в качестве элементов принимают достаточно сложные совокупности деталей. Метауровень характеризуется большим разнообразием типов используемых ММ. Для многих объектов ММ на метауровне по-прежнему представляются системами ОДУ. Однако так как в моделях не описываются внутренние фазовые переменные элементы, а фигурируют только фазовые переменные, относящиеся к взаимным связям элементов, укрупненное представление элементов на метауровне означает получение ММ приемлемой размерности для существенно более сложных объектов, чем размерность ММ на макроуровне.

В ряде предметных областей удается использовать специфические особенности функционирования объектов для упрощения ММ. Примером являются электронные устройства цифровой автоматики, в которых возможно применять дискретное представление таких фазовых переменных, как напряжения и токи. В результате ММ становится системой логических уравнений, описывающих процессы преобразования сигналов. Такие логические модели существенно более экономичны, чем модели электрические, описывающие изменения напряжений и токов как непрерывных функций времени.

Применение тензорных представлений об объектах проектирования дает возможность использовать для получения ММ сложных технических систем методы диакоптики.

Исследование сложных систем по частям реализуется в диакоптических методах исследования. Отличие диакоптического подхода проектирования от блочно-иерархического заключается в том, что диакоптика основана на использовании структурных особенностей анализируемых схем и выражающих их матриц, а не на принятии каких-либо упрощающих допущений. В диакоптических методах производится расчленение математических моделей на части, исследуемые самостоятельно.

Расчленение математических моделей на части позволяет упорядочить и минимизировать количество обменов информацией между оперативной и внешней памятью при анализе сложных систем, а также выбирать для исследования каждой части наиболее выгодные режимы анализа. Эти обстоятельства делают диакоптические методы экономичными по затратам машинного времени и оперативной памяти.

Макромоделирование лежит в основе направления, связанного с рациональным выбором математических моделей элементов при построении математической модели системы. Макромоделирование реализует возможность использования при анализе одного и того же объекта нескольких моделей, различающихся сложностью, точностью и полнотой отображения свойств объекта, трудоемкостью требующихся вычислений и т.п.

При макромоделировании должны выполняться условия:

1) адекватности модели (выполнение данного условия требует от инженера учета целей решения каждой конкретной задачи и степени влияния параметров выделяемых элементов на результаты решения этой задачи);

2) большей экономичности создания макромоделей элементов и их дальнейшего использования по сравнению с решением задачи на основе полной математической модели (обычно это условие выполняется при использовании макромоделей для элементов типовых или, по крайней мере, часто встречающихся в данной системе);

Событийность анализа заключается в том, что при имитации процессов, протекающих в исследуемом объекте, в каждый момент модельного времени вычисления проводятся только для небольшой части математической модели объекта. Эта часть включает в себя те элементы, состояние которых на очередном временном шаге может измениться. Использование принципа событийности существенно повышает экономичность анализа на функционально-логическом и системном уровнях проектирования.

Рациональное использование эвристических способностей человека в интерактивных процедурах позволяет инженеру вмешиваться в ход вычислений и выбирать наиболее перспективные продолжения на основе эвристических оценок. Это выгодно во всех тех проектных процедурах, в которых следование только формальным критериям выбора дальнейших действий связано с чрезмерными затратами машинного времени. При исследовании сложных элементов и устройств автоматизации часто используют методы многовариантного анализа и теорию чувствительности.

Основными видами многовариантного анализа в задачах проектирования являются анализы чувствительности и статистический.

Анализ чувствительности

Цель анализа чувствительности - определение коэффициентов чувствительности, называемых также коэффициентами влияния:

$$\begin{aligned} a_{ji} &= \delta Y_j / \delta x_i ; \\ b_{ji} &= a_{ji} x_{iном} / Y_{iном} \end{aligned} \quad (4.2)$$

где a_{ji} и b_{ji} - абсолютный и относительный коэффициенты чувствительности выходного параметра y_j к изменениям внутреннего параметра x_i ; $y_{iном}$ и $x_{iном}$ - номинальные значения параметров y_j и x_i . Результаты анализа чувствительности m выходных параметров к изменениям n внутренних параметров представляют собой mn коэффициентов чувствительности, составляющих матрицу абсолютной или относительной чувствительности.

Анализ чувствительности применяется, если параметры X и Q можно считать непрерывными величинами, а параметры y_j являются дифференцируемыми функциями своих аргументов X_i и $q_{кном}$.

Результаты анализа чувствительности используются при решении таких важных задач, как параметрическая оптимизация, расчет допусков, оценка точности выходных параметров. Именно по значениям коэффициентов чувствительности разработчик отделяет существенно влияющие параметры от мало влияющих, определяет направления изменений внутренних параметров для улучшения выходных параметров, оценивает допустимые отклонения параметров X и Q для выполнения точностных требований к параметрам Y .

В ряде случаев для получения результатов математических экспериментов используют *метод приращений*. Это основной метод анализа чувствительности в инвариантном МО САПР. Метод приращений есть метод численного дифференцирования зависимости

$$Y = F(X, Q).$$

Алгоритм метода приращений включает в себя $(n + 1)$ -кратное обращение к модели для вычисления Y , где n - количество варьируемых параметров, т.е. таких параметров (q_k), влияние которых на Y исследуется. В первом варианте задаются номинальные значения аргументов и, следовательно, результатом обращения к модели будет номинальное значение

$$Y_{ном} = (Y1_{ном}, Y2_{ном}, \dots, Ym_{ном})$$

вектора Y . В очередном $(i + 1)$ -м варианте среди оставшихся n вариантов задается отклонение Δx_i от номинального значения только по одному из варьируемых параметров. В результате выполнения $(i + 2)$ -го варианта получают для вектора Y значение

$$Y_i = (y1_i, y2_i, \dots, ym_i),$$

по которому оценивается очередной i -й столбец матрицы абсолютной чувствительности

$$A_i = (Y_i - Y_{ном}) / \Delta x_i$$

Основное достоинство метода приращений - его универсальность: метод применим к любым непрерывным математическим моделям.

Однако у метода приращений имеются и существенные недостатки: невысокая точность, что характерно для операций численного дифференцирования; сравнительно большая трудоемкость вычислений. Трудоемкость вычислений оценивается количеством обращений к модели, так как объем вычислений в алгоритмических моделях обычно велик и заметно превышает трудоемкость выполнения процедур по обработке результатов обращений к моделям. В методе приращений требуется $n + 1$ вариант обращения к модели.

Прямой и вариационный методы. Эти методы анализа чувствительности менее универсальны, чем метод приращений, но позволяют повысить точность или снизить затраты машинного времени. Они основаны на интегрировании специальных систем обыкновенных дифференциальных уравнений, относятся к специальному математическому обеспечению.

Регрессионный метод. В регрессионном методе анализа чувствительности коэффициенты чувствительности отождествляются с коэффициентами регрессии, рассчитываемыми в процессе статистического анализа по методу Монте-Карло. Этот метод требует выполнения очень большого объема вычислений; его применение выгодно, если в каком-либо маршруте проектирования нужно решать задачи как статистического анализа, так и анализа чувствительности. Тогда затраты времени, дополнительные к затратам на статистический анализ, будут пренебрежимо малы.

Статистический анализ

Цель статистического анализа - получение оценок рассеяния выходных параметров Y и вероятностей выполнения заданных условий работоспособности для проектируемого объекта. В случае объектов типа систем массового обслуживания сами выходные параметры имеют вероятностный смысл, тогда цель статистического анализа - расчет таких параметров. Причинами рассеяния выходных параметров Y являются нестабильность внешних параметров Q и случайный характер внутренних параметров X . Результатами статистического анализа могут быть гистограммы выходных параметров, оценки математических ожиданий M_j и среднеквадратичных отклонений σ_j каждого из выходных параметров y_j от номинальных значений, оценки коэффициентов корреляции Γ_{ji} между параметрами y_j и x_i , а также выходные параметры систем массового обслуживания. В качестве исходных данных фигурируют статистические сведения о рассеянии внутренних

параметров и данные *технического задания* о допустимых диапазонах изменения или законах распределения внешних параметров.

Статистический анализ исключительно важен, поскольку его результаты позволяют прогнозировать процент выхода годных изделий при их серийном изготовлении, т.е. оценить серийнопригодность проектируемого объекта. Если в исходных данных отразить старение внутренних параметров - их изменение в процессе эксплуатации и хранения под действием различных физико-химических факторов, то результаты статистического анализа можно непосредственно использовать для оценки надежности.

Наибольшее распространение в САПР при статистическом анализе получили методы наихудшего случая и статистических испытаний.

Метод наихудшего случая. Этот метод служит для определения диапазонов возможного рассеяния выходных параметров без оценки плотности распределения этих параметров.

Пусть на некоторый выходной параметр y задано условие работоспособности в виде $y < TT$. Тогда интерес представляет верхняя граница диапазона рассеяния, так как большие значения y наиболее опасны с точки зрения невыполнения условия работоспособности. Верхняя граница диапазона рассеяния достигается в наихудшем случае, когда все аргументы функциональной зависимости $y = f(X)$ принимают самые неблагоприятные значения. Самым неблагоприятным значением аргумента X_i будет максимально возможное значение $X_{imax} = x_{ином} + x_i$ при выполнении условий $y < TT$ и $dy/dxi > 0$. Так же самым неблагоприятным значением аргумента X_i будет минимальное значение $X_{imin} = x_{ином} - x_i$, если $y < TT$ $dy/dxi < 0$. Здесь x_i - допуск на внутренний параметр X_i . При этом предполагается, что знаки коэффициентов влияния остаются неизменными в пределах рассматриваемой области.

Алгоритм метода наихудшего случая включает в себя следующие этапы:

1) Анализ чувствительности, в результате которого определяются коэффициенты чувствительности dy/dxi .

2) Задание параметрам X_i самых неблагоприятных значений.

3) Расчет выходных значений параметров при неблагоприятных внутренних.

Каждому выходному параметру соответствует свой наихудший случай. Если объект характеризуется m выходными и n внутренними параметрами, то этапы 2 и 3 алгоритма повторяются m раз и всего требуется выполнить $m + n + 1$ вариантов обращения к модели объекта. Преимущество метода наихудшего случая в том, что для его применения не требуется знания законов распределения внутренних параметров. Достаточно знать лишь допуски x_i . Недостаток метода в том, что результаты анализа в наихудшем случае могут ввести в заблуждение пользователя относительно реального рассеяния выходных параметров.

Метод Монте-Карло (метод статистических испытаний). Этот метод позволяет получить более полные статистические сведения о выходных параметрах исследуемого объекта. Алгоритм метода статистических испытаний включает в себя следующие основные этапы:

1) Задание значений внутренних и внешних параметров (аргументов зависимости Y от X и Q в очередном статистическом испытании).

2) Расчет Y .

3) Накопление статистических сумм.

4) Обработка накопленных сумм для получения результатов статистического анализа.

Операторы 1-3 выполняются в каждом испытании и могут быть распараллелены. Оператор 4 завершает статистический анализ. Задание значений случайных параметров выполняется в соответствии с их законами распределения. Подпрограммы выработки псевдослучайных чисел для величин, распределенных по таким законам, как нормальный, равномерный, экспоненциальный, имеются в составе общего программного обеспечения большинства ЭВМ. Однако в практике проектирования могут встретиться задачи с коррелированными исходными данными, имеющими любое распределение. Часто исходные данные получают как результаты измерений параметров на партии изделий и представляют в виде гистограмм. Тогда алгоритм задания случайных значений параметров целесообразно построить на основе следующего преобразования:

$$X = 0(AZ),$$

где O - оператор преобразования значений нормально распределенных величин в значения внутренних параметров, имеющих заданные распределения;

Z - реализация n -мерного некоррелированного случайного вектора, элементы которого имеют нормированное нормальное распределение, т.е. характеризуются нулевыми математическими ожиданиями и единичными дисперсиями;

X - реализация n -мерного случайного вектора внутренних параметров в очередном статистическом испытании;

A - матрица преобразования вектора Z в нормальный вектор с коррелированными элементами.

3.2.3. Методика получения математических моделей объектов

В общем случае процедура получения математических моделей объектов включает в себя следующие операции:

1) Выбор свойств объекта, которые подлежат отражению в модели. Этот выбор основан на анализе возможных применений модели и определяет степень универсальности ММ.

2) Сбор исходной информации о выбранных свойствах объекта. Источниками сведений могут быть опыт и знания инженера, разрабатывающего модель, научно-техническая литература, прежде всего справочная, описания прототипов - имеющихся ММ для элементов, близких по свойствам к исследуемому, результаты экспериментального измерения параметров и т.п.

3) Синтез структуры ММ. Структура ММ - общий вид математических соотношений модели без конкретизации числовых значений фигурирующих в них параметров. Структура модели может быть представлена также в графической форме, например в виде эквивалентной схемы или графа. Синтез структуры - наиболее ответственная и с наибольшим трудом подлежащая формализации операция.

4 Основы работы с системой КОМПАС.

4.1 Основные сведения о системе

Основная задача, решаемая системой КОМПАС - моделирование изделий с целью существенного сокращения периода проектирования и скорейшего их запуска в производство. Эти цели достигаются благодаря возможностям

- быстрого получения конструкторской и технологической документации, необходимой для выпуска изделий (сборочных чертежей, спецификаций, детализовок и т.д.),
- передачи геометрии изделий в расчетные пакеты,
- передачи геометрии в пакеты разработки управляющих программ для оборудования с ЧПУ,
- создания дополнительных изображений изделий (например, для составления каталогов, создания иллюстраций к технической документации и т.д.).

Основные компоненты КОМПАС - собственно система трехмерного твердотельного моделирования, чертежно-графический редактор и модуль проектирования спецификаций.

Система трехмерного твердотельного моделирования предназначена для создания трехмерных ассоциативных моделей отдельных деталей и сборочных единиц, содержащих как оригинальные, так и стандартизованные конструктивные элементы. Параметрическая технология позволяет быстро получать модели типовых изделий на основе однажды спроектированного прототипа. Многочисленные сервисные функции облегчают решение вспомогательных задач проектирования и обслуживания производства.

Система КОМПАС-График предназначена для автоматизации проектно-конструкторских работ в различных отраслях деятельности. Он может успешно использоваться в машиностроении, архитектуре, строительстве, составлении планов и схем - везде, где необходимо разрабатывать и выпускать чертежную и текстовую документацию.

Совместно с любым компонентом КОМПАС может использоваться модуль проектирования спецификаций, позволяющий выпускать разнообразные спецификации, ведомости и прочие табличные документы.

Документ-спецификация может быть ассоциативно связан со сборочным чертежом (одним или несколькими его листами) и трехмерной моделью сборки. При разработке функций и интерфейса КОМПАС учитывались приемы работы, присущие машиностроительному проектированию.

4.2 Типы документов системы КОМПАС

Тип документа, создаваемого в системе КОМПАС-3D, зависит от рода информации, хранящейся в этом документе. Каждому типу документа соответствует расширение имени файла и собственная пиктограмма.

Деталь - модель изделия, изготавливаемого из однородного материала, без применения сборочных операций. Файл детали имеет расширение m3d.

Сборка - модель изделия, состоящего из нескольких деталей с заданным взаимным положением. В состав сборки могут также входить другие сборки (подсборки) и стандартные изделия. Файл сборки имеет расширение a3d.

Чертеж - основной тип графического документа в КОМПАС-3D. Чертеж содержит графическое изображение изделия, основную надпись, рамку, иногда - дополнительные объекты оформления (знак неуказанной шероховатости, технические требования и т.д.). Чертеж КОМПАС-3D всегда содержит один лист заданного пользователем формата. В файле чертежа КОМПАС-3D могут содержаться не только чертежи (в понимании ЕСКД), но и схемы, плакаты и прочие графические документы. Файл чертежа имеет расширение cdw.

Фрагмент - вспомогательный тип графического документа в КОМПАС-3D. Фрагмент отличается от чертежа отсутствием рамки, основной надписи и других объектов оформления конструкторского документа. Он используется для хранения изображений, которые не нужно оформлять как отдельный лист (эскизные прорисовки, разработки и т.д.). Кроме того, во фрагментах также хранятся созданные типовые решения для последующего использования в других документах. Файл фрагмента имеет расширение ftw.

Спецификация - документ, содержащий информацию о составе сборки, представленную в виде таблицы. Спецификация оформляется рамкой и основной надписью. Она часто бывает многостраничной. Файл спецификации имеет расширение sprw.

Текстовый документ - документ, содержащий преимущественно текстовую информацию - текстовый документ. Текстовый документ оформляется рамкой и основной надписью. Он часто бывает многостраничным. В текстовом документе могут быть созданы пояснительные записки, извещения, технические условия и т.п. Файл текстового документа имеет расширение kdw.

4.3 Интерфейс системы

КОМПАС — это стандартное приложение Windows. Поэтому рабочий экран, который вы видите после запуска системы и загрузки документа, практически не отличается по своему внешнему виду от окон других приложений.

Окно системы состоит из следующих элементов

- **Область окон документов**

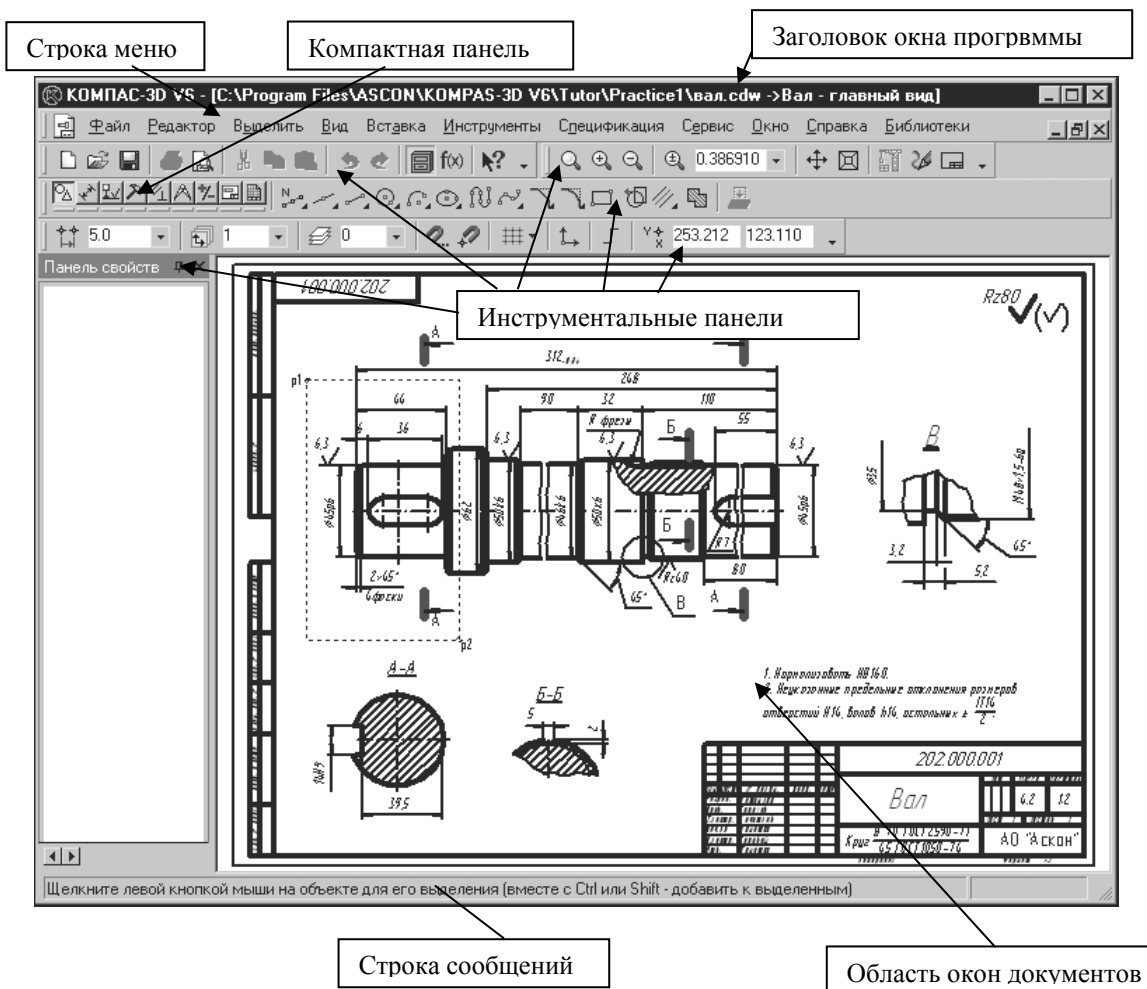
Эта область занимает основную часть окна программы КОМПАС!3D V7. Здесь размещаются окна открытых документов. В них выполняются все операции создания и редактирования документов системы.

- **Заголовок окна программы**

В нем отображается следующая информация:

- а) название и номер версии программы,
- б) тип открытого документа (чертеж, фрагмент и т.д.),
- в) полный путь к файлу (последовательность папок, определяющих положение файла на жестком диске компьютера),
- г) имя файла.

Если открытый документ является листом чертежа, то дополнительно отображается информация об имени текущего вида.



- **Строка меню**

Строка меню расположена в верхней части окна программы, ниже заголовка. В ней расположены страницы главного меню системы. Они содержат команды.

- **Инструментальные панели**

Это панели с кнопками, с помощью которых могут быть вызваны различные команды.

Среди них особым назначением обладают

- **Компактная панель**

Служит для быстрого переключения между присоединенными к ней инструментальными панелями

- **Панель свойств**

Служит для задания и редактирования параметров выбранного графического объекта системы

- **Строка сообщений**

Строка сообщений располагается в нижней части программного окна КОМПАС.

В ней отображаются различные сообщения и запросы системы. Это могут быть:

- а) краткая информация о том элементе экрана, к которому подведен курсор,
- б) сообщения о том, ввода каких данных ожидает система в данный момент,
- в) краткая информация о текущем действии, выполняемом системой.

4.4 Базовые приемы работы

Многие приемы работы с мышью и клавиатурой, которые используются практически во всех приложениях Windows, могут применяться и в КОМПАС.

Кроме того, система предоставляет пользователю ряд специальных приемов работы.

Курсор и управление им

Курсор — это главный инструмент при работе с КОМПАС. С помощью курсора осуществляется вызов команд из меню или с помощью кнопок, создание и редактирование объектов, выполняется множество других действий.

Внешний вид курсора зависит от типа активного документа и выполняемой операции. Стандартный вид курсора при нахождении в поле графического документа или документа-модели — это квадратная «ловушка». Для графических документов параметры курсора (размер, цвет и др.) могут настраиваться пользователем.

Основной способ управления курсором, доступный в документах всех типов — это его перемещение мышью. Можно также передвигать курсор, используя клавиши со стрелками на основной или расширенной клавиатуре. В этом случае перемещение будет не произвольным, как в случае использования мыши, а дискретным.

В текстовых документах и спецификациях (во время ввода текстовой части объекта) при нажатии на кнопку со стрелкой курсор перемещается на один символ или на одну строку.

В графических документах минимальное перемещение курсора при нажатии на кнопку со стрелкой зависит от установленного шага курсора. Для задания величины шага используется поле *Текущий шаг курсора* на панели *Текущее состояние*. Значения и количество шагов можно также установить в диалоге настройки курсора.

При работе с графическими документами можно также ввести координаты точки, в которую требуется поместить курсор, в поля Координаты курсора на панели *Текущее состояние*.

В графических документах и документах моделях после установки курсора в нужную точку его требуется зафиксировать — подтвердить, что для создания объекта должна использоваться именно эта точка. Фиксация производится щелчком левой кнопки мыши или нажатием клавиши **<Enter>**. Так, например, для построения отрезка необходимо указать и зафиксировать его начальную и конечную точки, при построении отверстия в модели можно изменить его расположение, предлагаемое по умолчанию указав и зафиксировав новое положение точки привязки.

Использование контекстных меню

Контекстное меню появляется на экран при нажатии правой кнопки мыши. Состав меню зависит от объекта, на который указывал курсор во время нажатия кнопки мыши, и от выполняемого действия. При этом в меню собраны команды, наиболее типичные для данного момента работы

Удобство работы с контекстными меню обеспечивается тем, что в нем сгруппированы команды, находящиеся в разных разделах Главного меню, но часто используемые при работе с данным объектом. Кроме того, контекстное меню всегда появляется на экране в том месте, где его вызвали щелчком мыши, что ускоряет выбор конкретной команды.

4.5 Приемы создания объектов

Основная задача, решаемая при помощи любой САПР — создание и выпуск различной документации. Скорость решения этой задачи, а значит, и эффективность работы с системой, в основном определяется тем, насколько удобные средства ввода и редактирования объектов она предоставляет пользователю.

Разрабатывая документы и модели с помощью КОМПАС, можно применять различные приемы создания и изменения объектов.

Параметры объектов

После вызова большинства команд создания объектов необходимо задать различные параметры этих объектов. Например, после вызова команды построения окружности требуется задание положения ее центра и радиуса, а после вызова команды построения тела выдавливания — направление, глубину выдавливания и величину уклона.

Каждому параметру объекта соответствует один элемент *Панели свойств*. В зависимости от типа параметра элемент может иметь вид поля ввода, опции, списка и др.

Существует два способа задания параметров объектов:

а) указание точек в рабочем поле мышью (так можно задать положение начальной и конечной точек отрезка, положение точки привязки отверстия в детали и т.п.),

б) ввод или выбор значений параметров на Панели свойств (так можно задать точное значение радиуса окружности, выбрать направление уклона граней тела выдавливания и т.п.).

Указание точек в рабочем поле — наиболее доступный и явный способ задания параметров. Его преимуществом является возможность использования привязок. Разновидностью этого способа в графических документах является ввод в поля панели *Текущее состояние* координат курсора и последующая его фиксация.

Задание параметров на *Панели свойств* – возможно, менее наглядный, но более универсальный способ. Его преимуществом при работе с графическими объектами является возможность использования геометрического калькулятора. Геометрический калькулятор вызывается щелчком правой кнопки мыши в поле какого-либо параметра объекта. Он выполнен в виде контекстное меню, содержимое которого зависит от типа параметра. С помощью команд меню можно выполнить снятие значений геометрических величин (координат точек, размеров, углов и т.п.) с объектов, при этом эти величины автоматически заносятся в поля параметров. Кроме того, в поле параметра можно ввести не только конкретные числовые значений, но и целые арифметические выражения, которые тут же вычисляются системой.

Практически любой графический объект можно построить, указывая его характерные точки. Большинство параметров трехмерных объектов, наоборот, можно задать только с помощью элементов управления *Панели свойств*.

Рассмотрим работу с числовыми параметрами графических объектов.

Рядом с названием параметра на *Панели свойств* находится переключатель, на котором отображается значок, соответствующий состоянию параметра.

<i>Значок</i>	<i>Состояние параметра</i>
Галочка	Параметр активен, т.е. система ожидает ввода именно его.
Перекрестие	Параметр зафиксирован, т.е. его значение остается постоянным при изменении остальных параметров.
Отсутствие значка	Параметр вспомогательный, т.е. его значение зависит от значений других параметров. Вспомогательный параметр в любой момент можно задать и зафиксировать.

Например, после вызова команды построения отрезка на *Панели свойств* появятся следующие числовые параметры: **t1** (координаты начальной точки), **t2** (координаты конечной точки), **Длина**, **Угол** (наклона к оси X текущей системы координат).

Таким образом, вы видите сразу все характеристики объекта и можете изменять любую из них непосредственно в процессе построения. При задании или изменении какого-либо параметра фантом объекта на экране соответствующим образом перестраивается.

Вы можете изменить очередность ввода параметров, т.е. вводить параметры в последовательности, отличной от предлагаемой системой.

По умолчанию после фиксации минимально необходимого количества параметров происходит фиксация самого объекта в документе. При необходимости вы можете отключить автоматическую фиксацию изображения.

При вводе выражений можно применять операции и функции, перечисленные в таблице.

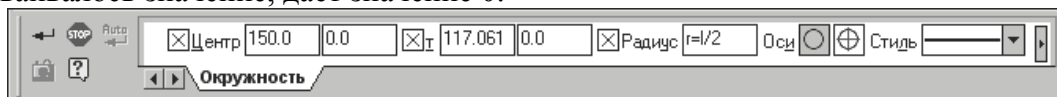
<i>Операция/Функция</i>	<i>Описание</i>
Арифметические операции:	
()	скобки операторные

*	умножить
/	разделить
%	разделить целочисленно
+	сложить, или унарный плюс
-	вычесть, или унарный минус
=	равно
Логические операции:	
==	тождественно
!=	не тождественно
>	больше
<	меньше
>=	больше или равно
<=	меньше или равно
!	логическое отрицание
&&	логическое И
	логическое ИЛИ
?:	логическое выражение вида $a ? b : c$ (если a — истина (не равно 0), то b , иначе c , где a , b и c могут быть выражениями)
Функции:	
sin	синус с аргументом в радианах
cos	косинус с аргументом в радианах
tan	тангенс с аргументом в радианах
atan	арктангенс с результатом в радианах
acos	арккосинус с результатом в радианах
asin	арксинус с результатом в радианах
sind	синус с аргументом в градусах
cosd	косинус с аргументом в градусах
tand	тангенс с аргументом в градусах
atand	арктангенс с результатом в градусах
acosd	арккосинус с результатом в градусах
asind	арксинус с результатом в градусах
deg	перевод из радиан в градусы
rad	перевод из градусов в радианы
sqrt	корень квадратный
exp	Экспонента
ln	натуральный логарифм
abs	абсолютное значение
ceil	округление до большего целого числа
floor	округление до меньшего целого числа
round	округление до ближайшего целого числа

После ввода выражения нужно дважды нажать клавишу **<Enter>**. После первого нажатия выражение будет зафиксировано, а после второго – вычислено.

Можете создать неограниченное количество переменных с буквенно-цифровыми именами и оперировать ими в выражениях. Имя переменной может состоять из букв латинского алфавита (заглавные и прописные буквы различаются), цифр и символа подчеркивания; оно должно начинаться с буквы или символа подчеркивания. Чтобы создать переменную и присвоить ей значение, в поле ввода любого параметра наберите выражение вида имя переменной = значение (например, « $a=25$ ») или имя переменной = выражение (например, « $b=a-8$ ») и нажмите клавишу **<Enter>**.

Выражение с использованием операций и функций, а также с вычисленными ранее переменными, можно набирать при любом запросе числа. Переменная, которой ранее не присваивалось значение, дает значение 0.



Любой из параметров создаваемого объекта можно зафиксировать. При этом значение данного параметра будет постоянным, а другие параметры останутся доступны для изменения. Признаком того, что параметр зафиксирован, является отображение перекрестия на переключателе рядом с полем параметра.

Фиксацию параметра можно рассматривать как ограничение возможных конфигураций создаваемого объекта.

Пусть нужно вычертить отрезок, длина которого должна точно равняться 60 миллиметрам. Задайте начальную точку отрезка, затем введите значение 60 в поле длины отрезка и нажмите клавишу **<Enter>**. Теперь при всех перемещениях курсора будет изменяться только угол наклона отрезка, а длина останется равной 60.

При указании точки в рабочем поле, а также при снятии параметра существующего объекта с помощью геометрического калькулятора фиксация этого параметра выполняется автоматически.

Если, перемещая курсор, вы достигли нужного значения какого-либо параметра и хотите его зафиксировать, то не сдвигая мышью, активизируйте поле параметра с помощью соответствующей комбинации клавиш (например, **<Alt>+<Y>** для угла наклона отрезка) и затем нажмите клавишу **<Enter>**.

Чтобы отменить фиксацию значения параметра, щелкните левой кнопкой мыши на переключателе рядом с названием этого параметра. Признаком того, что фиксация снята, является отсутствие перекрестия на соответствующем переключателе.

Пусть при вычерчивании отрезка была задана первая точка, а затем введено и зафиксировано значение длины, равное 60 миллиметрам. В дальнейшем при всех перемещениях курсора изменялся только угол наклона отрезка. Если теперь освободить длину отрезка, то можно будет вновь изменять как угол наклона, так и длину.

В процессе построения объекта можно многократно фиксировать и освобождать его параметры.

Значения некоторых параметров могут быть введены только с клавиатуры (то есть заданы явно) или сняты с уже существующих объектов с помощью геометрического калькулятора (к таким параметрам относятся, например, длина фаски, угол штриховки).

Значения других параметров могут задаваться также путем фиксации курсора в определенной точке поля документа (например, координаты точки, начальный и конечный углы дуги).

Во время задания параметра, относящегося к последней группе, на переключателе, соответствующем параметру, отображается «галочка», означающая, что параметр активный, т.е. система ожидает задания именно его значения. После задания значения параметр фиксируется и активным становится следующий параметр.

При создании объектов их параметры активизируются в порядке, установленном в системе по умолчанию. Однако при необходимости пользователь может изменить его.

Чтобы активизировать параметр объекта, щелкните левой кнопкой мыши на его названии (не на переключателе, соответствующем параметру, и не в поле ввода!). Параметр станет активным, и вы сможете задать его значение.

Пусть необходимо построить отрезок заданной длины, параллельный данному и оканчивающийся в заданной точке. Вызовите команду построения параллельного отрезка, укажите данный отрезок, введите значение длины отрезка и зафиксируйте его. По умолчанию точка, которую вы укажете в поле документа, будет восприниматься как начальная точка отрезка, поэтому сейчас «галочкой» отмечен параметр **t1**. Чтобы можно было указать эту же точку в качестве конечной точки отрезка, активизируйте параметр **t2**, щелкнув по его названию мышью. «Галочка» на переключателе параметра **t1** исчезнет, а на переключателе параметра **t2** появится; фантом

вводимого отрезка «повернется» на 180°. Теперь укажите положение конца отрезка— например, выполнив привязку к нужной точке.


Вы можете активизировать как вспомогательные, так и уже зафиксированные параметры (например, при редактировании объектов). В этом случае активизация параметра равносильна расфиксации его значения.

Часто требуется начертить несколько объектов, имеющих ряд одинаковых параметров.

Типичный пример— концентрические окружности (их совпадающими параметрами являются координаты точки центра). КОМПАС предоставляет возможность сохранить значение параметров и использовать их до завершения текущей команды при построении следующих объектов.

Для использования одинаковых параметров при создании объектов выполните следующие действия.

1. Задайте параметры, которые должны быть запомнены.

2. Нажмите кнопку *Запомнить состояние*.  Кнопка останется в нажатом состоянии, что свидетельствует о запоминании параметров.

3. Выполняйте построения до тех пор, пока нужны запомненные параметры.

4. Отожмите кнопку *Запомнить состояние*.



Если введенные параметры однозначно определяют объект (например, уже зафиксированы точка центра и радиус окружности), кнопка *Запомнить состояние* будет недоступна.


Запоминать параметры можно при построении различных объектов. Например:


- концентрические дуги с одинаковым углом раствора,
- отрезки одинаковой длины с одной и той же начальной точкой,
- отрезки одинаковой длины, параллельные одному и тому же объекту,
- окружности и дуги одинакового радиуса,

Автоматическое и ручное создание объектов

Когда изменяют параметры объекта при его построении, зачастую бывает не нужно создавать объект сразу после задания всех определяющих его параметров. Удобнее сначала оценить, правильно ли заданы значения параметров, а уже затем подтвердить создание объекта.

После вызова большинства команд ввода объектов на *Панели свойств* отображаются кнопки  («Автоматическое создание объекта») и  («Создать объект»)

Одна из них,  Автоматическое создание объекта, по умолчанию нажата. Пока она находится в этом состоянии, все объекты создаются (фиксируются) немедленно после ввода параметров, достаточных для построения. Если же не требуется, чтобы объекты создавались автоматически, нужно отожать эту кнопку.

Теперь, чтобы подтвердить создание каждого очередного объекта, нужно будет дополнительно нажать кнопку  Создать объект. До тех пор, пока эта кнопка не нажата, объект не считается зафиксированным, поэтому вы можете изменить любой его параметр любое количество раз. Каждое изменение будет немедленно отражаться на фантоме объекта в окне документа, что позволит контролировать правильность ввода значений.

Автосоздание невозможно при построении некоторых графических объектов (например, эквидистанты, штриховки) и практически всех трехмерных объектов. Их необходимо фиксировать вручную.

Прерывание команды

Чтобы завершить текущую команду, нужно выполнить одно из следующих действий:

Нажать клавишу <Esc>.

Отжать кнопку команды.

Вызвать из контекстного меню команду Отказ от команды.

Нажать среднюю кнопку трехкнопочной мыши (этот способ прерывания команды доступен, если на компьютере установлен драйвер трехкнопочной мыши.).

Вызвать любую другую команду.

Нажать кнопку «Прервать команду»  на Панели свойств.

Привязка

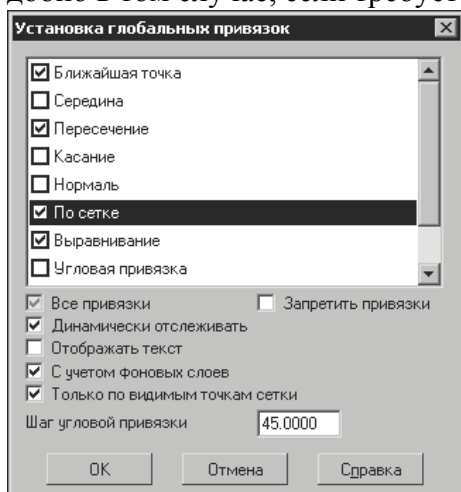
В процессе работы с графическим документом постоянно возникает необходимость точно установить курсор в некоторую точку (начало координат, центр окружности, конец отрезка и т.п.), иными словами, выполнить привязку к уже существующим точкам или объектам. Без такой привязки невозможно создать точный чертеж.

КОМПАС предоставляет возможности привязок к характерным точкам (пересечение, граничные точки, центр и т.д.) и объектам (по нормали, по направлениям осей координат). Все варианты привязок объединены в меню, которое можно вызвать при создании, редактировании или выделении объектов нажатием правой кнопки мыши.


Предусмотрены две разновидности привязки — глобальная (действующая по умолчанию) и локальная (однократная).

Глобальная привязка (если она установлена) постоянно действует при вводе и редактировании объектов. Например, если включена глобальная привязка к пересечениям, то при вводе каждой точки система автоматически будет выполнять поиск ближайшего пересечения в пределах ловушки курсора.

Локальную привязку требуется всякий раз вызывать заново. После того, как был использован один из вариантов привязки, система не «запоминает», какой именно это был вариант. Поэтому, когда потребуются выполнить к другой точке такую же привязку, ее придется вызвать снова. Это неудобно в том случае, если требуется выполнить несколько однотипных привязок подряд.



Локальная привязка является более приоритетной, чем глобальная, то есть при вызове какой-либо команды локальной привязки она подавляет установленные глобальные на время своего действия (до ввода точки или отказа).

Чтобы включить глобальные привязки в текущем окне, нажмите кнопку Установка глобальных привязок . На экране появится диалог установки глобальных привязок.

Виды привязок представлены в таблице.

<i>Пртвязка</i>	<i>Описание</i>
Ближайшая точка	Указание любой характерной точки существующего объекта (начальная точка, центр дуги/окружности и т.п.) или начало системы координат
Середина	Указание точки находящейся в середине линии (отрезка или дуги)
Пересечение	Указание точки пересечения двух линий
Касание	Созданная вновь линия будет касательна к выбранной линии (точка касания обычно задается второй или третьей при формировании линии)
Нормаль	Созданный вновь отрезок будет перпендикулярен выбранной линии (точка нормали задается второй при формировании отрезка)
По сетке	Привязывается к узлам мнимой сетки, размеры ячейки которой определяются настройками системы. Начало отсчета сетки совпадает с началом координат.
Выравнивание	Новая точка будет находится на одной вертикали или горизонтали и какой-либо характерной точкой

	существующего объекта.
Угловая привязка	При выборе данного способа привязки Новая точка будет находится относительно последней зафиксированной точки под углами, кратными указанному при настройке привязок значению. Например, при настройке привязок значение шага угловой привязки установлено равным 45°. Тогда в результате действия угловой привязки курсор будет перемещаться вдоль прямых, проходящих через последнюю зафиксированную точку, под углами 45°, 90°, 135° и т.д.
Центр	Указание точки центра окружности, дуги или эллипса. При указании точки необходимо наводить курсор на линию (дугу, окружность, эллипс), а не на ее центр.
Точка на кривой	Указание любой точки через которую проходит указанная линия.

Опция *Динамически отслеживать* управляет динамическим расчетом привязок. Если она включена, расчет выбранных привязок производится «на лету» – при подводе курсора к точке, к которой можно привязаться (она отмечается на экране фантомом в виде «крестика»). При выключенной опции расчет производится только после указания точки.

Опция *Отображать текст* управляет отображением названия сработавшей в данный момент привязки. Показ названия привязки возможен, если включено динамическое отслеживание.


Опция *С учетом фоновых слоев* управляет привязкой к объектам и точкам, лежащих в фоновых слоях документа.

Опция *Только по видимым точкам сетки* управляет привязкой *По сетке*. Включенная опция означает, что привязка по сетке будет возможна, только если сетка отображается на экране. При выключенной опции привязка по сетке возможна вне зависимости от ее присутствия на экране, а также к точкам сетки, ставшими невидимыми в результате разрежения.

Если включено несколько глобальных привязок и при текущем положении курсора возможно выполнение нескольких привязок, то срабатывает более приоритетная из них.

Список приоритетов совпадает с порядком перечисления привязок в диалоге их настройки. Например, если включены привязки *Ближайшая точка* и *Пересечение*, а «ловушка» при текущем положении курсора (например, при указании точки для выравнивания) захватывает характерную точку объекта и точку пересечения объектов, то сработает более приоритетная привязка *Ближайшая точка*.

Привязки Выравнивание и Точка на кривой (если они включены) могут срабатывать парно. Например, совместное использование привязок Выравнивание и Точка на кривой позволяет зафиксировать точку на кривой, имеющую ту же абсциссу или ординату, что и характерная точка какого-либо объекта.

Можно быстро отключить все глобальные привязки в активном окне («заморозить» привязки), а затем вновь включить их в прежнем составе, воспользовавшись кнопкой  (*Запретить/разрешить действие глобальных привязок*) на панели *Текущее состояние*.

Эта кнопка также служит индикатором действия глобальных привязок: нажатая кнопка означает, что глобальные привязки отключены, отжатая — включены.

Чтобы воспользоваться *локальной привязкой* при построении или редактировании графического объекта, вызовите из контекстного меню команду Привязка. Ее подменю – это перечень локальных привязок. Для включения привязки нужного типа вызовите соответствующую команду.

Курсор изменит свою форму, что свидетельствует о том, что привязка активна. Установите курсор так, чтобы его «ловушка» захватывала объект (или точку), к которому требуется привязаться. Например, если включена привязка Ближайшая точка, то требуется захватить характерную точку

какого-либо объекта; если включена привязка Центр, то — дугу, многоугольник или другой объект, имеющий центральную точку. В точке, соответствующей выбранной привязке, появится «крестик», свидетельствующий о срабатывании привязки

Клавиатурная привязка


Некоторые варианты привязки можно выполнять с помощью клавиатуры, нажимая для этого соответствующие комбинации клавиш. Эти комбинации представлены в таблице.

Комбинация	Описание
<Ctrl>+<0>	Установка курсора в точку (0,0) текущей системы координат.
<.> (точка)	Установка курсора по нормали в ближайшую точку ближайшего объекта.
<5>	Установка курсора в ближайшую к нему характерную точку объекта.
<Shift>+<5>	Установка курсора в ближайшую к нему середину объекта.
<Alt>+<5>	Установка курсора в ближайшую к нему точку пересечения объектов.
<Ctrl> +<←>, <Ctrl> +<→>, <Ctrl> +<↓>, <Ctrl> +<↑>	Установка курсора в ближайшую точку ближайшего объекта по направлению осей текущей системы координат.

Примечание Клавиши <0>, <.> и <5> следует нажимать на дополнительной клавиатуре. При этом должен быть включен цифровой режим ее работы (должен гореть индикатор NumLock).

Управление курсором с помощью клавиатурной привязки, в отличие от локальной, возможно в любое время, а не только при выполнении какой-либо команды.

Ортогональное черчение

Режим ортогонального черчения служит для быстрого создания объектов или их частей, ортогональных осям текущей системе координат. Включение и отключение этого режима производится кнопкой  **Ортогональное черчение**, расположенной на панели Текущее состояние. Другой способ управления ортогональным режимом — нажатие клавиши <F8>. Указанная кнопка служит также индикатором режима ортогонального черчения: нажатая кнопка означает, что ортогональное черчение включено, отжатая — выключено. Если в процессе построения объекта требуется временно отключить (или включить) этот режим, нажмите и удерживайте клавишу <Shift>.

Ортогональный режим используется при вычерчивании горизонтальных и вертикальных отрезков, обозначений ступенчатых разрезов, перпендикулярных друг другу участков ответвлений допуска формы и в других случаях.

4.6 Трехмерное моделирование в КОМПАС

Несмотря на то, что 2D-системы позволяют вполне успешно решать стоящие перед большинством пользователей задачи, по мере развития новых технологий все отчетливее проявляются серьезные ограничения, присущие плоскому проектированию.

Ограничения двумерного проектирования

Основной недостаток 2D-систем заключается в том, что при создании плоского чертежа конструктору приходится мыслить не в терминах проектируемой детали – *основание, отверстие,*

ребро жесткости, а в терминах традиционного набора геометрических примитивов – *отрезок, дуга, окружность* и т.д. Если возникает необходимость внести в деталь какие-либо изменения, то их необходимо заново отобразить на всех видах детали, что опять связано с большими затратами времени.

Ограничения 2D-систем особенно наглядно проявляются, когда поверхность детали имеет сложную форму или когда необходимо построить аксонометрическую проекцию.

Большая трудоемкость построения сложных поверхностей и аксонометрических проекций может заставить конструктора отказаться от их изображения или упростить форму детали. В первом случае это затрудняет понимание проекта, во втором – снижает привлекательность изделия с точки зрения потребителя.

Список ограничений двухмерного проектирования можно продолжить и дальше. В него можно включить сложность понимания взаимного положения и взаимодействия деталей в сборочных единицах, сложность или невозможность передачи данных в системы инженерного анализа и подготовки управляющих программ для станков с числовым программным управлением (ЧПУ) и так далее.

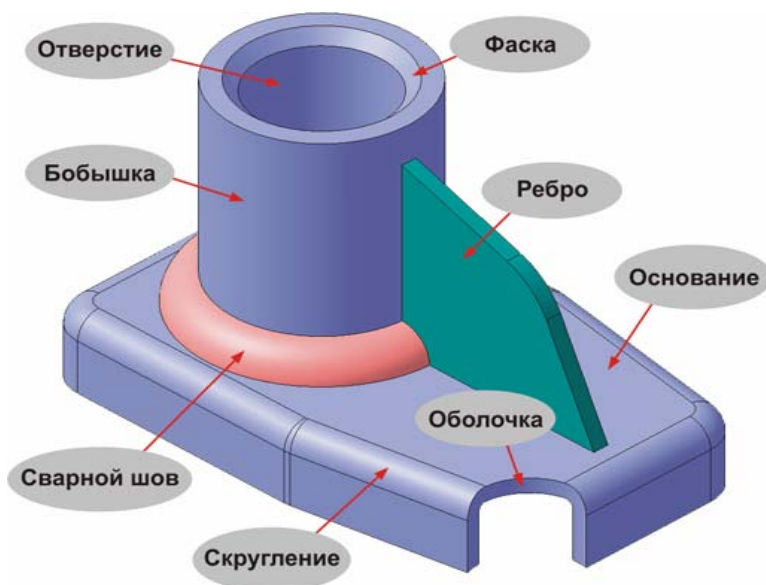
Приведенные примеры позволяют сделать вывод, что использование только лишь двухмерных систем как средства проектирования и подготовки чертежей порождает серьезные проблемы и замедляет выпуск новых изделий.

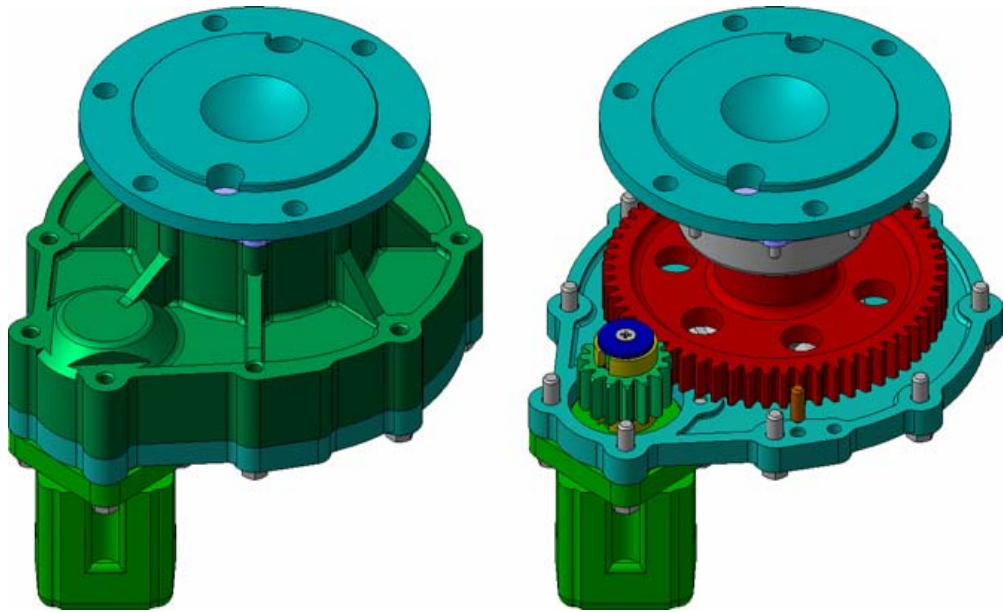
Переход на трехмерное моделирование

КОМПАС-3D располагает эффективными средствами моделирования, которые позволяют создавать трехмерные модели самых сложных деталей и сборок. При проектировании деталей, используя наглядные методы создания объемных элементов, конструктор оперирует простыми и естественными понятиями: *основание, бобышка, ребро жесткости, отверстие, фаска, оболочка*. При этом процесс проектирования часто воспроизводит технологический процесс изготовления детали.

В любой момент, непосредственно на экране монитора, конструктор может выполнить разрез модели стандартными или дополнительными плоскостями проекций, или построить свой, самый невероятный разрез.

При проектировании сборочных единиц конструктор работает с деталями, сборками, подсборками и стандартными изделиями. В процессе построения трехмерных моделей сборочных единиц конструктор имеет возможность временно отключать отображение любых элементов. Это особенно удобно, если модель включает в себя корпусные детали, в которых размещены остальные компоненты изделия.





После построения 3D-модели детали или сборки, либо непосредственно в ходе построения, конструктор может получить ее чертеж, избежав таким образом рутинного создания видов средствами плоского черчения. Для этого нужно лишь указать необходимые виды, провести линии разрезов или сечений. Плоский чертеж будет создан автоматически и с абсолютной точностью, не зависимо от сложности модели. Полученный таким образом документ можно дорабатывать встроенными в систему средствами 2D-черчения: проставить размеры, обозначения позиций, заполнить основную надпись или подготовить спецификацию.

В КОМПАС-3D объемные модели и плоские чертежи ассоциированы между собой. Это означает, что любое изменение, внесенное в модель, будет немедленно и точно отражено на всех видах чертежа.

КОМПАС-3D располагает мощными средствами редактирования модели, которые позволяют задавать параметрические связи и ассоциации как между отдельными элементами деталей, так и между деталями в сборочных единицах. Это дает возможность быстро вносить изменения в проект и создавать различные варианты как отдельных деталей, так и всего изделия в целом.

По трехмерной модели система легко определяет ее физические характеристики: площадь поверхности, объем, координаты центра масс и т.д. Если пользователь определил свойства материала, то автоматически вычисляется масса. Это касается как деталей, так и сборок любой сложности.