

Федеральное агентство по образованию  
Псковский государственный политехнический институт

*Е.И. Самаркина*

# **Автоматизация конструкторско- графических работ в машиностроении**

*Конспект лекций  
для студентов машиностроительных специальностей*

рекомендовано к изданию Научно-методическим советом Псковского  
государственного политехнического института

*Рекомендовано к изданию Научно-методическим советом  
Псковского государственного политехнического института*

Псков  
Издательство ППИ  
2008

УДК 681.5  
С17

Рекомендовано к изданию Научно-методическим советом  
Псковского государственного политехнического института

Рецензенты:

- Чижевский А.Б. – к.т.н. зам. генерального директора ООО «Уклад»;
- Ветлицын А.М. – к.т.н., доцент кафедры ТМ Псковского государственного политехнического института.

**Самаркина Е.И. Автоматизация конструкторско-графических работ.**

Конспект лекций. – Псковский государственный политехнический институт. – Псков: Издательство ППИ 2008 – 66 с.

В конспекте лекций рассматриваются теоретические основы компьютерной графики, представлены не только ставшие уже традиционными основы двумерного моделирования, но и принципы создания трехмерных объектов .

Преимуществом работы является объективность изложенного материала, непривязанного к реализации соответствующих подходов в одной единственной САПР, а также наглядная демонстрация принципов проектирования на многочисленных примерах.

Сведения, приведенные в конспекте могут быть полезными и более широкому кругу специалистов, использующих в своей работе системы автоматизированного проектирования для лучшего понимания основ их работ.

Конспект лекций предназначен для студентов машиностроительных специальностей всех форм обучения при изучении курсов «Компьютерная подготовка конструкторской документации», «Дизайн и конструирование изделий с применением САД/САЕ систем» и «Основы компьютерной визуализации».

УДК 681.5  
С17

© Псковский государственный  
политехнический институт, 2008  
© Самаркина Е.И., 2008

## Содержание

Лекция 1 .....	4
Введение.....	4
Понятия автоматизированного проектирования и САПР .....	4
САД. Область применения и классификация.....	5
Понятие конструкторской модели.....	7
Эволюция разработок в области САД.....	9
Лекция 2 .....	11
Введение.....	11
Настройка параметров чертежа .....	11
Базовые инструменты черчения.....	16
Функции аннотирования.....	19
Вспомогательные функции .....	20
Лекция 3 .....	23
Введение.....	23
Графические библиотеки.....	24
Изображение объекта на экране графического устройства.....	26
Окно и видовой экран .....	31
Дисплейный файл.....	33
Лекция 4 .....	35
Преобразования геометрии .....	35
Трансляция.....	36
Вращение.....	37
Отображение .....	39
Масштабирование .....	41
Зеркальное отображение.....	42
Визуализация объемных моделей.....	42
Лекция 5 .....	44
Введение.....	44
Классификация геометрических моделей.....	45
Область применения твердотельных геометрических моделей.....	45
Основные понятия и принцип твердотельного моделирования.....	47
Операции твердотельного моделирования .....	48
Лекция 6 .....	58
Структура данных для хранения описаний объемных тел .....	58
Структура данных В-Рер.....	58
Основные понятия параметрическое моделирование. ....	60
Область применения параметрических свойств модели.....	63
Философия моделирования детали (в среде Pro/ENGINEER).....	64
Литература .....	67

# Лекция 1

**Тема:** САПР и автоматизация конструирования.

**Цель:** Исследование области применения САПР при конструировании.

## **План:**

Введение

1. Понятия автоматизированного проектирования и САПР.
2. CAD. Область применения и классификация
3. Понятие конструкторской модели. Классификация конструкторских моделей
4. Эволюция разработок в области CAD

## **Введение**

Основная цель дисциплины: Получить представление об основных принципах и теоретических основах работы конструкторских систем автоматизированного проектирования (САПР).

Студент инженерной специальности со временем станет пользователем одной из систем автоматизации конструирования. В руководствах пользователя системы автоматизированного проектирования обычно уделяется основное внимание интерфейсу и предполагается, что пользователь имеет определенную теоретическую подготовку в области автоматизации проектирования. Без такой подготовки студент встретит серьезные затруднения с терминологией системной документации, возникнут сложности с анализом сообщений об ошибках. Таким образом, для эффективной работы с имеющимся в компьютерном классе программным обеспечением и для самостоятельного освоения новых САПР с конкретной средой проектирования в домашних условиях нужны фундаментальные знания.

## **Понятия автоматизированного проектирования и САПР**

Понятие САПР тесно связано с понятием автоматизированного проектирования.

*Автоматизированным проектированием* называется проектирование, которое выполняется в процессе взаимодействия (обычно в режиме диалога) пользователя/ей и ЭВМ.

В свою очередь автоматизация проектирования реализуется с помощью систем автоматизированного проектирования (САПР). Таким образом, автоматизированное проектирование – технология, состоящая в использовании САПР.

*Система автоматизированного проектирования (САПР) - комплекс средств автоматизации проектирования, взаимосвязанных с подразделением специалистов (пользователей системы), выполняющих автоматизированное проектирование.*

САПР объединяет в едином комплексе технические средства, математическое, программное, методическое, информационное, организационное обеспечения, характеристики и параметры которых определяются задачами проектирования и конструирования.

На первый взгляд при функционировании САПР у пользователя создается впечатление, что решение конкретных задач проектирования выполняется только программным обеспечением (ПО), введенным в аппаратные средства ЭВМ. Однако созданное разработчиками САПР ПО - это всего лишь воплощенное на языке программирования математическое обеспечение: объектные структуры, алгоритмы вычислений, математические модели. К тому же ПО, как правило, в ходе решения задачи проектирования взаимодействует с информационным обеспечением предметной области проектирования (например, данными о типовых (стандартных) деталях, об имеющемся на предприятии инструменте, станочном парке). Объем информационного обеспечения обычно во много раз превышает размеры ПО. Пользователь со своей стороны, для полноценной работы в составе САПР должен освоить, по крайней мере, лингвистическое и методическое обеспечение. Эти знания приобретаются в результате практического использования САПР.

Таким образом, задача курса не только в теоретической проработке основ САПР, но и в изучении различных прикладных систем с целью приобретения реального опыта работы.

## **CAD. Область применения и классификация**

*CAD (Computer Aided Design) – САПР, с помощью которой выполняется объемное и плоское геометрическое моделирование деталей и сборок, выпуск конструкторской документации.*

Основная функция CAD – определение геометрии конструкции (детали, механизма, электрической схемы, архитектурного элемента, планировки цеха, есть и другие применения).

Для конструирования среди CAD по назначению выделяются две группы программного обеспечения(ПО):

- Системы разработки чертежей;

- Системы геометрического моделирования.

Системы разработки чертежей и геометрического моделирования являются наиболее важными компонентами комплексной автоматизации проектирования, которая охватывает решение всех проектных задач на протяжении всего времени разработки нового изделия.

### **Системы разработки чертежей**

Любая система конструкторского проектирования(CAD) включает в себя инструменты, заменяющие кульман - систему или подсистему(модуль) разработки чертежей.

Основная задача этого модуля: исполнение сборочных чертежей и чертежей деталей, схем по всем правилам, т.е. в соответствии с ЕСКД, как правило, для последующего получения твердых копий.

При обычном компьютерном черчении автоматизируются:

1. Геометрические построения (прямые, окружности, дуги и т.д.) с использованием различных привязок (к уже существующим элементам поверхности детали, сетке, произвольной системе координат и т.д.) различными способами (дуга - по двум точкам, по трем точкам) с вводом значений посредством мыши и с клавиатуры.
2. Рутинные и трудоемкие работы (изменение надписей, штриховок, простановка размеров, получение изображений часто повторяющихся элементов, обновление модели при редактировании).
3. Процесс компоновки листа для печати.

Преимущества систем разработки чертежей:

- Позволяют подготовить конструкторскую документацию на изделие и его элементы средней и низкой сложности, каковых большинство, с наименьшими затратами;
- Позволяют подготовить технологическую (технологические эскизы) документацию для универсального и другого не оснащенного ЧПУ оборудования. (на отечественном предприятии большая часть станочного оборудования не имеет числового управления).

Недостатки систем разработки чертежей:

- Двумерное представление деталей сложной формы и сборок, состоящих из большого количества элементов не наглядно, что создает сложности в принятии конструкторских решений;

- При изготовлении изделий и их элементов на автоматизированном оборудовании с числовым программным управлением (станки с ЧПУ, сборочные автоматы) вывод на печать двумерных чертежей становится не обязательным;
- Из-за усложнения изделий они не могут быть адекватно представлены с помощью чертежей.

### **Системы геометрического моделирования**

Большинство CAD-систем органически сочетает в единой среде возможности электронного кульмана и мастерской макетчика. То есть позволяют выполнять следующие действия:

- создавать объемные модели деталей,
- разрабатывать сборочные узлы из объемных моделей деталей.

Преимущества систем геометрического моделирования:

- Наглядность модели делает ее удобной при принятии конструкторских решений;
- Точность и качество получаемой модели - необходимые свойства геометрии передаваемой в расчетные пакеты CAE, САМ-пакеты для разработки УП для оборудования с ЧПУ.
- модели деталей содержат внутренний объем, ограниченный внешней поверхностью. Такое представление позволяет определять объем тела, его массу, моменты инерции, центр тяжести и т.п. (эта возможность особенно важна при анализе конструкции и при принятии конструкторских решений).
- точное представление внешнего вида изделий или помещений позволяет использовать модели в симуляции процессов манипулирования изделием в виртуальной реальности, например, во время обучения работы на нем (модель пульта управления самолетом, локомотивом).

### **Понятие конструкторской модели**

Как правило, при автоматизированном проектировании и анализе изделия или технологического процесса инженер работает не с реальным изделием и даже не с его макетом, реализованным в материале, а с виртуальной компьютерной моделью, которая обладает лишь частью свойств своего прототипа.

В инженерной практике *компьютерная математическая модель* - математическая модель технического устройства(процесса), реализованная с помощью универсальных или специализированных САПР или расчетных программ.

*Конструкторская модель изделия* - это математическая модель изделия, которая разрабатывается на этапе проектирования.

Конструкторская модель изделия, как правило, включает все модели деталей и сборок изделия и относящуюся к изделию документацию, разработанные с помощью САПР.

Все модели деталей и сборок изделия относятся к классу геометрических моделей.

Геометрические модели содержат информацию об элементном составе детали или узла, связях между элементами, о форме и размерах элементов составляющих модель, об их взаимном расположении.

Конструкторские модели деталей и сборок разрабатываются с помощью CAD средствами проволочного, твердотельного, поверхностного, моделирования. В зависимости от того, какая из концепций создания геометрических моделей использована, и какой математический аппарат реализован при построении моделей, конструкторскую геометрическую модель детали или сборки можно классифицировать как:

- каркасную
- твердотельную
- поверхностную
- гибридную модель.

Модели деталей и сборок изделия, разработанные с использованием системы (модуля) разработки чертежей - двумерные модели (2D), относящиеся к классу каркасных.

Модели деталей и сборок изделия, разработанные с использованием системы геометрического моделирования – объемные трехмерные модели (3D).

Если геометрическую 3D модель получили в системе твердотельного моделирования(Solid Works), то такая модель объемная твердотельная (определение «объемная» можно опустить).

Если геометрическую 3D модель получили в системе поверхностного моделирования, то такая модель поверхностная

Геометрическая 3D модель, включающая поверхности как элементы модели и твердотельные элементы - гибридная модель



Каркасная модель может быть получена как в системе твердотельного, так и в системе твердотельного моделирования – это способ представления твердотельной модели с помощью ребер и вершин.

## **Эволюция разработок в области САД**

Термин САПР (в английской нотации CAD) появился в конце 50-х годов. Профессор Росс дал название своему проекту, над которым он работал в Массачусетском Технологическом Институте.

Эволюцию практических разработок САПР с достаточной степенью условности рассмотрим, используя понятие поколение систем.

Эволюцию CAD свяжем с развитием возможностей геометрического моделирования.

### **Поколения САД**

- I. –е поколение. Системы 2D-моделирования. Первые коммерческие CAD-системы появились в 60-е годы. Применялись больше для черчения, чем для проектирования. Они позволяли строить плоские чертежи и работали на универсальных компьютерах с дорогими графическими мониторами. Программы были написаны на Фортране.
- II. –е поколение. Системы каркасного моделирования. В 70-е годы к функциям плоского черчения были добавлены возможности каркасного 3-D моделирования. В качестве аппаратных средств стали применяться компьютеры класса мини ЭВМ.
- III. –е поколение. Системы поверхностного, твердотельного моделирования. Быстрое развитие аппаратного и программного обеспечения в 80-е обусловило появление более совершенных технологий поверхностного и твердотельного моделирования. Программное обеспечение переводится на объектно-ориентированные языки программирования высокого уровня, переносится на автоматизированные рабочие места на базе графических станций Unix. Графическая станция Unix открыла дверь к многочисленным усовершенствованиям CAD-технологий.

Последнее десятилетие было отмечено активным развитием CAD-систем на основе твердотельного и поверхностного моделирования. Правда распространение систем сдерживала высокая стоимость автоматизированного рабочего места (АРМ), но в конце 90-х сначала CAD-системы 2-х мерного, а затем 3-х мерного

моделирования стали переводиться на более дешевые, нежели графические станции персональные компьютеры (ПК). В настоящее время появились предпосылки для появления следующего поколения САПР.

IV. –е поколение. Системы гибридного моделирования. К 2004 году появились системы, позволяющие создавать модель одновременно средствами поверхностного и твердотельного моделирования, причем получаемая модель полностью параметризована, а значит, может быть в любой момент отредактирована.

## Лекция 2

**Тема:** Основные понятия графического моделирования. Типичные функции плоского моделирования в CAD

**Цель:** Знакомство с основными операциями плоского моделирования.

### План:

Введение

1. Настройка параметров чертежа
2. Базовые инструменты черчения
3. Функции аннотирования
4. Вспомогательные функции

### Введение.

*Система автоматизированной разработки чертежей (computer-aided drafting system)* — это программный продукт, позволяющий разработчику в интерактивном режиме создавать и изменять машиностроительные, архитектурные, инженерные чертежи, электрические схемы и чертежи множества других разновидностей. Эта программа, кроме того, обновляет базу данных, сохраняя готовые чертежи и их изменения.

Работа с системой автоматизированной разработки чертежей аналогична использованию текстового процессора. Единственное отличие в том, что на выходе пользователь получает чертеж, а не текстовый документ. Как в текстовом процессоре можно очень быстро подготовить новый документ на базе существующего, так и в системе автоматизированной разработки чертежей можно получить новый чертеж, изменив имеющийся.

При изменении существующих чертежей преимущества системы автоматизированной разработки чертежей становятся очевидными и неоценимыми, благодаря сокращению трудоемкости подготовки конструкторской документации.

Далее будут кратко рассмотрены наиболее типичные функции, имеющиеся в большинстве систем автоматизированной разработки чертежей. Конкретные команды вызова функции в каждой системе могут быть свои, так что при необходимости следует обратиться к руководству пользователя соответствующей системы.

### Настройка параметров чертежа

Работу с системой автоматизированной разработки чертежей следует начинать с установки параметров. Основные параметры:

- стандарты оформления,
- единицы измерения,
- размеры чертежа,
- параметры сетки,
- параметры слоев.

Для быстрого и точного построения чертежей необходимо, чтобы все эти параметры имели правильные значения.

Чертеж можно построить без сетки и без слоев, но на это уйдет много времени, а изменить получившийся чертеж будет очень сложно.

Рассмотрим указанные параметры чертежа более подробно.

### **Стандарты оформления**

Использование существующих стандартов оформления и создание новых является прогрессивным направлением в автоматизированной разработке чертежей.

На сегодняшний день наиболее широко используемыми являются следующие стандарты:

- ISO - стандарт международного комитета по стандартизации,
- ANSI дюймовый, метрический - США,
- DIN- национальный немецкий,
- ГОСТ - национальный российский,
- HASCO- стандарт предприятия, фирмы по производству оснастки, и многие другие.

Выбор стандарта оформления повлечет появление особенностей в оформлении чертежей - единообразного оформления, особенностей в простановке размеров, подключение стандартного набора библиотек по-умолчанию и других.

В Компасе, например, можно включить режим оформления по ЕСКД или строительный стандарт. Можно внести изменения в существующие стандарты оформления, сохранить эти настройки и использовать как стандарты оформления своего предприятия.

### **Единицы измерения**

Пользователь должен выбрать формат и точность единиц измерения расстояний и углов.

Единицы измерения расстояний представлены:

- в научном,
- десятичном,

- дробном,
- инженерном,
- архитектурном форматах.

Единицы измерения углов:

- градусы,
- градусы/минуты/секунды,
- грады,
- радианы,
- геодезические единицы.

### **Размеры чертежа**

Рисую чертеж на бумаге, конструктор не выходит за границы листа. Точно так же и при работе с графическим устройством чертеж должен иметь определенные границы, поскольку этот чертеж когда-нибудь все равно будет напечатан на бумаге конечного размера. Таким образом, пользователь должен заранее установить размеры чертежа.

При выборе размеров чертежа обычно учитывают следующие факторы:

- фактический размер чертежа;
- пространство для нанесения размеров, примечаний, списков материалов и других необходимых данных;
- расстояние между разными видами (чертеж не должен выглядеть загроможденным);
- пространство для рамки и заголовка, если таковые предусмотрены.

Перед тем как задавать размеры чертежа, рекомендуется построить его эскиз, чтобы грубо оценить необходимое пространство.

Например, размеры вида спереди для какого-то объекта равны 65мм, размеры вида сбоку — 45мм и размеры вида сверху — 64мм. Предположим, вы хотите, чтобы расстояние между видами спереди и сбоку составляло 30мм, а расстояние между видами спереди и сверху — 20мм. Ограничения должны быть установлены таким образом, чтобы вместить весь чертеж и все относящиеся к нему данные. Подходящий формат можно подобрать по таблице указанной ниже. Если есть пожелания о конкретном размере чертежа, дополнительно следует применить масштабирование объекта.

**Таблица**

Размер бумаги	Размер листа	Границы чертежа (М1:1)	Границы чертежа (М1:5)	Границы чертежа (М5:1)
A4	210x297	210, 297	1050, 1485	42, 59,4
A3	297x420	297, 420	1485, 2100	59,4, 84
A2	420x594	420, 594	2100, 2970	84, 118,8
A1	594x841	594, 841	2970, 4205	118,8, 168,2
A0	841x1189	841, 1189	4205, 5945	168,2, 237,8

## **Слой**

В Компасе, например, каждому виду соответствует свой набор слоев или хотя бы один слой.

Под *ВИДОМ* здесь понимается прямоугольная область с собственной системой координат (произвольно расположенная в формате чертежа, которую можно переместить развернуть и т.д.) независимая от других видов. У вида также может быть собственный масштаб отображения.

*Слой* – группа объектов в пределах вида, которой принадлежат элементы чертежа. Слой аналогичен прозрачной кальке наложенной на ватман, все объекты на кальке воспринимаются как единая группа. Элементы чертежа, которые принадлежат слою, приобретают ряд свойств, характерных для слоя.

Система автоматизированной разработки чертежей предоставляет возможность переключения между режимами просмотра слоев.

Свойства слоя характеризуются состоянием и параметрами.

Состояния:

- видим, не видим,
- не активен, активен

При этом возможны сочетания состояний слоя. Слой может быть видим, но не активен, видим и активен, не видим и не активен.

К параметрам относятся

- имя слоя,
- цвет,
- тип линий вида,
- материал.

Можно создавать группы слоев – папки. Ряд программ использует слои для группировки объектов по смыслу (например, AutoCAD Mechanical)

*Слой, с которым вы работаете в данный момент, считается **активным**, тогда как все остальные слои считаются **неактивными**.*

Как и любой другой фон, графические элементы неактивных слоев нечувствительны к графическим операциям, таким как выбор или удаление. Поэтому сложность чертежа остается той же самой, как если бы вы работали с одним-единственным активным слоем.

Разбивать чертеж на множество слоев очень удобно, особенно если чертеж достаточно сложный.

Распределив детали отдельных узлов изделия по слоям, можно значительно упростить себе задачу создания сборочного чертежа.

Рекомендуется в сложных случаях разбивать содержимое и детализированного чертежа следующим образом:

- 1 слой вспомогательные построения,
- 2 слой основных линий контура детали (образующих чертеж изделия),
- 3 слой для осей, штриховок и т.п.,
- 4 слой размеров, особых требований,
- 5 слой комментариев, пояснений.

В случае необходимости слоев может быть сколько угодно, по желанию разработчика. Но зачастую количество слоев и их параметры регламентированы стандартом предприятия.

Преимущества послойного создания чертежа:

- Значительно проще выполнять все графические операции с отдельным слоем, чем с большим чертежом, содержащим все объекты,
- Накладывая слои друг на друга можно анализировать окружение проектируемой детали или узла, то есть можно получить представление об относительном расположении элементов из разных слоев,
- Переключаясь между режимами просмотра слоев можно легко выбирать отдельные фрагменты сложного чертежа для документирования (детализировки) и других преобразований.

### **Сетка и привязка**

В черчении на бумаге широко используются вспомогательные линии, которые строятся заранее при помощи рейсшины. Они облегчают построение линий чертежа и делают их более точными. Линии сетки в системах автоматизированной разработки чертежей имеют то же назначение, что и вспомогательные линии построения в черчении. Горизонтальные и вертикальные линии сетки автоматически отрисовываются на равных расстояниях друг от друга в соответствии с заданным разрешением, а линии чертежа строятся поверх них. В некоторых системах автоматизированной разработки чертежей строятся только точки на перекрестьях линий сетки.

Чтобы провести прямую линию поверх линии сетки, нужно задать положение двух ее концов. Их координаты можно ввести с клавиатуры или указать мышью, установив курсор в нужное положение и нажав кнопку.

Вторым способом курсор отслеживает движение мыши, когда она находится в режиме локатора. Положение точки может быть не совсем точным из-за дрожания человеческой руки или неточности механизма мыши. Чтобы справиться с этой проблемой, можно включить привязку курсора к ближайшему пересечению линий сетки. При нажатии кнопки мыши компьютер будет воспринимать точные координаты этого пересечения. Точность задания координат будет определяться разрешением сетки, которое пользователь может настраивать по своему желанию. Эта функция называется привязкой (snapping).

## **Базовые инструменты черчения**

К таким инструментам относятся:

- точка,
- вспомогательная прямая (бесконечная)
- отрезок прямой линии (полилиния),
- дуга, окружность,
- эллипс, парабола, гипербола – конические сечения,
- сплайн.

Как отдельный инструмент выносятся построение прямоугольника, скругления, но они являются сложной модификацией базовых функций

Рассмотрим некоторые базовые инструменты.



## **Прямая линия**

В системах автоматизированной разработки чертежей существует множество способов построения отрезков.

1. Наиболее популярным из них является построение по двум конечным точкам. Положение точек может быть задано различными способами. Ранее эти методы упоминались: ввод координат с клавиатуры и нажатие кнопки мыши в режиме локатора. Помимо этого вы можете указать конечную точку отрезка, выбрав одну из уже имеющихся на экране точек.
2. Отрезок можно построить и без явного указания обоих концов. Один из способов, - ввести ограничение провести касательную линию к имеющейся кривой на указанной точке. В этом случае явно указывается только одна точка, а вторую точку система определяет самостоятельно.

В качестве атрибутов линии могут быть указаны ее тип, цвет и толщина. Типы линий, поддерживаемые большинством систем автоматизированной разработки чертежей: утолщенная, тонкая, основная, осевая, прерывистая.

## **Окружность и дуга окружности**

Простейший метод задания окружности - указание ее центра и длины радиуса. Другой способ - задание трех точек на самой окружности. Большинство систем автоматизированной разработки чертежей позволяют создавать окружности и другими методами – с использованием геометрических взаимосвязей. Например, система может построить окружность, касательную к двум прямым или к другой окружности и прямой. В любом случае необходимо выбрать соответствующие объекты.

Дуга окружности - это частный случай окружности, она определяется заданием точек начала и конца (помимо параметров, задаваемых для обычной окружности).

## **Слайн**

Слайны используются для построения произвольных кривых подобно тому, как в черчении от руки это делается с помощью лекала. Пользователь указывает точки на кривой, а система строит интерполяционную кривую, проходящую через эти точки. Получившаяся кривая обычно представляется уравнением третьего порядка. Иногда кривые могут строиться по задающим точкам, которые определяют кривую, но не обязаны лежать на ней.

## Скругление и снятие фасок

*Скругление и закругление (filleting, rounding)* состоят в построении дуги окружности между двумя пересекающимися отрезками таким образом, что построенная дуга оказывается касательной к обоим отрезкам.

Скругление используется для вогнутых углов, а закругление — для выпуклых.

Снятие фасок (chamfering) — примерно то же, что и скругление, но вместо дуги строится отрезок прямой.

Скругление и снятие фасок осуществляются в следующем порядке.

1. Указывается радиус скругления или размер фаски.
2. Выбираются два пересекающихся отрезка. Скругление или фаска будут построены около точки пересечения.
3. Ненужные части исходных отрезков удаляются после построения скругления или фаски. В некоторых системах удаление производится автоматически, а в других это приходится делать вручную.

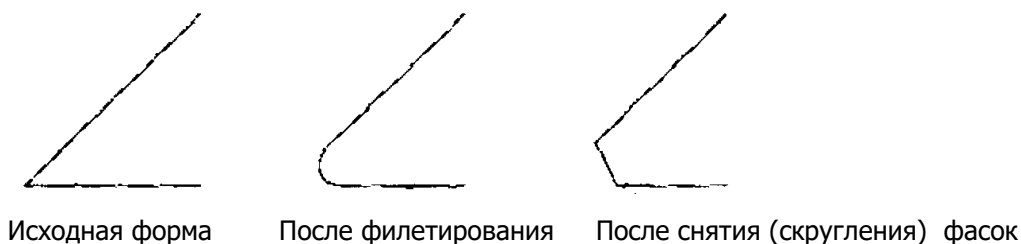


Рис. 1. Скругление и снятие фасок

## Штриховка

Штриховка — одна из функций систем автоматизированной разработки чертежей, повышающих производительность чертежника.

*Штриховкой* называется заполнение, как правило, замкнутого многоугольника каким-либо шаблоном.

Штриховка часто используется для обозначения сечений в машиностроительных чертежах и выделения разных материалов в архитектурных чертежах. Некоторые наиболее типичные шаблоны, предоставляемые большинством систем автоматизированной разработки чертежей, показаны на рис. 2.

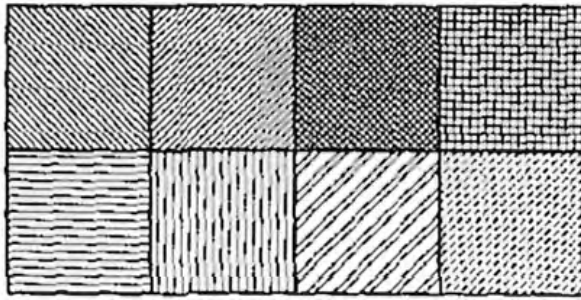


Рис. 2. Образцы штриховки

Последовательность действий при штриховке:

1. Задание образца штриховки,
2. Задание области штриховки.

Штриховка начинается с указания замкнутого многоугольника. Эта операция может осуществляться по-разному. В некоторых системах вам придется указать все отрезки, составляющие многоугольник. В других системах достаточно указать один из них, а все остальные система найдет автоматически.

Указание областей, которые не штрихуются. Если внутри многоугольника имеются участки, которые штриховать нежелательно, их границы также должны быть указаны.

## Функции аннотирования

### Простановка размеров

Возможность автоматизированной простановки размеров считается одной из наиболее привлекательных особенностей систем автоматизированной разработки чертежей. Вручную проставить размеры с той же быстротой просто невозможно.

В системах автоматизированной разработки чертежей простановка размеров осуществляется следующим образом:

1. Чтобы указать горизонтальный или вертикальный размер, достаточно всего лишь выбрать два графических элемента (обычно точки).
2. Указать желаемое положение размерной линии.

В этом случае расстояние между точками автоматически определяется по чертежу. Стрелки, размерные линии, выносные линии и значение размера наносятся системой самостоятельно. Система автоматически измеряет расстояние по вертикали, если графические элементы расположены на вертикальной линии, или расстояние по горизонтали, если они находятся на горизонтальной линии. Если

объекты расположены как-то иначе, система просит уточнить, какой именно размер вы хотите проставить: вертикальный, горизонтальный или реальный.

Размеры радиусов и диаметров проставляются путем выбора окружности или дуги и последующего указания положения размерной линии.

Угловые размеры проставляются аналогичным образом: нужно выбрать два отрезка и указать положение размерного текста. Какой именно угол будет измерен зависит от порядка выбора отрезков.

Может возникнуть вопрос: зачем проставлять размеры в интерактивном режиме, если чертеж уже содержит все сведения о размерах и положении объектов? Теоретически можно было бы полностью автоматизировать простановку размеров на чертежах, но на практике разработчики сталкиваются со следующими проблемами. Существует много способов проставить размеры в одном и том же чертеже. Вообще говоря, проектировщик должен учитывать методы производства, контроля и сборки детали, изображенной на чертеже, и на основании главным образом своего опыта выбирает оптимальную схему образмеривания. Воспроизвести опыт проектировщика на компьютере очень сложно, Более того, найти хотя бы одну полную и безызбыточную схему расстановки размеров не так-то просто. В настоящий момент ведутся исследования возможности решения этой задачи.

### **Примечания**

Чтобы добавить к чертежу примечание, то есть текстовую строку, нужно задать расположение и ориентацию этой строки, а также размер и шрифт символов. Последние три параметра обычно имеют некоторые значения по умолчанию, которые используются в том случае, если пользователь не указывает никаких конкретных значений.

## **Вспомогательные функции**

### **Копирование**

Функция копирования работает точно так же, как операции «Вырезать» и «Вставить» в текстовом процессоре.

В соответствии с этим последовательность копирования:

1. Выделение набора графических элементов может.
2. Сохранение в буфере.
3. Указание места куда должны быть вставлены графические элементы того же или любого другого чертежа.

#### 4. Вставка.

Выбор графических элементов производится путем обведения их прямоугольником нужного размера или другим способом. Прямоугольник рисуется на экране точно так же, как и при задании окна просмотра. Графические элементы, пересекаемые границами прямоугольника, могут по желанию пользователя быть включены в копируемый набор или исключены из него. Курсор устанавливается в той точке, куда должны быть вставлены выбранные объекты.

Функция копирования удобна в том случае, если на чертеже есть повторяющиеся элементы, например, фрагменты, содержащие сплайны или другие кривые, повторение которых трудоемко. Копирование полезно и при разработке чертежей деталей, потому что вы можете скопировать часть чертежа устройства в целом, после чего уточнить получившийся чертеж.

Частным случаем копирования является зеркальное отражение, которое позволяет строить формы, обладающие осевой симметрией. Эта функция полезна при построении объектов с одной или несколькими осями симметрии. Многие системы автоматизированной разработки чертежей предоставляют дополнительные функции, располагающие повторяющиеся объекты упорядочение.

#### **Дополнительные функции**

Помимо функций, описанных в предыдущих разделах, имеются вспомогательные функции, позволяющие обновлять чертежи в базе данных, получать чертежи из базы данных и составлять списки материалов.

#### **Совместимость файлов чертежей**

Ранее мы показали, что настоящим преимуществом системы автоматизированной разработки чертежей является возможность сохранения файла чертежа в базе данных, откуда его смогут получить сотрудники разных отделов. Этим достоинством легко воспользоваться на практике, если все сотрудники работают в одной системе автоматизированной разработки чертежей и не испытывают проблем с чтением файлов, сделанных другими сотрудниками. Однако преимущество легко утратить, если разные отделы в одной и той же компании работают с разными системами, не способными читать файлы друг друга. Проблема становится еще более серьезной, если речь идет о системах разных производителей. В этом случае единственным реальным методом взаимодействия будет построение

чертежей на бумаге и механическое их воспроизведение подобно тому, как это делалось в прошлом.

Чтобы избежать этой проблемы, можно потребовать от всех производителей систем автоматизированной разработки чертежей сохранения файлов в стандартном формате.

Наиболее популярным стандартным форматом в настоящий момент является:

- Initial Graphics Exchange Specification (IGES), принятый Американским Национальным институтом стандартов (American National Standards Institute -ANSI),
- формат DXF, DWG,
- STEP,
- SAT,
- AGS.

Практически все коммерческие системы автоматизированной разработки чертежей поддерживают формат IGES. Следовательно, файлы, созданные в одной системе, могут быть перенесены в другую систему. Однако для некоторых символов проблема корректного переноса еще не решена.

Формат DXF — формат чертежей AutoCAD, который становится стандартом де факто благодаря популярности AutoCAD.

## Лекция 3

**Тема:** Терминология и основные понятия графического моделирования

**Цель:** Изучить терминологию и основные понятия графического программирования.

### **План:**

Введение

1. Графические библиотеки
2. Изображение объекта на экране графического устройства
3. Окно и видовой экран.
4. Дисплейный файл

### **Введение.**

Важнейшей составляющей систем CAD - Автоматизированной разработки чертежей и Объемного геометрического моделирования является программное обеспечение, создающее графическое изображение на экране монитора.

Студент инженерной специальности со временем станет пользователем одной из систем автоматизации конструирования. В руководствах пользователя, поставляемых с CAD, обычно уделяют основное внимание пользовательскому интерфейсу и синтаксису, предполагается, что пользователь уже имеет определенную теоретическую подготовку. Поэтому необходимо изучить терминологию и основные понятия графического программирования.

Фундаментальные знания помогают студенту быстро изучить любую конкретную систему и использовать ее максимально эффективно.

Основными компонентами, системы CAD являются:

Аппаратное;

Программное обеспечение, в контексте лекции, служит для интерактивного манипулирования геометрическими примитивами.

Основа аппаратного обеспечения – графическая подсистема.

Графическое устройство состоит из дисплейного процессора; устройства отображения, или дисплейного устройства (монитора), и одного или нескольких устройств ввода.

Вывод графического изображения геометрических объектов на экран выполняется дисплейным процессором (сейчас специализированный набор микросхем, установленной в видеоадаптере, например, от фирмы NVIDIA), что отличает графику от, например, текста или интерфейса программ, для отображения которых нет необходимости в специальном дисплейном процессоре, так как они

выводятся средствами Windows. Однако, ряд систем для работы с 3D графикой (например, 3D Studio) используют для обеспечения повышенного качества графики и других специальных задач, собственные программные решения.

Дисплейный процессор получает коды графических команд, генерирует электронные пучки и направляет их в нужное место монитора, порождая желаемое изображение.

В зависимости от принципа функционирования графические устройства подразделяются на:

Растровые;

Векторные (применяются все реже).

Аппаратному обеспечению уделялось достаточно внимания в курсе Информатики. Далее остановимся подробно на программном обеспечении используемом для решения задач компьютерной графики.

## Графические библиотеки

Термин *программирование на компьютере (computer programming)* означает написание текста на языке компьютерных команд в соответствии с predetermined правилами грамматики. Результатом выполнения программы является ответ на вводимые числа на терминале или в файле данных в виде ожидаемых чисел и символов.

В наши дни на входе и выходе все чаще находится графическая информация.

Программирование на входе и выходе, которого графическая информация называется *графическим (graphics programming)*, а область его применения *компьютерной графикой (computer graphics)*.

Помимо основного системного программного обеспечения, необходимого для обычного программирования (операционная система, редактор и компилятор), графическое программирование требует наличия специальных графических программ.

Системные графические программы делятся на два класса:

- драйверы устройств;
- графические библиотеки.

*Драйвер графического устройства* – системное программное обеспечение, как правило, определенного производителя, позволяющее управлять конкретным графическим устройством.



Драйвер устройства может рассматриваться как набор аппаратно-зависимых кодов, непосредственно управляющих процессором графического устройства.

Драйверы обязательно являются аппаратно-зависимыми, то есть жестко привязанными к конкретным графическим процессорам, см. Рис.1. Такую графическую программу при переходе на другое графическое устройство придется переписывать с использованием новых команд драйвера. Команды драйвера устройства весьма примитивные, поэтому, если нужно решать какую-либо сложную задачу, такая программа получится очень длинной. Такие программы плохо читаемы и обычно пишутся на языках программирования типа ассемблер.

Программисты предпочитают писать программы на языках высокого уровня. Графическое программирование не могло стать исключением.

Поэтому для того чтобы упростить графическое программирование с графическими устройствами стали поставляться *библиотеки*, получившие название *графических (graphics libraries)*.

*Графическая библиотека* представляет собой набор подпрограмм, предназначенных для решения определенных задач, в частности, может изображать на экране прямую, круг или иной графический объект.

Графическая библиотека основывается на командах драйвера устройства.

Каждая подпрограмма графической библиотеки создается с использованием поддерживаемого набора команд драйвера. Например, подпрограмма, изображающая круг, может быть составлена из отдельных команд драйвера, рисующих на экране точки или короткие отрезки, аналогично тому, как вызываются функции синуса и косинуса, когда программисту требуется вычислить их значения см. Рис. 2.

Рис. 1 Непосредственное использование драйвера устройства

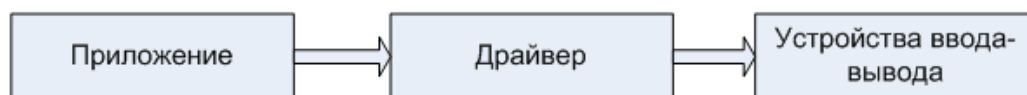
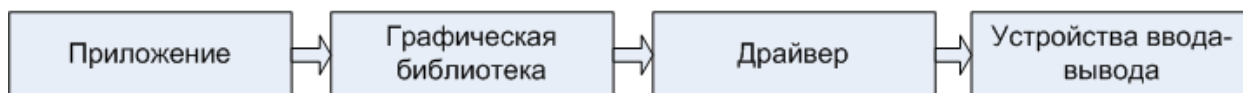


Рис. 2 Использование графической библиотеки



Одна графическая библиотека могла бы работать со всеми существующими устройствами, если бы все существующие драйверы устройств поддерживали ее. Однако по некоторым причинам производители программного обеспечения не хотят

или не могут создать такую графическую библиотеку. Таким образом, конкретная библиотека может работать лишь с ограниченным набором графических устройств.

Среди графических библиотек наиболее широко распространены:

- OpenGL,
- DirectX/Direct3D.

Коммерческая графическая библиотека OpenGL развивалась независимо от организации по стандартизации, однако с течением времени она набрала все большую популярность благодаря гибкости в управлении рабочими станциями и персональными компьютерами, в том числе в сетевой среде.

OpenGL - расширение графической библиотеки GL (фирменной графической библиотеки для компьютеров Silicon Graphics, — компании, производящей графические рабочие станции, такие как Indigo). Благодаря популярности компьютеров Silicon Graphics в областях, связанных с компьютерной графикой, библиотека OpenGL постепенно приобретает статус графического стандарта де-факто.

DirectX и Direct3D - разработки фирмы Microsoft ориентированы на работу графических приложений на платформе Windows. DirectX аналогична OpenGL библиотека, но в большей степени ориентирована на работу с игровой графикой, поэтому позволяет реализовать также звуковое сопровождение и поддержку специальных внешних устройств, используемых игровыми программами.

Подпрограммы OpenGL, DirectX и Direct3D проводят расчеты с помощью видеокарты или процессора. Поэтому говорят о графических процессорах и устройствах с аппаратной поддержкой на уровне микросхем (например, видеокарты GeForce7600) и драйвера.

## **Изображение объекта на экране графического устройства**

Вывод изображения объекта на экран графического устройства предполагает решение следующие основные задачи, которые решаются с помощью графической библиотеки:

- Указание положения всех точек объекта в пространстве;
- Определение положения и цвета точек объекта на мониторе.

Для задания положения точек в пространстве и на мониторе используются системы координат. Важно понимать, как связаны между собой различные системы координат.

## Определение положения точек объекта на мониторе

Положение точки на плоском экране позволяют задавать:

- Виртуальная;
- Обычная система координат устройства.

## Указание положения всех точек объекта в пространстве

Для работы с трехмерным пространством используются:

- Внешняя система координат (world coordinate system);
- Система координат модели (model coordinate system);
- Система координат наблюдателя (viewing coordinate system).

*Система координат устройства (device coordinate system)* - система координат, которая определяет положение точки на экране. Эта система состоит из горизонтальной оси  $u$  и вертикальной оси  $v$ .

Свойства:

- начало отсчета может выбираться произвольно, для разных графических устройств устанавливается достаточно произвольно,
- положение любой точки задается двумя целыми числами  $u$  и  $v$ , равно числу пикселей между началом координат и точкой по осям  $u$  и  $v$ ,
- третья ось, перпендикулярная первым двум осям  $u$  и  $v$ , не определяется,
- аппаратные координаты в графической программе могут потребовать изменения при смене графического устройства.

*Виртуальная система координат устройства (virtual device coordinate system)*  
Виртуальная система координат устройства фиксирует точку отсчета, направление и масштаб осей для всех рабочих станций. Слово «виртуальный» означает, что данная система отсчета существует только в воображении программиста.

Свойства:

- обычно начало отсчета располагается в левом нижнем углу экрана, ось  $u$  откладывается вправо, а ось  $v$  — вверх,
- обе координаты могут изменяться в диапазоне от нуля до единицы,.
- точка, положение которой задается в виртуальной системе координат, на любом экране будет попадать в одно и то же место.

Эти свойства дают программисту возможность единообразно определять формы, не заботясь о конкретных системах координат устройств.

### **Связь между виртуальной и системой координат устройства.**

Графическая программа(CAD) и передает виртуальные координаты подпрограмме драйвера устройства, которая преобразует их в координаты конкретного устройства.

*Внешняя, или мировая система координат (world coordinate system), — это опорная система, используемая для описания интересующего нас мира. Внешней она является по отношению к объектам этого мира.*

Например, такая система может использоваться для описания расположения и ориентации парт, стульев и доски, если интересующий нас мир представляет собой класс.

*Система координат модели (model coordinate system) - система координат, связанная с некоторыми характеристическими точками объекта, форма каждого объекта мира определяется координатами по отношению к этим характеристическим точками.*

Свойства:

- Координаты точек объекта не изменяются даже тогда, когда объект перемещается или вращается в пространстве.
- Координаты точек объекта зависят только от формы самого объекта.
- Система координат модели перемещается вместе с тем объектом, к которому она привязана.
- Если объектов несколько, форма каждого объекта определяется в его собственной системе координат.

### **Связь внешней и модельной систем координат**

Расположение и ориентация любого объекта задаются относительным положением и ориентацией модельной системы координат данного объекта по отношению к внешней системе координат.

Внешняя система координат и модельные системы для всех объектов полностью определяют мир, то есть расположение и форму всех объектов данного мира.

### **Проецирование трехмерных объектов или их точек на монитор.**

*Проецирование* в компьютерной графике - отображения трехмерных объектов на плоский экран монитора.

В проекционном черчении используется два вида проекций:



Свойства:

- Начало этой системы координат располагается в рассматриваемой точке,
- Ось  $Z_v$  направлена из начала координат в точку зрения,
- Ось  $Y_v$  параллельна вертикальной оси экрана (см. рис. Расчет проекции).
- Ось  $X_v$  определяется как векторное произведение первых двух,
- Положение точки зрения и точки наблюдения также задается в мировых координатах.

После определения наблюдательской системы координат и вычисления координат точек всех интересующих нас объектов остается только вычислить положение их проекций на экране.

Координаты интересующей нас точки  $X_v$   $Y_v$   $Z_v$  легко вычислить из подобия треугольников:

$$\begin{aligned} X_s &= \frac{S}{L - Z_v} X_v; \\ Y_s &= \frac{S}{L - Z_v} Y_v \end{aligned} \quad (1.1)$$

где  $X_s$  и  $Y_s$  — расстояния до проекции выбранной точки. Расстояния измеряются в горизонтальном и вертикальном направлениях от точки, где ось  $Z_v$  пересекается с экраном,  $L$  — расстояние между точкой наблюдения и центром проекции, а  $S$  — расстояние между центром проекции и экраном.

Рис. 4 Преобразования между системами координат



На рис. 4 приведены преобразования, обычно выполняемые внутри графической библиотеки. Программисту необходимо только указать сведения необходимые для преобразований.

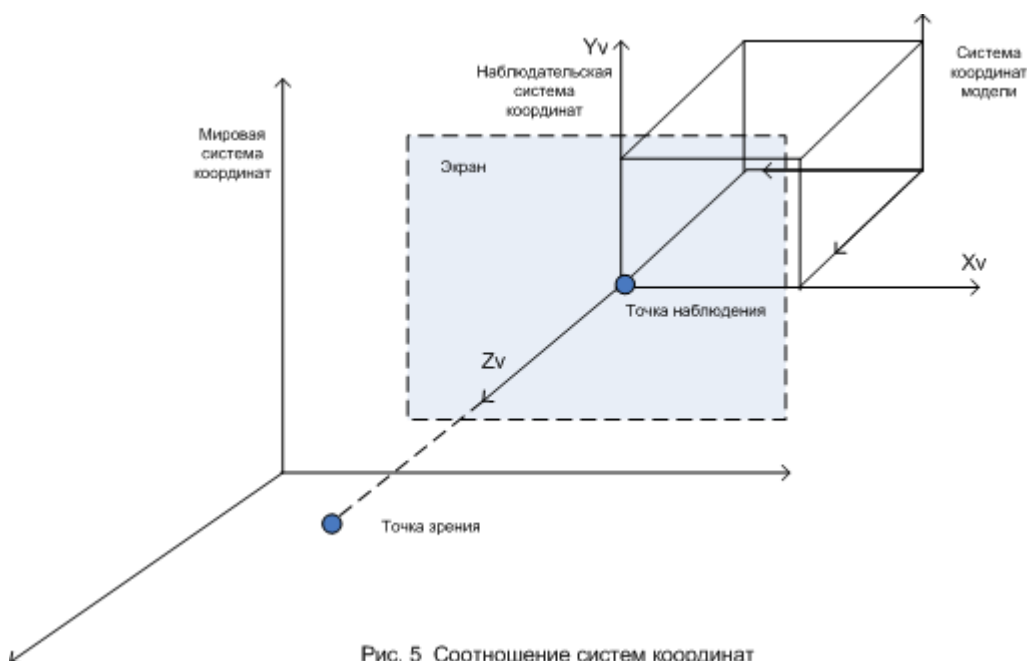


Рис. 5 Соотношение систем координат

## Окно и видовой экран

*Окно (window)* — это область пространства, проецируемая на монитор. Объекты, находящиеся вне окна, на мониторе не проявляются.

В сетевой компьютерной среде этот термин имеет иное значение и обозначает область экрана монитора рабочей станции, посредством которого пользователь взаимодействует с вычислительными ресурсами, подключенными к той же сети.

Окно определяется как прямоугольник, лежащий на экране и заданный значениями  $X_v$  и  $Y_v$  в системе координат просмотра.

*Просматриваемым объемом (viewing volume)* называется видимая область пространства, которая зависит от типа проекции. Для параллельной проекции эта область имеет форму параллелепипеда, а для перспективной — форму пирамиды.

*Видовой экран (viewport)* — это область экрана, куда отображаться проецируемое изображение (просматриваемый объем)(см. рис. 6).

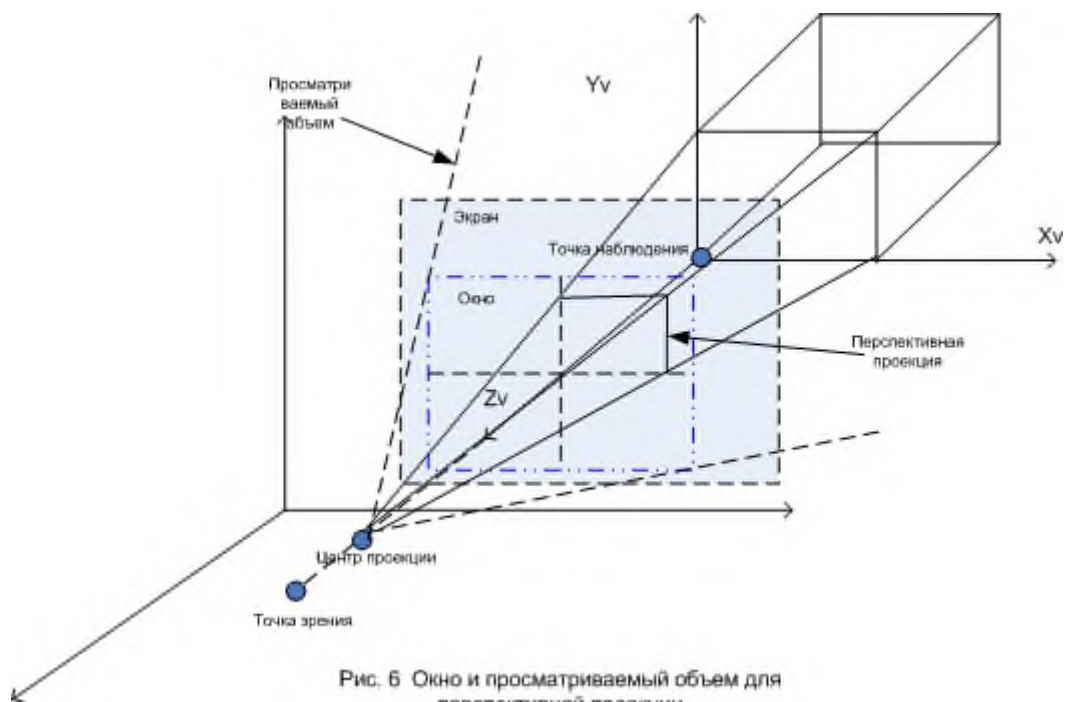


Рис. 6 Окно и просматриваемый объем для перспективной проекции

Вообще говоря, большинство графических программ поддерживают многозадачную и многооконную работу, поэтому на одном экране графического устройства может одновременно размещаться сразу несколько окон с видовыми экранами (см. рис. 7). Отсюда следующие свойства видового экрана, обеспечивающие качественное отображение.

Свойства:

1. Видовой экран задается координатами окна (области монитора, через которую пользователь взаимодействует с компьютером). Окно открывается и обрабатывается диспетчером операционной системы
2. Размеры проецируемого на видовой экран изображения зависят от параметров окна в компьютерном смысле.
3. Значения  $X_s$  и  $Y_s$ , полученные по формулам (1.1), должны быть увеличены или уменьшены таким образом, чтобы центр окна попадал в центр видового экрана, а не в центр монитора
4. Значения  $X_s$  и  $Y_s$  должны быть подвергнуты масштабированию таким образом, чтобы четыре граничные точки окна совпадали с четырьмя граничными точками видового экрана.
5. Соотношение сторон у окна должно быть таким же, как и у видового экрана, в противном случае изображение будет искажено, и круг, например, превратится в эллипс



Программный код графической библиотеки OpenGL, задающий окно и видовой экран состоит из трансляции и масштабирования, учитывающих расстояние между центром видового экрана и центром монитора, а также разницу размеров окна и видового экрана.



Рис. 7 Видовые экраны

## Дисплейный файл

*Дисплейный файл (display list)* — файл, в котором хранится для последующего выполнения группа команд графической библиотеки.

Чтобы поместить графические элементы в дисплейный файл необходимо:

1. Открыть дисплейный файл(`glNewList`);
2. Поместить список команд (драйвера/графической библиотеки) работы с графикой (отрисовка отрезков, круга, трансляция и т.д.);
3. Закрыть дисплейный файл(`glEndList`).

Технология создания дисплейного файла берет свое начало со времен, когда для работы с графикой использовались только дорогие графические станции. То есть не каждый компьютер был способен выполнять расчеты для интерактивной работы с графикой. Команды дисплейного файла позволяли отображать примитивы, составляющие рисунок на дисплей компьютера, который в принципе не предназначен, для того чтобы проводить соответствующие расчеты.

В настоящее время все современные компьютеры используют графические библиотеки и позволяют работать с графикой, поэтому проблема создания и использования дисплейных файлов не стоит так остро как раньше.

Дисплейный файл OpenGL ориентирован на оптимизацию производительности

Оптимизация обеспечивается благодаря тому, что дисплейный файл хранится в виде списка команд (при запуске дисплейного файла графическое устройство

работает как компилятор), а не в виде динамической базы данных (при запуске дисплейного файла графическое устройство - интерпретатор).

Свойства:

- созданный дисплейный файл изменить уже нельзя;
- дисплейные файлы, как правило, работают значительно быстрее, чем обычные последовательности команд графической библиотеки, не объединенные в группы;
- дисплейные файлы позволяют повысить производительность при работе в сети за счет сокращения сетевого трафика;
- к созданному дисплейному файлу могут быть применены следующие операции:
  - множественное выполнение (multiple execution) — один и тот же файл можно выполнять много раз;
  - иерархическое выполнение (hierarchical execution) — иерархическим называется дисплейный файл (родительский), вызывающий другие дисплейные файлы (дочерние). Иерархические дисплейные файлы удобны для объектов, состоящих из отдельных компонентов, особенно если некоторые компоненты входят в объект в нескольких экземплярах;
  - удаление (deletion) — дисплейный файл может быть удален.

На практике дисплейные файлы вызываются программно (без участия пользователя) при отображении первых 2/3 операций моделирования. То есть создают дисплейные файлы разработчики САПР для того, чтобы ускорить процесс отката при редактировании модели пользователем.

## Лекция 4

**Тема:** Операции преобразования в графическом моделировании.

**Цель:** Ознакомиться с основными видами преобразований объектов

**План:**

1. Преобразования геометрии
2. Преобразование трансляция
3. Преобразование вращение
4. Преобразование масштабирование
5. Преобразование зеркальное отображение
6. Визуализация объемных моделей

### Преобразования геометрии

В CAD новое положение модели обычно задается через повороты и смещения относительно исходного положения. Для интерактивного манипулирования координаты модели необходимо преобразовывать.

Преобразования координат модели позволяют реализовать такие функции панели инструментов «Вид» как:

- Изменение в размер экрана,
- Масштабирование выбранной с помощью рамки области,
- Увеличение/уменьшение масштаба при перетаскивании указателя вверх/вниз,
- Увеличение выбранного объекта,
- Вращение вида,
- Перемещение вида,
- Обзор модели через одну из камер.

Без указанных функций трудно себе представить современное проектирование. Убедиться в полезности указанных функций можно на практике.

Каждому преобразованию координат графического объекта в графической библиотеке (OpenGL) соответствует математическое действие над матрицами. Действия реализуются в библиотеке в виде соответствующих команд.

По соглашениям о записи в OpenGL в матричной форме представлены:

- исходная координата объекта – начального положения,
- новая координата объекта,
- преобразование, позволяющее получить новую координату объекта.

Пересчет координат точек объекта из системы модели в мировую систему состоит в следующем:

- Совмещение модельной системы координат с мировой (поэтому, проектируя модель, лучше совместить одну из ее граней с мировой системой координат).
- Трансляция (сдвиг) и/или поворот точек модели из исходного положения в мировой системе координат в ее новое положение при ее перемещении и/или повороте.
- Вычисление координат точек в наблюдательской системе по мировым координатам всех точек модели в новом положении.

**Замечание:** Все преобразования, о которых пойдет речь ниже не изменяют действительного объема модели(детали), ее размеров, но позволяют ее лучше видеть.

Далее мы рассмотрим матрицы преобразования, действующие на точки объекта при его перемещении, повороте, масштабировании и зеркальном отражении.

Большинство графических библиотек выполняют преобразования координат самостоятельно. Проектировщик задает только величину смещение и/или поворот для интересующего его объекта (с помощью функций панели «Вид»). Несмотря на это проектировщику все равно нужно знать законы преобразований, чтобы рисовать объекты в нужных местах без проб и ошибок, в особенности, если эти объекты перемещаются достаточно сложным образом.

## Трансляция

Трансляция (сдвиг) объекта из исходного положения с координатами  $X_m, Y_m, Z_m$  осуществляется на величины  $a, b$  и  $c$  в направлениях  $x, y$  и  $z$  соответственно. В исходном положении модельная система  $X_m, Y_m, Z_m$  координат совпадает с мировой  $X_w, Y_w, Z_w$  (рис. 1). Мировые координаты точек объекта в новом положении  $X_w, Y_w, Z_w$  вычисляются следующим образом:

$$\left. \begin{aligned} X_w &= X_m + a \\ Y_w &= Y_m + b \\ Z_w &= Z_m + c \end{aligned} \right\} (1.2)$$

В этой формуле числа  $X_m, Y_m, Z_m$  являются также модельными координатами точки.

Формула (1.2) может быть записана в матричной форме:

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} \quad (1.3)$$

Полученная, матрица однородного преобразования (т.к. ее можно только умножать) - матрица преобразования транслирования (сдвига) (homogeneous transformation matrix).

Матрица преобразования (в частности, трансляции) точки в двумерном пространстве представляет собой однородную матрицу преобразования размерностью 3x3, полученную удалением третьей строки и третьего столбца из матрицы размерностью 4x4. Новая матрица действовала бы на вектор координат размерностью 3x1, полученный из вектора 4x1 удалением z-координаты.

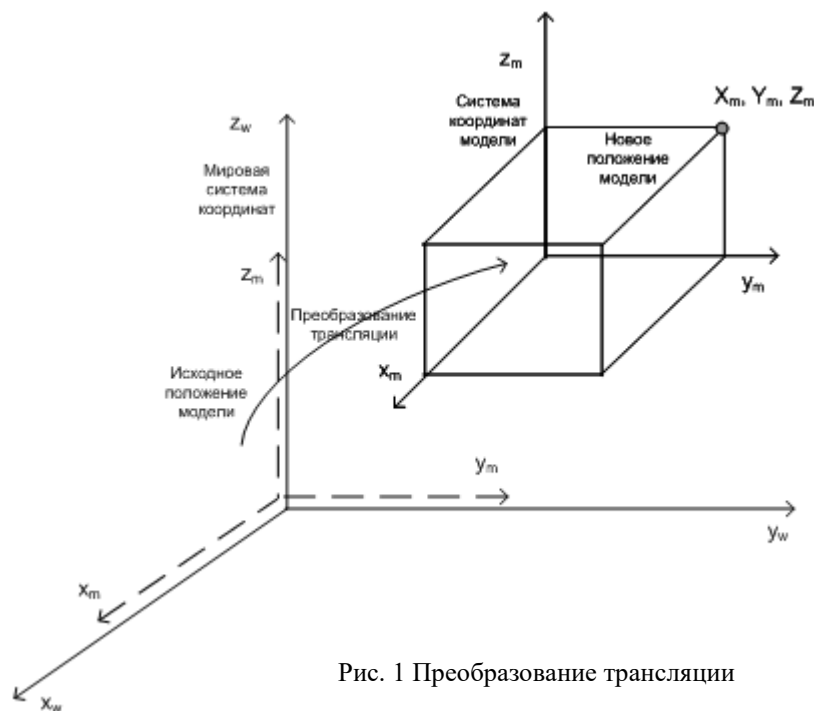


Рис. 1 Преобразование трансляции

## Вращение

Вращение объекта из исходного положения с координатами  $X_m, Y_m, Z_m$  осуществляется на угол  $\phi$  вокруг оси  $x_w$  в мировой системе координат  $x_w, y_w, z_w$  вместе с модельной системой, которая, как и в предыдущем случае в исходном положении совпадает с мировой  $x_w, y_w, z_w$  (рис. 2).

Мировые координаты точек объекта в новом положении  $X_w, Y_w, Z_w$  можно получить:

$$X_w = X_m; \quad (1.4)$$

$$\begin{aligned}
 Y_w &= l \cos(\varphi + \alpha) = l(\cos \varphi \cos \alpha - \sin \varphi \sin \alpha) = \\
 &= l \cos \varphi \cos \alpha - l \sin \varphi \sin \alpha = Y_m \cos \varphi - Z_m \sin \varphi; \quad (1.5)
 \end{aligned}$$

$$\begin{aligned}
 Z_w &= l \sin(\varphi + \alpha) = l(\sin \varphi \cos \alpha + \cos \varphi \sin \alpha) = \\
 &= l \sin \varphi \cos \alpha + l \cos \varphi \sin \alpha = Y_m \sin \varphi + Z_m \cos \varphi \quad (1.6)
 \end{aligned}$$

Равенства (1.4-1.6) для вращения вокруг оси  $x_w$  могут быть записаны в матричной форме:

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \varphi & -\sin \varphi & 0 \\ 0 & \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} \quad (1.7)$$

Равенства (1.4-1.6) для вращения вокруг оси  $y_w$  могут быть записаны в матричной форме:

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \varphi & 0 & \sin \varphi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \varphi & 0 & \cos \varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} \quad (1.8)$$

Равенства (1.4-1.6) для вращения вокруг оси  $z_w$  могут быть записаны в матричной форме:

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 & 0 \\ \sin \varphi & \cos \varphi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} \quad (1.9)$$

Подобно матрице трансляции, для двумерного объекта однородная матрица вращения редуцируется до размера  $3 \times 3$ .

Поворот вокруг любой произвольной оси раскладывается на повороты вокруг осей  $x_w, y_w, z_w$ . Таким образом, матрица преобразования для произвольной оси получается перемножением матриц (1.7-1.8).

Как уже отмечалось, матрицы преобразования, описываемые в этом разделе, обычно вычисляются соответствующими подпрограммами графических библиотек.

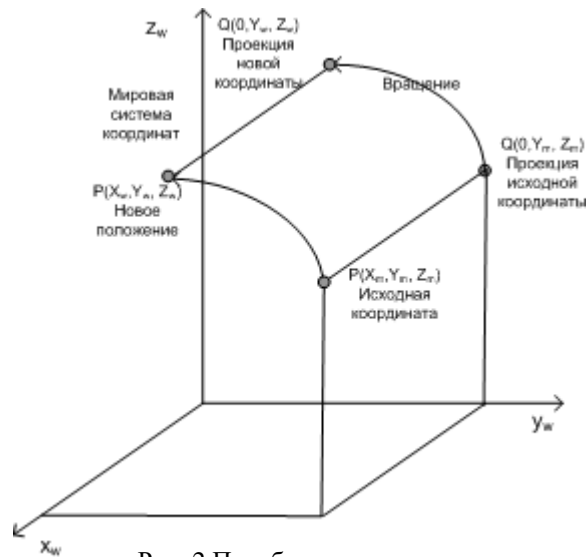


Рис. 2 Преобразование вращение

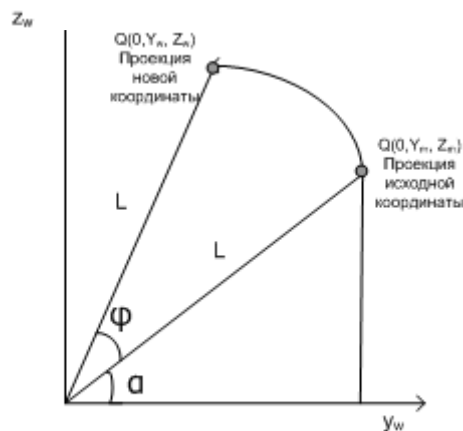


Рис. 3 Преобразование вращение.  
Проекция на плоскость ZOY

## Отображение

Получив мировые координаты всех точек объекта в его текущем положении, необходимо вычислить координаты точек в наблюдательской системе.

Примером такого преобразования является переход из системы координат Сборки( которая выступает в роли мировой системы координат) в систему координат редактируемой детали(выступает в роли наблюдательской системы координат) этой сборки.

Перевод координат из одной системы координат в другую называется *отображением (mapping)*.

Отображение состоит в вычислении координат точки в некоторой системе координат по известным координатам точки в другой системе координат.

Рассмотрим две системы координат  $x_1, y_1, z_1$  и  $x_2, y_2, z_2$  смотри рисунок.

Предположим, что необходимо вычислить координату  $x_2, y_2, z_2$  в системе координат  $x_2, y_2, z_2$  по координатам этой точки  $x_1, y_1, z_1$  в системе координат  $x_1, y_1, z_1$ .

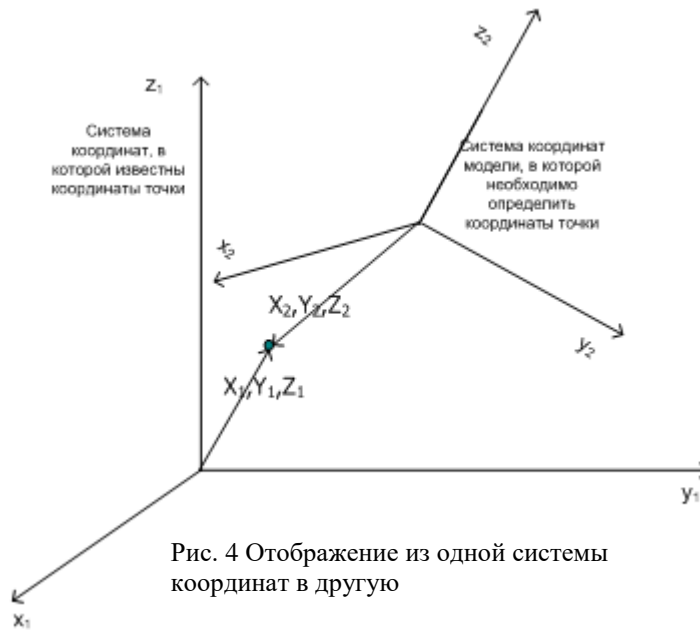


Рис. 4 Отображение из одной системы координат в другую

Вычисление координаты  $X_2, Y_2, Z_2$  производится по следующей формуле:

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix} \quad (1.10)$$

Здесь  $p_x, p_y, p_z$  определяют начало отсчета системы координат  $x_2, y_2, z_2$  в системе координат  $x_1, y_1, z_1$ ;

$n_x, n_y, n_z$  – компоненты  $x_2, y_2, z_2$  единичного вектора, направленного вдоль оси  $x_1$  системы координат  $x_1, y_1, z_1$ ;

$o_x, o_y, o_z$  – компоненты  $x_2, y_2, z_2$  единичного вектора, направленного вдоль оси  $y_1$  системы координат  $x_1, y_1, z_1$ ;

$a_x, a_y, a_z$  – компоненты  $x_2, y_2, z_2$  единичного вектора, направленного вдоль оси  $z_1$  системы координат  $x_1, y_1, z_1$ .

Пример отображения: По заданному положению точки зрения  $(-10, 0, 1)$ , точки наблюдения  $(0, 0, 1)$  и вектора вертикали  $(0, 0, 1)$  строится наблюдательская система координат (см. представленный ниже рисунок). Все координаты и компоненты векторов даны в мировой системе координат.

Зная относительное положение системы координат просмотра и мировой системы координат, необходимо рассчитать:

- Матрицу преобразования;
- Координаты точки с мировыми координатами  $(5, 0, 1)$  в наблюдательской системе координат.



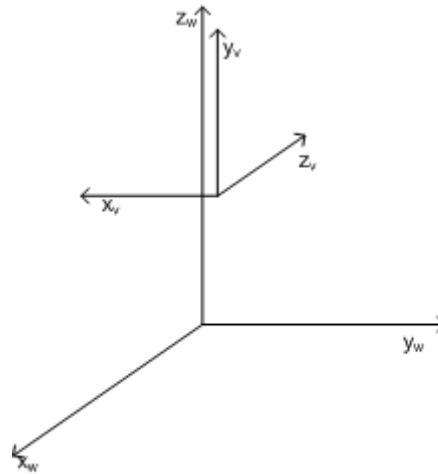


Рис. 5 Пример отображения из одной системы координат в другую

Матрица отображения для данного случая будет иметь вид:

$$\begin{bmatrix} X_v \\ Y_v \\ Z_v \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (1.11)$$

Здесь  $p_x, p_y, p_z$  равны  $(0, -1, 0)$  определяют начало отсчета системы координат  $x_w, y_w, z_w$  в системе координат  $x_v, y_v, z_v$ ;

$n_x, n_y, n_z$  равны  $(0, 0, -1)$  – компоненты  $x_v, y_v, z_v$  оси  $x_w$  (определяют ось  $x_w$  в системе координат  $x_v, y_v, z_v$ );

$o_x, o_y, o_z$  равны  $(-1, 0, 0)$  – компоненты  $x_v, y_v, z_v$  оси  $y_w$ ;

$a_x, a_y, a_z$  равны  $(0, 1, 0)$  – компоненты  $x_v, y_v, z_v$  оси  $z_w$ .

Координаты точки  $(5, 0, 1)$  в наблюдательской системе координат имеют значения  $(0, 0, -5)$ .

$$\begin{bmatrix} 0 \\ 0 \\ -5 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 5 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad (1.12)$$

## Масштабирование

Эффекта масштабирования можно достичь, изменив размеры видового экрана или окна, не меняя значений координат, с помощью команд панели инструментов «Вид». Матрица преобразования (1.13) используется при масштабировании объекта относительно начала координат.

Для масштабирования объекта с коэффициентом  $s_x$  по оси  $x$ ,  $s_y$  по оси  $y$ ,  $s_z$  по оси  $z$  применяется следующая матрица преобразования:

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1.13)$$

## Зеркальное отображение

Отражение относительно зеркальной плоскости  $xу$  может быть достигнуто при помощи приведенной ниже матрицы преобразования. Преобразование заключается в изменении знака координаты  $z$ .

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1.14)$$

Матрицы преобразования для других отражений (относительно плоскостей  $xz$  и  $yz$ ) выводятся аналогичным образом.

## Визуализация объемных моделей

Различают алгоритмы визуализации, характерные для :

- Каркасного представления,
- Тонированного,
- Фотореалистического изображения модели.

В каркасном представлении требуется

- Удалить невидимые линии(скрытые поверхностями модели);
- Изобразить невидимые линии иначе, чем видимые(пунктиром, тонкими линиями).

В тонированном изображении требуется

- Изобразить видимые грани;
- Закрасить видимые грани с учетом освещения и заданного цвета.

Как правило, используются алгоритмы Гуро и Фонга. Алгоритм Гуро дает фасетчатые поверхности, но не требователен к вычислительным ресурсам.

Алгоритм Фонга дает гладкие поверхности, но более требователен к вычислительным ресурсам.

В фотореалистическом изображении дополнительно учитываются оптические свойства материала(цвет, текстура, неровности, отражающие и преломляющие

свойства материала). В настоящее время для фотореалистического изображения используются алгоритмы рэйтрейсинга, фотометрический.

В алгоритме рэйтрейсинга рассчитывается ход световых лучей по сцене до видового экрана.

В фотометрическом алгоритме рассчитывается баланс «световой энергии» и та его часть, которая попадает на видовой экран.

На сегодняшний день фотореалистическое изображение формируется без аппаратной поддержки видеокарты, исключая графические адаптеры от фирмы NVidia торговой марки Quattro, которые имеют полностью графический конвейер.

## Лекция 5

**Тема:** Системы геометрического моделирования CAD. Геометрические модели.

**Цель:** Использование операций твердотельного моделирования

### План:

Введение

1. Классификация геометрических моделей
2. Область применения твердотельных геометрических моделей
3. Основные понятия и принцип твердотельного моделирования деталей
4. Операции твердотельного моделирования

### **Введение.**

Системы геометрического моделирования CAD были созданы для того, чтобы преодолеть проблемы, связанные с использованием физических моделей в процессе проектирования. В большинстве случаев очень сложно извлечь из реальных моделей необходимые сведения для точного их воспроизведения (например, для обработки на станке с ЧПУ).

CAD создают среду, подобную той, в которой создаются и изменяются физические модели. В системе геометрического моделирования разработчик изменяет форму модели, добавляет и удаляет ее части, детализируя форму визуальной модели таким же образом, как ребенок формирует фигурку из пластилина. Визуальная модель может выглядеть точно так же, как физическая, но она нематериальна.

Как правило, при автоматизированном проектировании и анализе изделия или технологического процесса инженер работает не с реальным изделием и даже не с его макетом, реализованным в материале, а с виртуальной компьютерной моделью, которая обладает лишь частью свойств своего прототипа.

Реализация возможностей математического моделирования средствами САПР повышает эффективность инженерных разработок в целом, позволяет экспериментировать с математической моделью, получать результаты возможно более точные, во много раз быстрее и дешевле.

В инженерной практике *компьютерная математическая модель* - математическая модель технического устройства(процесса), реализованная с помощью универсальных или специализированных САПР или расчетных программ.

В качестве **объекта моделирования** в САПР выступают технические устройства (агрегат, узел), процессы (технологический процесс обработки резанием, давлением, сварка), явления (процесс течения жидкости).

В инженерной практике применения САПР нашла свое распространение **компьютерная модель изделия**.

*Компьютерная модель изделия* – виртуальный аналог изделия, состоящий комплекса взаимосвязанных компьютерных моделей (заготовки, детали) существующих параллельно с реальной продукцией при ее проектировании и производстве.

*Конструкторская модель изделия* - это модель изделия, которая разрабатывается на этапе конструкторского проектирования.

Конструкторская модель изделия, как правило, включает все **модели деталей и сборок изделия** и относящуюся к этому этапу документацию.

### **Классификация геометрических моделей**

Все модели деталей и сборок изделия относятся к классу **геометрических моделей**. Данный класс моделей содержит информацию об элементном составе модели и связи между элементами, о форме и размерах элементов составляющих модель, об их взаимном расположении.

Конструкторские модели деталей и сборок разрабатываются с помощью CAD средствами проволочного, твердотельного, поверхностного, моделирования. В зависимости от того, какая из концепций создания геометрических моделей использована, и какой математический аппарат реализован при построении моделей, конструкторскую геометрическую модель детали или сборки можно классифицировать как:

- каркасную
- твердотельную
- поверхностную
- гибридную модель.

### **Область применения твердотельных геометрических моделей**

На сегодняшний день большинство CAD разрабатываются как системы для широкого спектра приложений (конструирование изделий быта, технологической

оснастки, строительных конструкций, проектирование цехов и т.д.), поэтому их функциональность достаточно универсальна.

Основные возможности CAD:

- разработка технической документации в ее классическом виде - чертежи;
- построение объемных моделей различных изделий;
- разработка чертежей по объемным моделям;
- получение реалистичных изображений и изготовление анимированных презентаций.

Компоненты, реализующие данные функции, объединены в едином пакете программного обеспечения (ПО) с общим рабочим полем и пользовательским интерфейсом.

Построение объемных моделей различных изделий включает:

1. Создание объемных деталей с помощью набора средств создания твердотельных(поверхностных) моделей;
2. Разработка сборочных узлов из объемных моделей деталей.

Преимущества объемной модели по сравнению с плоским чертежом:

- Наглядность модели делает ее удобной при принятии конструкторских решений,
- Визуальная ориентация моделей деталей и узлов в ходе, позволяет устранить ошибки, возникающие при сборке.
- Точность и качество получаемой модели - необходимые свойства геометрии передаваемой в расчетные пакеты CAE, САМ-пакеты для разработки УП,
- Позволяет определять объем тела, его массу, моменты инерции, центр тяжести и т.п. (Эти параметры зачастую являются критическими при оценке эффективности конструкции изделия и чрезвычайно важны для конструктора).

Создание геометрической твердотельной модели предполагает ее использования в дальнейшей расчетной и технологической практике.

Возможность расчета основных физических характеристик обусловлена основной идеей, реализованной в твердотельном моделировании: Всегда гарантировать физически непротиворечивое представление геометрических объемных тел и оперировать понятиями добавления/удаления материала. Корректное твердое тело содержит внутренний объем, ограниченный внешней поверхностью тела.

## Основные понятия и принцип твердотельного моделирования

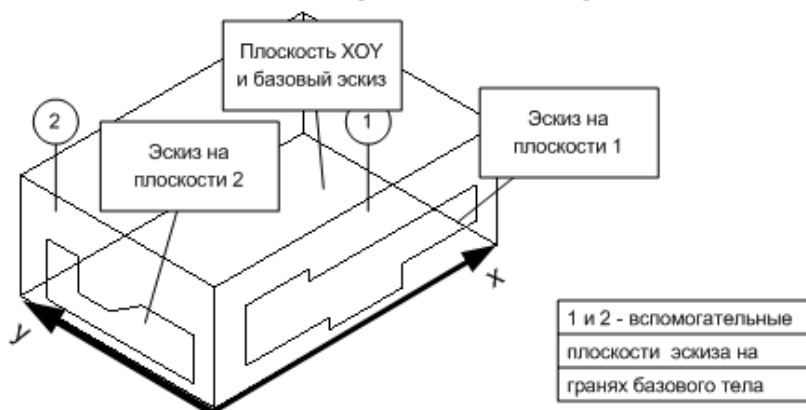


Рисунок 1

Работа по созданию объемной геометрической модели детали в большинстве систем твердотельного моделирования состоит из двух укрупненных шагов:

1. Создания эскиза(-ов);
2. Задания операций объемного преобразования.

*Эскизом* называется фигура на плоскости (координатной, вспомогательной, грани и т.д.), на основе которой в дальнейшем формируется объемное преобразование. Эскиз профиля и эскиз траектории определяют размер и форму элемента геометрической модели.

Эскиз изображается на плоскости стандартными средствами (точка, отрезок, полилиния, дуга и т.д. с учетом ограничений накладываемых на параметры). В процессе его создания формируется часть необходимых параметров для задания большинства операций (инструментов) (а,б...см. ниже).

*Операция (инструмент)* задает некоторое формообразующее перемещение эскиза. Поэтому оставшиеся параметры операции задают конечную точку и/или траекторию перемещения эскиза в угловых или линейных единицах измерения.

Форма детали может быть получена с помощью одной операции (инструмента), но, как правило, процесс создания модели включает многие операции, образующие последовательность - дерево конструирования модели.

Отсюда основной **принцип твердотельного моделирования деталей**:

Деталь создается по шагам (так называемый фичерный метод — Feature Based Modeling). Каждый такой шаг имеет целью добавление или вычитания материала детали.

Этот способ моделирования зарекомендовал себя как наиболее естественный, понятный конструкторам; практически он является стандартом для всех новых разработок в области CAD.

При создании эскизов следует помнить несколько правил:

- делайте эскиз простым;
- создавайте эскиз поэтапно;
- рисуйте каждый объект эскиза в крупном масштабе;
- избегайте без лишней необходимости использовать (рисованные фаски, вписанные дуги, рисованные отверстия, рисованные массивы элементов);
- пытайтесь создавать эскиз по следующему алгоритму:
  - в первую очередь, добейтесь принципиального соответствия эскиза замыслу
  - назначьте закрепления
  - создайте размерную схему
  - приведите эскиз к желаемому масштабу
  - назначьте точные размеры;
- эффективно используйте настройки среды в режиме Эскиз;
- избегайте в эскизах соседства несопоставимых размеров (например, 3000мм и 0,3мм);
- используйте такую размерную схему эскиза, какую желаете в дальнейшем видеть в чертеже;
- используйте вспомогательные элементы эскиза для получения желаемой размерной схемы;
- используйте кнопки возврата для редактирования эскиза;
- при необходимости, сохраняйте промежуточные варианты эскиза.

## **Операции твердотельного моделирования**

К основным операциям твердотельного моделирования относятся:

- создание типичных примитивов (брус, цилиндр, конус, шар, клин, тор);
- булевские операции теории множеств (объединение, пересечение, разность);
- заметание (вращение эскиза вокруг оси, лежащей в плоскости эскиза);
- заметание (выдавливание эскиза в направлении перпендикулярном плоскости эскиза);
- выдавливание эскиза вдоль произвольной траектории;



- лофтинг - создание тела переменного сечения;
- создание оболочек различной формы;
- создание ребер;
- создание отверстий;
- создание массивов элементов (размножение элементов на делительной окружности, размножение по числу элементов по каждому координатному направлению);
- создание скруглений, фасок;
- вычитание объемов операциями а, б, в, г;
- поднятие грани;
- моделирование границ;
- объектно-ориентированное моделирование типовых элементов (крепежных отверстий, шпоночных пазов, профилей).

### **Функции создания примитивов**

Большинство систем твердотельного моделирования позволяют пользователям получать объемы простых форм, на основе которых можно моделировать более сложные объемы.

Функции создания примитивов позволяют выбирать и создавать простейшие объекты.

Процедуры, используемые для создания примитивов, заранее заданы авторами системы моделирования.

Примитивы, поддерживаемые большинством систем твердотельного моделирования, показаны на рис. 2. Размер и положение примитива задается пользователем.

Размеры, указанные на рис. 2 буквами, могут устанавливаться пользователем. Примитивы сохраняются в базе данных процедурой, осуществляющей их создание, а параметры примитивов передаются этой процедуре в качестве аргументов.

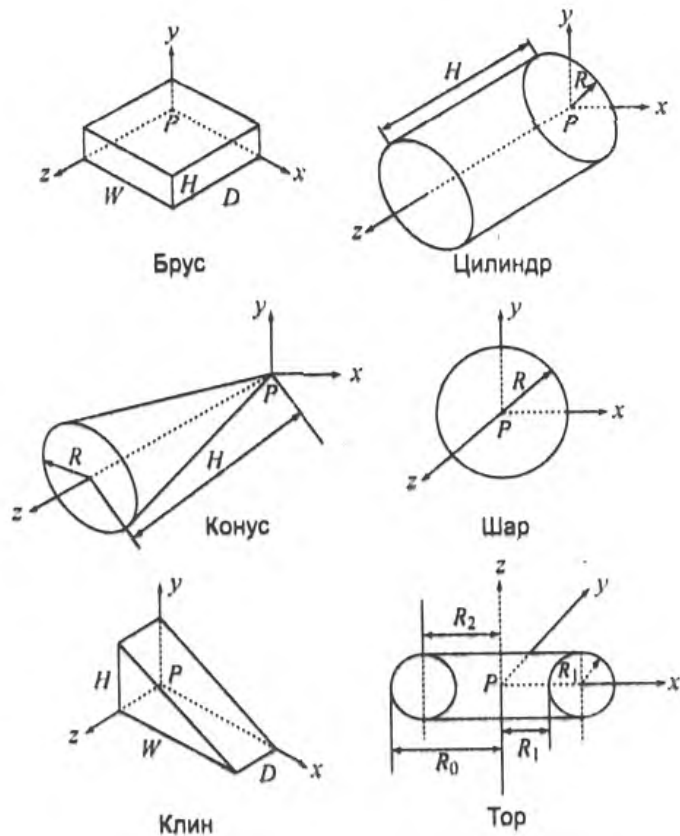


Рисунок 2

### Создание примитива

Границы объемных тел состоят из элементарных геометрических объектов: вершин, ребер и граней.

*Грань* — часть граничной поверхности, граница которой состоит из криволинейных сегментов, при пересечении которых происходит существенное изменение вектора нормали к поверхности.

Криволинейные сегменты, ограничивающие грань, называются *ребрами*.

Точки, в которых встречаются соседние ребра, называются *вершинами*.

Для построения объемных тел из этих элементарных геометрических объектов используются Операторы Эйлера.

*Операторы Эйлера* - операторы для манипулирования (создания/удаления) небольшими группами элементов, удовлетворяющих уравнению Эйлера-Пуанкаре.

Уравнение Эйлера-Пуанкаре:

$$v - e + f - h = 2(s - p).$$

Где  $v$  - количество вершин;

$e$  - количество ребер;

$f$  – количество граней;

$h$  – количество колец отверстий;

$s$  – количество оболочек;

$p$  – количество сквозных проходов через отверстия в теле.

Использование операторов Эйлера позволяет устранить следующее противоречие: Манипулирование отдельными элементами оказывается малоэффективным, поскольку в большинстве случаев добавление или удаление элемента приводит к изменению состояния других элементов (Например добавление диагонального ребра приводит к делению исходной грани на две).

Большинство примитивов можно создать с помощью функций моделирования производящих трансляционное или вращательное замещение. Эти процедуры обычно состоят из операторов Эйлера, производящих трансляционное или вращательное замещение (операторы Эйлера используются для хранения примитивов в системах твердотельного моделирования). В свою очередь параметрами операторов Эйлера являются координаты элемента в системе координат модели. Расчет координат был рассмотрен в предыдущей лекции в пункте «Преобразования геометрии (трансляция, вращение...)».

Рассмотрим создание параллелепипеда с помощью операторов Эйлера, используемых в системе твердотельного моделирования SINUMOD (см. рис.3):

1. MEVLS(B, &E1, &V1, &V2, &L1, &S1, X1, Y1, Z1, X2, Y2, Z1) - Добавление пустоты к существующему объекту(выделяется место в памяти под оболочку, объявляется о ее создании);

Входные-выходные аргументы: элементы, перед которыми стоит символ &, являются выходными элементами, генерируемыми оператором MEVLS, а остальные элементы являются входными. Поэтому данное выражение можно интерпретировать так, что на объеме B создаются новое ребро E1, новые вершины V1 и V2 в точках (x1, y1, z1) и (x2, y2, z2), новое внешнее кольцо L1 и новая оболочка S1. Тот же способ записи мы будем использовать для различения входных и выходных аргументов в других операторах Эйлера.

2. MEV(B, L1, V1, &E1, &V2, X, Y, Z) - Оператор создает ребро E1, проходящее от вершины V1 кольца L1 до заданной точки (x,y,z) и добавляет это ребро к кольцу L1, в заданной точке создается вершина V2.

3. MEL(B, L1, V1, V2, &E1, &L2) - Оператор делит исходное кольцо L1 на два новых L1и L2(например, кольцо отверстия)

4.  $KEV(B, \&L1, \&V1, E2, V3, \&X, \&Y, \&Z)$ - Оператор удаляет ребро  $E2$  вершину  $V3$  (необходим для удаления лишних вершин и ребер, которые неизбежно придется создавать).

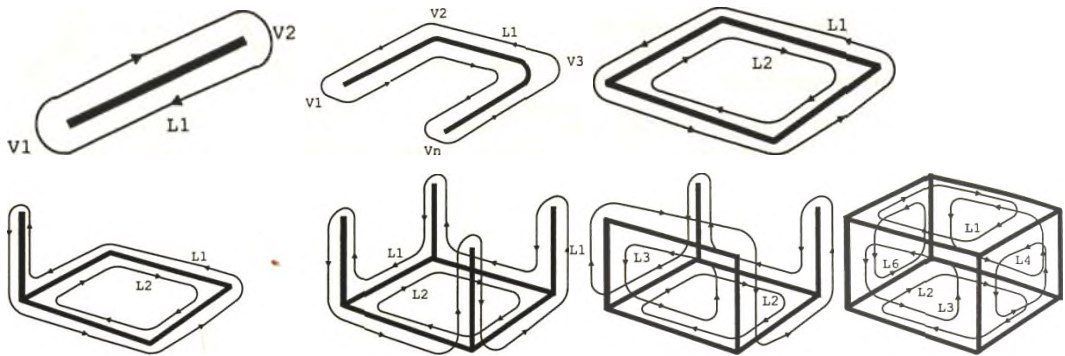


Рис. 3 Иллюстрация применения операторов Эйлера

### Булевские операции

В списке объемных примитивов (см. выше) нельзя найти любое объемное тело, поскольку сохранить заранее все мыслимые формы невозможно из-за разнообразия возможных применений систем геометрического моделирования.

Гораздо проще приблизиться к решению, предоставив пользователю инструмент для комбинирования примитивов. В качестве метода комбинирования в твердотельном моделировании применяются булевские операции теории множеств.

Согласно теории множеств каждое объемное тело считается множеством точек. К множествам применимы булевские операции (сложения, вычитания, пересечения см. Рис.4). В результате использования булевской операции получается объемное тело, состоящее из точек, полученных после преобразований.

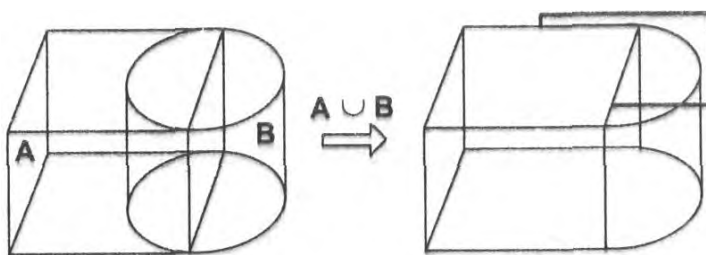


Рисунок 4.1 Объединение примитивов

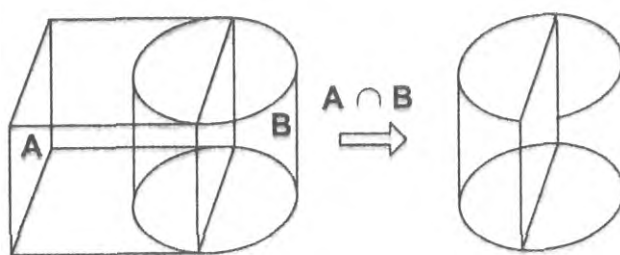


Рисунок 4.2 Пересечение примитивов

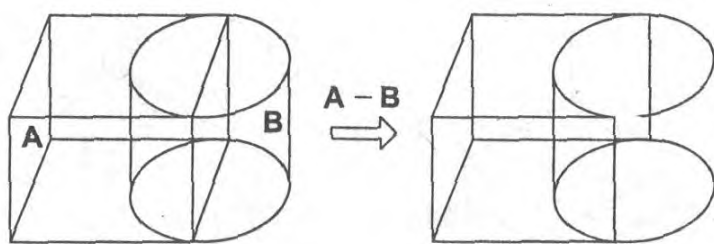


Рисунок 4.3 Разность примитивов

### **Заметание**

Заметание одна из функций, используемых для создания простых форм на основе объемных заготовок, имеющихся в программе, — так называемых функций создания примитивов (primitive creation functions). Примитивы сохраняются в базе данных процедурой, осуществляющей их создание, а параметры примитивов передаются этой процедуре в качестве аргументов.

Функция заметания (sweeping) позволяет создавать объемное тело трансляцией или вращением области, заданной на плоскости.

Построение тела вращения из плоской кривой называется *вращательным заметанием (swinging)*.

Задавая замкнутую плоскую область (эскиз), пользователь может указывать геометрические ограничения или вводить данные о размерах, а не рисовать форму вручную. Здесь под геометрическими ограничениями понимаются соотношения между элементами эскиза (перпендикулярность отрезков, касание дуги окружности отрезком и т. д.). В этом случае система построит точную форму, удовлетворяющую ограничениям, самостоятельно.

### **Лофтинг**

Функция лофтинга (lofting) создает объемное тело, натягивая поверхность на заданные поперечные сечения.

Функции позволяют проектировщику начать моделирование с формы, весьма близкой к конечному результату, поскольку одних поперечных сечений вполне достаточно для точного описания конечного объемного тела.

### **Скругление или плавное сопряжение**

*Скругление (minding), или плавное сопряжение (blending)* - операция модифицирования существующей модели, состоящего в замене острого ребра или вершины гладкой криволинейной поверхностью.

Поверхность сопряжения - гладкая криволинейная поверхность, векторы нормали к которой совпадают с векторами нормалей поверхностей, сошедшихся у исходного ребра или вершины.

Замену острого прямого ребра цилиндрической поверхностью демонстрирует рис.5,а. На рисунке векторы нормали к цилиндрической поверхности совпадают с векторами соседних плоских граней.

Замена острой вершины сферической поверхностью показана на рис.5 а. Здесь также обеспечивается непрерывность векторов нормали.

Выкружка (filleting) - частный случай процедуры скругления с добавлением, а не удалением материала (показан на рис. 5 б).

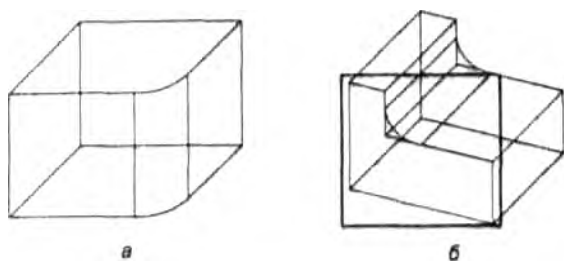


Рисунок 5

### Поднятие

*Поднятие (lifting)* - процедура перемещения всей грани объемного тела или ее части в заданном направлении с одновременным удлинением тела в этом направлении (рис. 6).

Если нужно поднять только часть грани (рис. 6, б), эту грань необходимо заранее разделить. Для этого достаточно добавить ребро, которое будет разбивать грань на нужные части. Но внутри системы при этом будут произведены некоторые дополнительные действия, результатом которых станет деление грани. Обычно при этом обновляются сведения о связности поверхности. Внутри системы подобные действия выполняются при помощи операторов Эйлера.

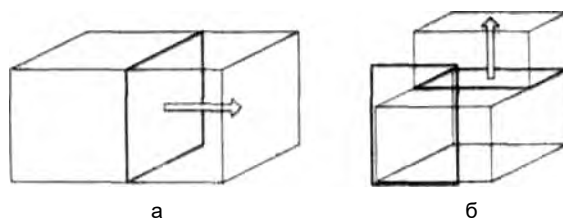


Рисунок 6

При работе с функцией поднятия необходимо правильно указывать направление и дальность поднятия, чтобы добавленная часть тела не пересекалась с исходной.

Поднятие используется для смещения отдельных поверхностей детали, сохраняя замкнутый объем. Типичная область применения - гибридные модели, состоящие из импортированных поверхностей, не имеющих параметризации и твердотельных элементов.

### **Моделирование границ**

Функции моделирования границ используются для добавления, удаления и изменения элементов объемного тела — его вершин, ребер и граней.

Процедура, использующая функции моделирования границ, будет выглядеть точно так же, как в системах поверхностного моделирования:

1. Вначале создаются точки,
2. Затем создаются ребра, соединяющие эти точки,
3. Наконец, граничные ребра определяют поверхность.

Однако, в системах твердотельного моделирования, в отличие от систем поверхностного моделирования, нужно определить все поверхности таким образом, чтобы образовался замкнутый объем.

Создание клина при помощи функций моделирования границ иллюстрирует рис. 7. Процедура включает создание точек, граней и поверхностей.

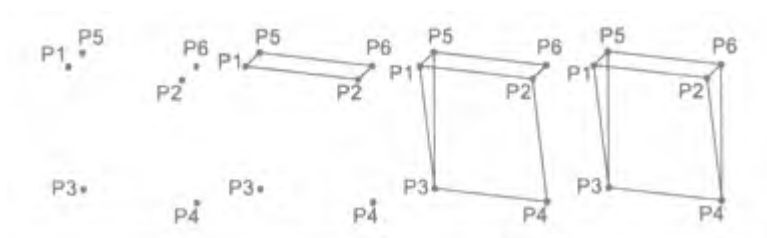


Рисунок 7 Создание тела при помощи функций моделирования границ

Области применения:

- Создавать объемное тело исключительно при помощи функций моделирования границ очень утомительно.
- Моделирование границ используется главным образом для создания плоских фигур, которые затем служат сечениями объемным телам, образуемым заметанием или лофтингом.
- Функции моделирования границ удобно применять для изменения формы уже существующего тела. Вершину можно передвинуть в новое положение, изменив

соседние ребра и грани. Прямое ребро можно заменить криволинейным, в результате чего изменятся связанные грани.

### **Объектно-ориентированное моделирование**

*Объектно-ориентированное моделирование (feature-based modeling)* – набор функций позволяющих конструктору создавать объемные тела, используя привычные элементы форм (features).

Большинством систем объектно-ориентированного моделирования поддерживаются такие элементы, которые используются при изготовлении деталей:

- фаски,
- крепежные(резьбовые, центровые) отверстия,
- шпоночные пазы,
- выемки,
- шлицы,
- зубчатые профили и т. д.

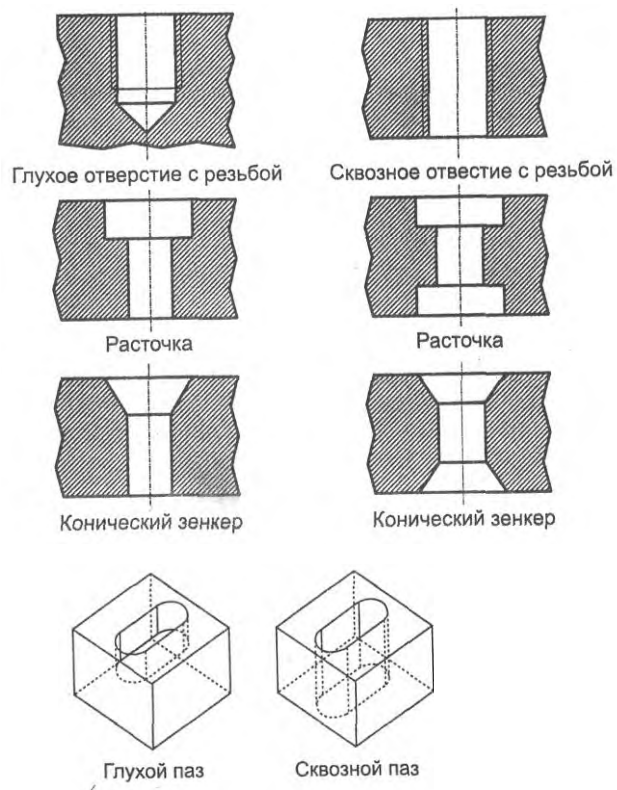
Такие элементы называются производственными, поскольку каждый из них может быть получен в результате конкретного процесса производства. Например, отверстие создается сверлением, зенкованием, нарезанием резьбы метчиком, а пазы, шлицы — фрезерованием.

Конструктор может давать команды типа «сделать отверстие под крепеж со следующими параметрами в таком-то месте» или «сделать фаску такого-то размера в таком-то месте», и получившаяся фигура будет содержать сведения о наличии в конкретном месте отверстия (или фаски) конкретного размера. При этом конструктор не задает эскизов и не выбирает операций традиционных для твердотельного моделирования.

На основании сведений о наличии, размере и расположении производственных элементов, задаваемых при использовании команд объектно-ориентированного моделирования облегчит планирование технологического процесса обработки моделируемой детали на станке с ЧПУ.

На рис.8 изображены шпоночный паз, и некоторые виды отверстий под крепеж - производственные элементы, часто используемые в моделировании машиностроительных деталей.





**Рисунок 8 Производственные элементы**

## Лекция 6

**Тема:** Системы геометрического моделирования CAD. Параметрические свойства.

**Цель:** Использование параметрических свойств моделей

### План:

Введение

1. Структуры данных для хранения описаний объемных тел
2. Структура данных B-Rep
3. Основные понятия параметрического моделирования
4. Область применения параметрических свойств модели.
5. Философия моделирования детали (Pro/ENGINEER)

### Структура данных для хранения описаний объемных тел

При создании объемного тела с применением функций объемного моделирования, рассмотренных ранее, в компьютере сохраняется его математическое описание.

Для хранения информации, отражающей однозначное математическое описание объемных тел, создаются структуры данных (форматы графических файлов). В структурах данных хранятся такие элементы как оболочки, грани, кольца и вершины, создаваемые с помощью операторов Эйлера.

Структуры данных, используемые для описания объемных тел, обычно делятся на три типа в зависимости от того, какие тела ими описываются:

- дерево CSG (GSG tree) - структура представляет собой дерево, описывающее историю применения булевских операций к примитивам. Журнал операций называется конструктивным представлением объемной геометрии (Constructive Solid Geometry — CSG representation).
- структурой B-rep (B-rep data structure) - структура содержащая сведения о границах объема (вершинах, ребрах, гранях и их соединении друг с другом). Это представление называется граничным представлением (boundary representation — B-rep).
- воксельное представление - структура представляет объем в виде комбинации элементарных объемов (например, кубов). Трехмерный аналог растрового представления плоской фигуры.

### Структура данных B-Rep

Границы объемных тел состоят из элементарных геометрических объектов: вершин, ребер и граней.

*Структуре данных B-Rep* – структура хранения всех вершин, ребер и граней модели вместе со сведениями о том, как они соединены друг с другом.

Одна из простейших структур данных, если не самая простая, приведена в табл. 1. Структура данных представляет объемное тело, изображенное на рис. 1.

В табл. 1 - таблице граней хранится список ограничивающих ребер для каждой грани.

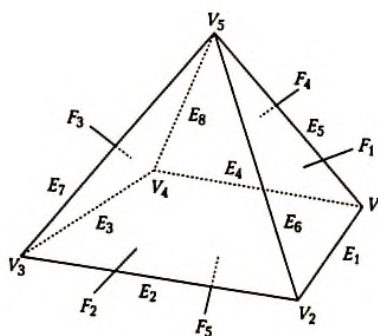


Рисунок 1

Таблица граней	Таблица ребер	Таблица вершин			
		Ребро	Вершины	Вершина	Координаты
F <sub>1</sub>	E <sub>1</sub> , E <sub>5</sub> , E <sub>6</sub>	E <sub>1</sub>	V <sub>1</sub> , V <sub>2</sub>	V <sub>1</sub>	x <sub>1</sub> , y <sub>1</sub> , z <sub>1</sub>
F <sub>2</sub>	E <sub>2</sub> , E <sub>6</sub> , E <sub>7</sub>	E <sub>2</sub>	V <sub>2</sub> , V <sub>3</sub>	V <sub>2</sub>	x <sub>2</sub> , y <sub>2</sub> , z <sub>2</sub>
F <sub>3</sub>	E <sub>3</sub> , E <sub>7</sub> , E <sub>8</sub>	E <sub>3</sub>	V <sub>3</sub> , V <sub>4</sub>	V <sub>3</sub>	x <sub>3</sub> , y <sub>3</sub> , z <sub>3</sub>
F <sub>4</sub>	E <sub>4</sub> , E <sub>8</sub> , E <sub>5</sub>	E <sub>4</sub>	V <sub>4</sub> , V <sub>1</sub>	V <sub>4</sub>	x <sub>4</sub> , y <sub>4</sub> , z <sub>4</sub>
F <sub>5</sub>	E <sub>1</sub> , E <sub>2</sub> , E <sub>3</sub> , E <sub>4</sub>	E <sub>5</sub>	V <sub>1</sub> , V <sub>5</sub>	V <sub>5</sub>	x <sub>5</sub> , y <sub>5</sub> , z <sub>5</sub>
		E <sub>6</sub>	V <sub>2</sub> , V <sub>5</sub>	V <sub>6</sub>	x <sub>6</sub> , y <sub>6</sub> , z <sub>6</sub>
		E <sub>7</sub>	V <sub>3</sub> , V <sub>5</sub>		
		E <sub>8</sub>	V <sub>4</sub> , V <sub>5</sub>		

Таблица 1.

Описание структуры:

- Последовательность ребер для каждой грани дается обходом против часовой стрелки, если смотреть на тело снаружи.

Ребра хранятся согласованно, вместе с каждой гранью сохраняется информация о том, с какой стороны от нее находится внутренний объем тела. (Таким образом, можно определить, где расположена конкретная точка: снаружи или внутри тела.)

- Вершины, ребра и грани нумеруются системой геометрического моделирования в произвольном порядке в момент сохранения сведений.

- В каждой строке таблицы ребер хранятся вершины, находящиеся на концах соответствующего ребра, а в строках таблицы вершин хранятся координаты всех вершин.
- координаты вершин обычно определяются в модельной системе координат, связанной с данным телом.
- Если убрать из таблицы грани, эту структуру данных можно использовать для хранения форм, созданных в системах каркасного моделирования. Структура данных для каркасной модели может использоваться в качестве базовой для систем автоматизированной разработки чертежей, если допустить указание двумерных координат для точек.

### **Основные понятия параметрическое моделирование.**

Задание и изменение пользователем геометрических ограничений или размеров каждый раз даст другой эскиз и другое объемное тело. Такой подход в моделировании называется параметрическим моделированием, поскольку изменение параметров позволяет получить разные объекты.

*Параметрическое моделирование (parametric modeling)* – определение формы модели путем задания геометрических ограничений и размеров. Построение модели является результатом решения системы уравнений, выражающих геометрические ограничения, соотношения между размерами.

Геометрические и размерные зависимости задаются для того, чтобы, модель при изменении своих параметров обновлялась ожидаемым образом. Параметрические зависимости позволяют ограничить возможные изменения размеров и формы модели при ее редактировании.

К параметрическим зависимостям относятся:

- Геометрические зависимости, контролирующие форму и связи между линиями и дугами эскиза;
- Размерные зависимости, контролирующие размеры эскизной геометрии.

**К геометрическим зависимостям** относятся следующие типы связей между геометрическими элементами:

- линия горизонтальна,
- линия вертикальна,
- вершины лежат на одной линии (коллинеарность),

- элементы перпендикулярны друг другу,
- элементы параллельны друг другу,
- элементы касательны (тангенциальные) друг другу,
- вершины/точки совпадают; либо вершина/точка лежит на элементе,
- вершина лежит на середине элемента,
- вершины симметричны относительно осевой линии,
- оси дуг и/или окружностей совпадают (соосность, концентричность),
- Совпадение дуг, окружностей (корадиальность),
- длины/радиусы элементов одинаковы.

Панель инструментов «Эскиз», содержит опцию «Добавить взаимосвязи», с помощью которой можно наложить ограничения в поведении, как отдельных параметрических элементов, так и их групп (по-другому, наложить геометрические связи).

Во время создания любого элемента эскиза система проверяет, насколько он «ей понятен». При этом система делает ряд допущений (аппроксимаций). Вот, например, некоторые приближения, которые может делать система.

- приблизительно равные диаметры/радиусы равными;
- приблизительно вертикальные и горизонтальные линии;
- приблизительно перпендикулярные и параллельные линии таковыми;
- приблизительно касательные - касательными;
- дуги приблизительно в 90, 180 и 270 градусов точно такими;
- приблизительно коллинеарные объекты коллинеарными;
- точки, приблизительно лежащие на одной горизонтали или вертикали, таковыми.

В тот момент, когда создаваемый элемент попадает в зону допущения, система высвечивает специальный знак геометрической связи - закрепление, а элемент как бы «залипает» в том или ином положении. Если пользователь желает исключить данное приближение, то он должен вывести создаваемый элемент из зоны допущения (до исчезновения значка закрепления), т.е. сделать различие между элементами ощутимым. Бывает необходимо создать элемент приблизительно равный, но все-таки отличный. В этом случае пользователь может активизировать или останавливать действие закреплений, используя настройки эскиза в главном меню.

Вообще говоря, все геометрические объекты современных CAD считаются параметрическими, то есть для них характерны параметрические свойства:

- Каждый элемент имеет имя, параметры и их значения. Например, Вытянуть1 – имя бобышки, которое назначается системой автоматически при ее создании. Пользователь сам может переименовать этот геометрический элемент, назвав его, например, Основание. Элементы эскиза тоже имеют свои имена, например, Линия1, Линия2, Дуга3.
- Для параметрических элементов можно задать параметрические зависимости (размер, геометрические взаимосвязи).

Для того, чтобы задать геометрическую зависимость, достаточно задать имена элементов и выбрать характер зависимости между ними. В простейшем случае это может быть равенство двух элементов, возможна зависимость от другого размера по сложному закону или от нескольких элементов.

### Размеры

Каждый размер на эскизе имеет свое численное значение и имя - символическое обозначение (например, размер 32 имеет имя D3@Эскиз1 на рис.2).

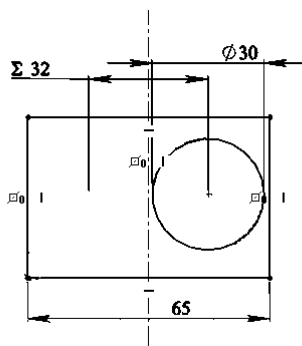


Рисунок 2

Свойства параметрического эскиза:

- Эскиз образмеривается автоматически (во время перетаскивания геометрических элементов размеры автоматически изменяются, приходя в соответствие с текущим состоянием эскиза).
- Конфигурацию эскиза можно изменять, меняя непосредственно размеры (если они не противоречат ранее добавленным геометрическим взаимосвязям).
- Конфигурацией эскиза можно управлять с помощью уравнений (соотношений между размерами). Например, "D3@Эскиз1" = "D1@Эскиз1"+2.
- Конфигурацией эскиза можно управлять с помощью таблиц параметров.

Символьные обозначения размеров используются в качестве переменных в математических выражениях, устанавливая тем самым соотношения (уравнения) между ними. Каждый управляющий размер является переменной с определенным именем (например, D3@Эскиз1). Вместо того, чтобы указывать конкретное значение размера числом, можно задать имя уже существующего размера либо выражение, в котором используется такое имя (или имена).

С моделью или чертежом могут быть также связаны переменные, которые не являются значениями размеров, но косвенно влияют на них (крутящий момент и материал определяют размеры сечения).

### **Область применения параметрических свойств модели.**

*Параметрический элемент* – это геометрический элемент, фактически являющийся библиотекой деталей или форм, создаваемый для того, чтобы не тратить время на одинаковые построения. Элемент можно добавлять при построении моделей и редактировать, при этом исходный файл параметрического элемента остается неизменным. Из параметрических элементов состоят библиотеки унифицированных деталей/узлов/изделий.

В процессе получения экземпляров унифицированных деталей используются свойства параметризации. При задании основных параметров (управляющих переменных) модели, отличающих данный экземпляр, производится расчет, в результате которого произойдет изменение других параметров (управляемых переменных) и как следствие изменится геометрия.

Принципиальное значение параметризация имеет:

- при проработке различных вариантов одного и того же изделия на начальных стадиях проектирования;
- когда необходимо часто вносить изменения в параметры изделия (связана с необходимостью создания параметрических типоразмерных и модельных рядов изделий);
- при создании библиотеки унифицированных деталей/узлов/изделий.

Одной из первоочередных для конструктора является задача создания концептуального дизайна будущего продукта и первоначальная увязка элементов конструкции. Чтобы облегчить работу конструктора используется параметризация — инструмент, позволяющий за короткое "время" «проиграть» различные

конструктивные схемы и избежать принципиальных ошибок - изменить один или несколько параметров и посмотреть, как будет вести себя при этом изделие.

Для управления формой элементов модели создается схема параметризации.

Задание параметризации модели детали включает:

- задание относительного расположения вспомогательной геометрии (плоскостей, осей, вспомогательных координатных плоскостей) если таковая необходима,
- задание геометрических зависимостей для элементов каждого эскиза,
- простановку размеров в соответствии с размерной схемой (с учетом последовательности простановки, выбора замыкающего размера),
- задание зависимостей между размерами с помощью формул, таблиц.

*Размерная схема* - совокупность размеров, определяющих форму, размеры изделия и расположение размеров на чертеже.

Требования при разработке размерной схемы:

- Размерная определенность – все элементы изделия должны быть закоординированы и иметь или собственный размер или возможность рассчитать собственный размер;
- Размерная избыточность. Допускается избыточность размеров (размеры для справок);
- Группы взаимосвязанных размеров проставляются либо цепочкой либо от одной поверхности (базы).
- Размерная избыточность в эскизах не допускается. Система преобразует избыточный размер в управляемый (расчетный) размер.
- Порядок простановки размеров в эскизе важен и размеры проставляются от малых к большим, так как при корректировке модель перестраивается.

## **Философия моделирования детали (в среде Pro/ENGINEER)**

Здесь приводится практически без авторских изменений.

Перед началом моделирования детали в среде необходимо заглянуть на несколько шагов вперед:



- сделать анализ функционального назначения детали и анализ стратегии построения ее виртуальной модели.
- предусмотреть возможные изменения в конструкции при подготовке производства и в процессе проведения последующих модернизаций изделия.
- произвести анализ - делать ли модель "с чистого листа" или переделать из уже существующей модели.

До начала построения геометрии детали необходимо ответить на следующие вопросы:

- Какие размеры являются критическими для геометрии, т.е. могут кардинально изменить ее пропорции и внешний вид?
- Какие размеры могут/будут меняться в будущем?
- Какие размеры важны с точки зрения обработки детали?
- Какие размеры должны управлять другими через соотношения?
- Будет ли деталь принадлежать групповому чертежу (т.е. входить в состав конфигураций (Таблиц семейства))
- Как (и с какими конструктивными элементами) взаимодействует деталь в сборке(ах)?

Шаги построения модели:

1. Определение базовой операции;
2. Создание эскиза базовой операции;
3. Определение желаемой размерной схемы для базовой операции (Размерную схему желательно выбирать такую же, как на чертеже);
4. Сопоставление характеристик операции с операциями в других моделях;
5. Добавление конструктивных операций при соблюдении размерной схемы;
6. Добавление соотношений между размерами для повышения устойчивости модели при модификациях;
7. Назначение баз обработки, полей допусков размерам, отклонение формы и т.д.;
8. Добавление дополнительных атрибутов(физических свойств) модели (плотности, параметров и т.д.).

Советы построения модели:

- начинайте моделирование с создания базовых опорных плоскостей, которые обязательно используйте в качестве плоскостей симметрии;

- при моделировании старайтесь сложную геометрию модели формировать последовательностью несложных операций (это облегчит возможные исправления и откаты);
- минимизируйте количество «рисованных» операций;
- моделируйте деталь таким образом, чтобы опорная плоскость была бы параллельна плоскости чертежа на главном виде;
- создавайте такие размерные схемы в операции, которые бы максимально соответствовали размерной схеме будущего чертежа.

### ***Литература***

1. Кунву Ли. Основы САПР (CAD/CAM/CAE). Спб.: Питер, 2004.– 560 с.: ил.
2. Фоли Дж., вэн Дэм А. Основы интерактивной машинной графики: В 2-ух книгах. Книга 2. Пер с англ. – М.: Мир, 1985. – 386 с.: ил.

**САМАРКИНА Елена Ивановна**

**АВТОМАТИЗАЦИЯ КОНСТРУКТОРСКО-  
ГРАФИЧЕСКИХ РАБОТ В МАШИНОСТРОЕНИИ**

Конспект лекций  
для студентов машиностроительных специальностей

Редактор Е.И. Самаркина  
Компьютерная верстка Е.И. Самаркина

---

Подписано в печать 17.12.2007 г. Формат 60x90/16.  
Гарнитура Таhоmа. Усл. п.л. 4,2  
Тираж 120 экз. Заказ № 682

Адрес издательства:  
Россия, 180000, Псков, ул. Л.Толстого 4.  
Издательство ППИ