

ФУ-39

ФГБОУ ВПО  
«Воронежский государственный технический университет»

Кафедра графики, конструирования и информационных технологий в промышленном дизайне

**9-2014**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

по изучению раздела «Компьютерная графика»  
дисциплины «Инженерная и компьютерная графика»  
для студентов всех направлений подготовки бакалавров  
очной формы обучения



Воронеж 2014

Составители: канд. техн. наук Е.А. Балаганская,  
ст. преп. Е.К. Лахина

УДК 004.92

Методические указания по изучению раздела «Компьютерная графика» дисциплины «Инженерная и компьютерная графика» для студентов всех направлений подготовки бакалавров очной формы обучения / ФГБОУ ВПО «Воронежский государственный технический университет»; сост. Е.А. Балаганская, Е.К. Лахина. Воронеж, 2014. 59 с.

В методических указаниях представлены и раскрыты темы по разделу «Компьютерная графика», отвечающие современному состоянию и перспективам развития проектно-конструкторских работ в системах автоматизированного проектирования (САПР), конструирования (АСК) и технологии (АСТПП) изготовления сложных технических объектов в соответствии с требованиями ФГОС-3.

Предназначены для студентов первого и второго курсов.

Ил. 7. Табл. 1. Библиогр.: 4 назв.

Рецензент д-р техн. наук, проф. Ю.А. Цеханов

Ответственный за выпуск зав. кафедрой д-р техн. наук,  
проф. А.В. Кузовкин

Печатается по решению редакционно-издательского совета Воронежского государственного технического университета

МС-132 9025-07.15. Караваев

© ФГБОУ ВПО «Воронежский  
государственный технический  
университет», 2014

## ВВЕДЕНИЕ

Современный уровень программных и технических средств электронной вычислительной техники позволяет перейти от традиционных, ручных методов конструирования к *новым информационным технологиям* с использованием ЭВМ, создавать *системы автоматизации разработки и выполнения конструкторской документации (АКД)*, удовлетворяющие стандартам ЕСКД как по качеству исполнения документов, так и по соблюдению требований стандартов.

Средства для реализации АКД предоставляет *компьютерная графика (КГ)* – область информатики, предназначенная для создания, хранения и обработки моделей *геометрических объектов (ГО)* и их *изображений* с помощью ЭВМ. Наиболее эффективными для автоматизации конструкторской деятельности являются постоянно развивающиеся *интерактивные средства* компьютерной графики, обеспечивающие процесс конструирования в режиме диалога «человек - ЭВМ».



## 1. ПОНЯТИЕ О КОМПЬЮТЕРНОЙ ГРАФИКЕ

Компьютерная графика является подсистемой САПР.

Под компьютерной (машинной) графикой понимается совокупность методов и приемов преобразования при помощи ЭВМ данных в графическое представление или графического представления в данные. Под графическим представлением понимается изображение либо комплекс изображений (чертеж).

Компьютерная графика предназначена для создания, хранения и обработки математических и графических моделей объектов и формирования их изображений с помощью ЭВМ.

К задачам компьютерной графики традиционно относят процесс *создания и обработки изображений*.

*Областью применения* компьютерной графики является автоматизация проектно-конструкторских работ.

Различают три вида компьютерной графики. Это *растровая* графика, *векторная* графика и *фрактальная* графика. Они отличаются принципами формирования изображения при отображении на экране монитора или при печати на бумаге.

*Растровую* графику применяют при разработке электронных (мультимедийных) и полиграфических изданий. В последнее время для ввода растровых изображений в компьютер нашли широкое применение цифровые фото- и видеокамеры. Соответственно, большинство графических редакторов, предназначенных для работы с растровыми иллюстрациями, ориентированы не столько на создание изображений, сколько на их обработку.

*Векторная* графика предназначена для создания иллюстраций и в меньшей степени для их обработки. Такие средства широко используют для выпуска чертежно-конструкторской документации, в рекламных агентствах, дизайнерских бюро, редакциях и издательствах.

*Фрактальная* графика предназначена для автоматической генерации изображений путем математических расчетов. Создание фрактальной художественной композиции состоит не в рисовании или оформлении, а в программировании. Фрактальную графику редко применяют для создания печатных или электронных документов, но ее часто используют в развлекательных программах.

### 1.1. Векторная графика

**Векторная графика** - описывает изображение с помощью математических формул.

Структуры данных *векторной (объектной)* модели изображения соответствуют разомкнутым или замкнутым линиям (*геометрическим примитивам*), из которых составляется изображение. Каждая такая структура соответствует независимому объекту. У каждого объекта имеется имя, тип и набор параметров, задающих его геометрические характеристики (*атрибуты*).

В векторной графике изображения строятся с помощью математических описаний объектов. В основе векторной графики лежат математические представления о свойствах геометрических фигур. Ключевым моментом векторной графики является то, что она использует комбинацию компьютерных команд и математических формул для описания объектов.

Векторную графику называют *объектно-ориентированной* или *чертежной* графикой.

Минимальными элементами векторной модели изображений являются *геометрические примитивы*, предусмотренные *графическим редактором*, либо их комбинирование.

Объекты векторной графики хранятся в памяти в виде набора параметров, но не надо забывать и о том, что на экран все изображения все равно выводятся в виде точек (просто



потому, что экран так устроен). Перед выводом на экран каждого объекта программа производит вычисления координат экранных точек в изображении объекта, поэтому векторную графику иногда называют вычисляемой графикой. Аналогичные вычисления производятся и при выводе объектов на принтер.

Файлы векторной графики способны содержать растровые изображения в качестве одного из типов объектов. Это возможно, поскольку растровый рисунок просто набор инструкций для компьютера.

#### *Достоинства векторной графики:*

- Она использует все преимущества разрешающей способности любого устройства вывода, что позволяет изменять размеры векторного рисунка без потерь его качества. Векторные команды просто сообщают устройству вывода, что необходимо нарисовать объект заданного размера, используя столько точек, сколько возможно. Растровый же формат, в отличие от векторного, точно определяет, сколько необходимо создать пикселей, и это количество не изменяется вместе с изменением разрешающей способности устройств вывода. Это приводит при увеличении разрешающей способности принтера либо к уменьшению размера изображения, либо для каждого пиксела используется большее количество точек. Сравнение векторного и растрового изображения окружности показано на рис. 1.1.

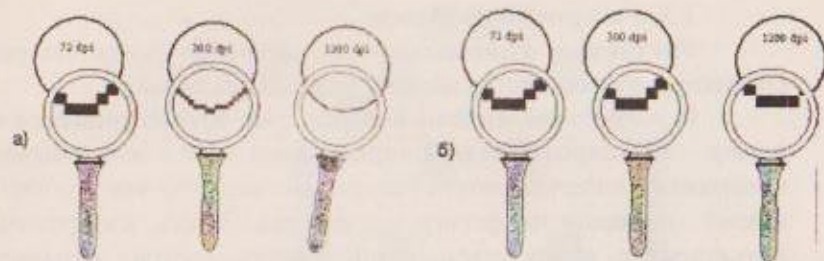


Рис. 1.1. Печать окружности на принтерах с различным разрешением: а) векторная, б) растровая.

- Векторная графика позволяет редактировать отдельные части рисунка, не оказывая влияния на остальные (в растровых изображениях пришлось бы редактировать каждый пиксел).

- Векторные изображения, не содержащие растровых объектов, занимают в памяти компьютера относительно небольшое место (в 10 - 1000 раз меньше, чем его растровый аналог).

#### *Недостатки векторной графики:*

- Рисунки часто выглядят достаточно искусственно, так как основным компонентом векторного рисунка является прямая линия, а она в природе встречается достаточно редко. Поэтому до недавнего времени векторная графика использовалась только для технических иллюстраций, чертежей.

- Возможны проблемы при печати, как правило, сложных рисунков на отдельных типах принтеров из-за того что не все команды могут ими правильно интерпретироваться.



## 1.2. Растровая графика

**Растровая графика** – это прямоугольная матрица, состоящая из множества мелких неделимых точек.

В *растровой* модели изображение рассматривается как *растр* – регулярная сетка, покрывающая собой всю плоскость изображения. Регулярность растра означает, что все его ячейки имеют одинаковую форму и размеры. Часть изображения, размещенная в пределах одной ячейки растра, называется *пикселом*. В простейшем варианте пиксельная (растровая) модель представляет собой последовательность описаний всех пикселов изображения – *дескрипторов*.

Растр (по-английски *bitmap* массив битов) это просто совокупность битов, расположенных на сетчатом поле. Единицы измерения информации: бит (минимальная единица - 1 бит), байт (8 бит), килобайт (1024 байт), мегабайт (1024 килобайт), гигабайт (1024 мегабайт). *Бит* единица информации в компьютере, обозначающая ячейку памяти, которая может находиться во включенном (1) или выключенном (0) состоянии. Эти состояния можно отождествить с черным и белым цветом, т.е. соединив несколько битов, можно создать изображение из черных и белых точек. Из таких элементов (кирпичиков) как витраж собирается растровое изображение, которое напоминает лист клетчатой бумаги, на котором каждая клеточка закрашена черным или белым цветом, формируя при этом рисунок (рис. 1.2).

Основным термином растровой графики является *пиксел* - отдельный элемент растрового изображения.

Для обозначения количества пикселов в матрице рисунков по горизонтали и по вертикали используется коэффициент прямоугольности изображения. Этот коэффициент часто называют размером изображения и записывают в виде 800 600 (800 пиксел по горизонтали и 600 строк по вертикали). Произведение этих двух чисел дает общее количество пиксел изображения.

Первые десять				■	■	■				
Вторые десять			■					■		
И т.д.		■							■	
	■			■		■				■
	■									■
	■		■					■		■
		■		■	■	■			■	
			■					■		
				■	■	■				

Рис. 1.2. Хранение и воссоздание растрового изображения

Исторически термин «пиксел» возник как транслитерация аббревиатуры, составленной из начальных слогов двух английских слов - *picture* (картина) и *element* (элемент) (*Pixel - Picture Element* - элемент изображения).

Термином «пиксел» обозначают несколько разных понятий:

- отдельную точку на экране компьютера;
- отдельную точку на изображении, напечатанном на лазерном принтере;
- отдельный элемент растрового изображения.

Поэтому при использовании данного термина рекомендуется уточнять область его применения.

Пиксел, состоящий из двух битов, может давать четыре возможных цвета или градации серого, соответственно четыре бита информации дадут 16 цветов, 8 бит - 256 цветов, 24 бита обеспечат более 16 миллионов доступных цветов. Следует помнить, что пиксел сам по себе не обладает никаким



размером. Он всего лишь область памяти компьютера, хранящая информацию о цвете. Поэтому коэффициент прямоугольности изображения не соответствует никакой размерности. Размеры изображения хранятся отдельно, пиксели запоминаются один за другим, обычно как один большой блок данных. Таким образом, компьютер не сохраняет отдельные позиции для каждого пиксела, он всего лишь воссоздает сетку по размерам, заданным коэффициентом прямоугольности.

Растровые изображения содержат большое количество пикселов, каждый из которых занимает определенную часть памяти. Например, отсканированное с фотографии цветное изображение может занимать десятки и сотни Мбайт.

Наибольшее влияние на количество памяти, занимаемой растровым изображением, оказывают три фактора:

*размер изображения* (коэффициент прямоугольности);

*битовая глубина изображения;*

*формат файла*, используемый для хранения изображения.

*Достоинства растровой графики:*

- Растровые изображения выглядят вполне реалистично. Это связано со свойствами человеческого глаза: он приспособлен для восприятия реального мира как огромного набора дискретных элементов, образующих предметы.

- Легко управлять выводом изображения на устройства представляющие изображения в виде совокупности точек: принтеры, фотонаборные автоматы.

*Недостатки растровой графики:*

- Большой объем памяти, требуемый для хранения изображения хорошего качества.

- Трудности редактирования изображений. Так как сами изображения занимают много памяти компьютера, то, очевидно, и для их редактирования потребуется так же много памяти.

### 1.3. Понятие о фрактальной графике

**Фрактальная** графика, как и векторная - вычисляемая, но отличается от нее тем, что никакие объекты в памяти компьютера не хранятся. Изображение строится по уравнению (или по системе уравнений), поэтому ничего, кроме формулы, хранить не надо. Изменив коэффициенты в уравнении, можно получить совершенно другую картину.

Простейшим фрактальным объектом является фрактальный треугольник. Постройте обычный равносторонний треугольник со стороной  $a$ . Разделите каждую из его сторон на три отрезка. На среднем отрезке стороны постройте равносторонний треугольник со стороной, равной  $a/3$  стороны исходного треугольника, а на других отрезках постройте равносторонние треугольники со стороной, равной  $a/9$ . С полученными треугольниками повторите те же операции (рис. 1.3). Вскоре вы увидите, что треугольники последующих поколений наследуют свойства своих родительских структур. Так рождается фрактальная фигура.

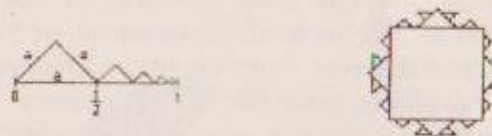


Рис. 1.3. Фрактальная графика

Процесс наследования можно продолжать до бесконечности. Фрактальными свойствами обладают многие объекты живой и неживой природы. Обычная снежинка, многократно увеличенная, оказывается фрактальным объектом. Фрактальные алгоритмы лежат в основе роста кристаллов и растений. Способность фрактальной графики моделировать образы живой природы вычислительным путем часто используют для автоматической генерации необычных иллюстраций.



#### 1.4. Цвет

Понятие цвета и света в компьютерной графике являются основополагающими.

*Свет* можно рассматривать как поток электромагнитных волн (в этом случае *цвет* определяется длиной волны).

Из курса физики известно, что белый свет в действительности состоит из всех цветов радуги. Видимый свет состоит из спектрального распределения электромагнитной энергии с длинами волн в диапазоне от 400 до 700 Нм. Цвет излучений, длины волн которых расположены в диапазоне видимого света в определенных интервалах вокруг длины какого-либо монохроматического излучения, называются спектральными цветами. Пропуская белый свет через призму, получим видимый спектр света красный (red), оранжевый (orange), желтый (yellow), зеленый (green), голубой (blue), синий (indigo) и фиолетовый (violet). Излучения с длинами волн от 380 до 470 Нм имеют фиолетовый и синий цвет, от 470 до 500 Нм, сине-зеленый, от 500 до 560 Нм зеленый, от 560 до 590 нм желто-оранжевый, от 590 до 760 Нм красный. Каждый кусочек этого видимого спектра имеет свое уникальное значение, которое и называется *цветом*. В видимом спектре содержатся миллионы различных цветов, и разница между двумя соседними практически не заметна.

*Излучаемый свет* это свет, выходящий из активного источника (солнца, лампочки, экрана монитора).

*Отраженный свет* это свет "отскочивший" от поверхности объекта. Именно его мы видим, когда смотрим на некоторый предмет, не излучающий своего собственного света.

Излучаемый свет может содержать все цвета (белый свет), любую их комбинацию или только один цвет. Излучаемый цвет, идущий непосредственно от источника к глазу, сохраняет в себе все цвета, из которых он был создан. Некоторые волны излученного света (которые воспринимаются нами как цвета) поглощаются объектом, на

который они попадают, поэтому глазом воспринимаются только не поглощенные, отраженные волны. Таким образом, некоторые предметы мы видим потому, что они излучают свет, а другие потому, что они его отражают.

Фундаментальные различия механизмов образования цвета излученным и отраженным светом приводят к необходимости применения различных цветовых моделей.

В компьютерной графике существуют два противоположных метода его описания: системы *аддитивных* и *субтрактивных* цветов.

*Аддитивный* цвет получается путём добавления (addition) лучей света разных цветов к черному. В этой системе отсутствие всех цветов дает черный цвет, а присутствие всех цветов белый. Система аддитивных цветов работает с *излучаемым светом*, например от монитора компьютера.

В аддитивной модели RGB (рис. 1.4) используются три основных цвета: *красный* (Red), *зеленый* (Green) и *синий* (Blue). Смешивая их в разных пропорциях можно получить

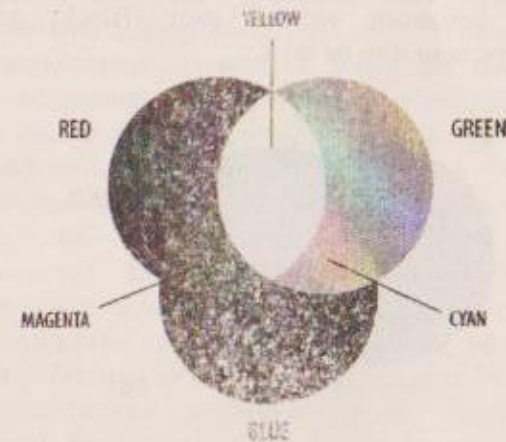


Рис. 1.4



любой цвет. Известная задолго до появления компьютеров, она оказалась наиболее к ним приспособленной, так как монитор компьютера создает цвет излучением света, а экран его состоит из мельчайших точек красного, зеленого и синего цвета, интенсивностью свечения которых можно управлять. Однако, что хорошо для монитора не всегда хорошо при печати.

В системе *субтрактивных* цветов, основанной на вычитании, (subtract) происходит обратный процесс: цвет получается, вычитая другие цвета из общего луча света. В такой системе белый цвет соответствует отсутствию всех цветов, тогда как их наличие дает черный цвет. Система субтрактивных цветов работает с *отраженным светом*, например от листа бумаги.

Субтрактивные модели CMY и CMYK (рис.1.5) – основные модели для всех случаев, когда используется *отраженный свет*. Разница между этими моделями в том, что в модели CMYK к базовым цветам модели CMY: голубому (Cyan), пурпурному (Magenta) и желтому (Yellow), – дополнительно добавлен черный цвет (Black) для более полной цветопередачи при печати.

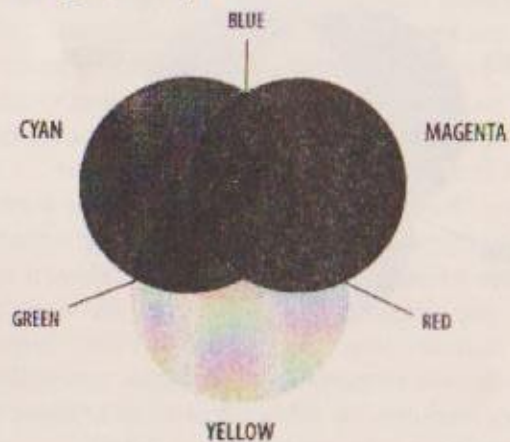


Рис. 1.5

Система CMYK была широко известна задолго до использования компьютеров для создания графических изображений, так как она широко применялась в цветной печати.

Преобразование рисунков из системы RGB в систему CMYK сталкивается с рядом проблем. Основная сложность заключается в том, что в разных системах цвета могут меняться. Напомним, что система RGB работает с излучаемым светом, а система CMYK с отраженным; у них различна сама природа получения цвета, и то, что видим на экране монитора, никогда нельзя точно повторить на печати. Процесс преобразования усложняется еще и тем, что необходимо корректировать несовершенство типографских красок добавлением черного цвета.

Системы цветов RGB и CMY, CMYK базируются на ограничениях, накладываемых аппаратным обеспечением (монитором, типографскими красками).

*Интуитивная* модель HSB – сокращение от английских слов Hue (*оттенок или цвет*), Saturation (*насыщенность*) и Brightness (*яркость*). HSB-математическая модель, основанная на модели RGB, но имеющая другую систему координат, представляет собой цветовой круг. Цветовая модель очень удобна для подбора оттенков и цветов. Вариантами этой модели является HLS (Hue – *тон*, Lightness – *освещенность*, Saturation – *насыщенность*) и HSV (Hue, Saturation, Value – *значение*).

Наиболее важный *атрибут* цвета *цветовой тон* ассоциируется в человеческом сознании с обусловленностью окраски предмета определенным типом пигмента, краски, красителя. *Насыщенность* характеризует степень, силу, уровень выражения цветового тона. Этот *атрибут* в человеческом сознании связан с количеством (концентрацией) пигмента, краски, красителя. *Яркость* (или *освещенность*) цвета показывает величину черного оттенка, добавленного к цвету, что делает его более темным.



Система HSB больше соответствует природе цвета, но для работы на мониторах и для печати ее надо преобразовывать в другие системы RGB и CMYK, соответственно.

Модель HSB имеет широкий цветовой охват, он меньше чем у модели Lab, но больше цветового охвата модели CMYK.

*Цветовая трехканальная модель Lab* разрабатывалась специально как аппаратно-независимая, то есть определяющая цвета без учета технологии цветовоспроизведения (на мониторе, на принтере, на печатном станке).

Все описанные ранее системы (RGB, CMY, CMYK, HSB, HLS) имеют дело со всем спектром цветов: миллионами возможных оттенков. Однако большинству пользователей компьютеров часто бывает достаточно 256 или даже 16 доступных цветов.

*Индексированные палитры цветов* это наборы цветов, из которых можно выбрать необходимый цвет. Преимуществом ограниченных палитр является то, что они занимают значительно меньше памяти, чем полные системы. Компьютер создает палитру цветов и присваивает каждому цвету номер. Затем при сохранении цвета отдельного пиксела или объекта просто запоминается номер, который имеет этот цвет в палитре.

### 1.5. Графические языки

Фактически любая графическая операция сводится к работе с отдельными пикселями: поставить точку заданного цвета или узнать цвет заданной точки. Однако большинство библиотек поддерживает работу и с более сложными объектами, поскольку работа на уровне отдельно взятых пикселей была бы очень затруднительной для программиста и к тому же неэффективной.

Среди подобных объектов (представляющих собой объединение пикселей) можно выделить следующие основные группы:

- линейные изображения (растровые образы линий);
- сплошные объекты (растровые образы двумерных областей);
- шрифты;
- изображения (прямоугольные матрицы пикселей).

Графические языки – формальные языки, предназначенные для описания графических изображений и алгоритмов их обработки на ЭВМ.

Графические языки относятся к классу проблемно-ориентированных языков.

Проблемно-ориентированные графические языки, которые используются в системах машинной графики, принято классифицировать по следующим признакам:

- оперативность;
- наличие средств для описания операций обработки;
- связь с универсальными алгоритмическими языками программирования;
- способ задания команд (операторов) языка;
- место в процессе обработки графических данных.

По *оперативности* языки делят на *диалоговые* (*оперативные*) и *пассивные*. *Диалоговые* обеспечивают работу в реальном масштабе времени путем обработки операторов языка в режиме интерпретации, что позволяет оперативно получать результат выполнения программы в графической форме. В диалоговых языках для задания операторов рядом с алфавитно-цифровыми данными используются и графические построения, которые выполняются на графическом терминале с помощью устройств ввода.

*Пассивные* языки разрешают задавать совокупность графических операций в виде некоторого символического описания с последующей компиляцией этих описаний и выполнением в режиме пакетной обработки.

По *наличию средств для описания операций обработки* выделяют *информационные* и *алгоритмические* языки. *Информационные* позволяют описывать только графические



данные. *Алгоритмические* предназначены для описания графических данных и операций над ними, включая вычислительные операции, операции управления вводом-выводом и хранения данных.

По связи с универсальными алгоритмическими языками различают *автономные* и *расширяющие* языки. *Автономный* язык имеет собственную грамматику, соответствующий транслятор и может применяться независимо от других языков программирования. *Расширяющие* языки строятся на основе грамматики другого языка и являются его графическим дополнением. Базой расширения чаще всего служат универсальные алгоритмические языки. Такой подход позволяет использовать все имеющиеся в базовом языке мощные средства обработки данных и упростить связь машинной графики с проектирующими компонентами системы, а также обеспечить в значительной мере независимость языка от типа ЭВМ.

По способу задания операторов языки выделяют *символические* (алфавитно-цифровые), *цифровые*, и *графо-символические* языки. Программа на *символическом* языке представляет последовательность текстовых строк фиксированного или произвольного формата. *Цифровые* языки представляют совокупность кодовых комбинаций, в которых числами задаются как коды графических команд, так и их параметры. *Графо-символические* языки, как правило, диалоговые и позволяют задавать графическую информацию в форме комбинации текстовых директив и графических построений.

По месту в процессе обработки графических данных различают *входные*, *внутренние* и *выходные* графические языки. *Входные* графические языки предназначены для описания и ввода в ЭВМ графических данных и задания действий над ними. Обычно входные графические языки включают некоторый базисный набор графических операторов и предусматривают возможность расширения их в

зависимости от специфики области и условий применения. Общими для многих входных графических языков являются подмножества команд, которые обеспечивают:

- построение графических примитивов;
- задание атрибутов графических примитивов;
- построение графических изображений произвольной конфигурации;
- построение изображения из ограниченного множества элементов, которые имеют типичную конфигурацию;
- преобразование изображения (аффинные и др. преобразования);
- документирование информации в графическом, текстовом виде или запись на машинные носители;
- прием и передачу информации;
- управление устройствами ввода.

Подмножества этих команд могут быть расширены или сокращены в зависимости от области и условий использования конкретного языка.

*Внутренние* языки предназначены для программной обработки данных, накопления и сохранения их в системе, обеспечения протоколов связи между различными компонентами системы. В современных системах внутренние языки используются также для записи данных в так называемые графические метафайлы, в которых изображения сохраняются в форматах, независимых от команд конкретных графических приборов.

*Выходные* языки предназначены для вывода данных с ЭВМ с целью графического отображения и документирования. Форматы и набор операторов исходных языков очень зависят от используемых устройств графического вывода.

Как правило, каждый компилятор имеет свою графическую библиотеку, обеспечивающую работу с основными группами графических объектов.

Графические команды имеются, например, в языках Basic, Borland C++, Pascal.



Язык Lisp находит применение при программировании для систем, построенных на основе AutoCAD.

Достаточно интересным является язык PostScript, предназначенный для описания страниц. PostScript создавался в качестве простого стандартного языка для описания вида текста, чертежей и простых изображений на печатаемой странице, что наложило определенный отпечаток на используемые в PostScript конструкции. Язык содержит около 250 операторов, что позволяет одни и те же действия запрограммировать самыми разными способами. Треть языка PostScript посвящена графике, остальное - это обычный процедурный язык программирования, который включает в себя элементы из многих других языков, но наиболее близок к языку Форт. Очень важно, что описание страниц на PostScript не зависит от устройства, на котором страница будет воспроизведена. Как правило, программы на PostScript генерируются приложениями, например текстовыми процессорами, программами для настольных издательских систем, в частности CorelDraw.

## 2. ТЕХНИЧЕСКИЕ СРЕДСТВА КОМПЬЮТЕРНОЙ ГРАФИКИ

*Основные компоненты компьютерной графической системы:*

- графическая станция,
- программное обеспечение автоматизированного выполнения графических изображений,
- устройства для ввода графической информации,
- устройства вывода графической информации.

### 2.1. Графическая станция

Графическая станция - IBM-совместимый компьютер<sup>1</sup> с мощным процессором, большим объемом оперативной памяти и быстрым жестким диском.

На рис. 2.1. представлен типичный персональный компьютер. Цифрами обозначены его основные составные части: 1 — монитор, 2 — материнская плата, 3 — центральный процессор, 4 — оперативная память, 5 — карты расширений, 6 — блок питания, 7 — оптический привод, 8 — жесткий диск, 9 — компьютерная мышь, 10 — клавиатура.

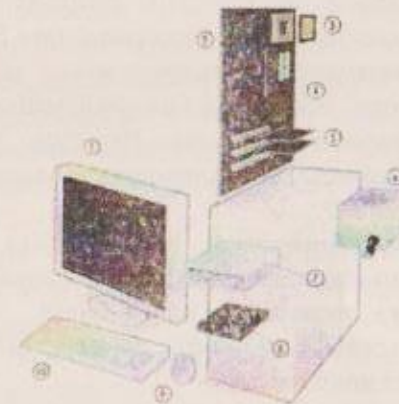


Рис. 2.1

Какие же требования предъявляются к графическим станциям и почему для решения многих задач обычного PC недостаточно? При работе с графикой возникает необходимость передавать большие объемы информации (с этим у PC все в порядке), осуществлять огромное количество целочисленных вычислений или вычисления с переменными

<sup>1</sup> IBM PC — первый массовый персональный компьютер производства фирмы IBM выпущен в 1981 году.

Персональный компьютер, ПК (*personal computer, PC*) — компьютер, предназначенный для эксплуатации одним пользователем, то есть для личного использования.



одинарной точности. Эти вычисления являются специфическими для работы с графикой, особенно трехмерной. У PC в этой области все обстоит значительно хуже, что привело к созданию сначала MMX-технологии, а затем и SSE-технологии.

*MMX — Multimedia Extensions.* Коммерческое название дополнительного набора инструкций, выполняющих характерные для процессов кодирования/декодирования потоковых аудио/видео данных действия за одну машинную инструкцию. Впервые появился в процессорах Pentium MMX.

*SSE* — набор инструкций, разработанный Intel, и впервые представленный в процессорах серии Pentium III.

Все технологии направлены на увеличение производительности обработки целочисленной информации или чисел одинарной точности. Их суть заключается во введении новых команд, повышающих производительность — так называемых *SIMD*.

*SIMD (single instruction, multiple data* — одиночный поток команд, множественный поток данных, ОКМД) — принцип компьютерных вычислений, позволяющий обеспечить параллелизм на уровне данных. *SIMD*-процессоры называются также векторными.

*SIMD*-компьютеры состоят из одного командного процессора (управляющего модуля), называемого контроллером, и нескольких модулей обработки данных, называемых процессорными элементами. Управляющий модуль принимает, анализирует и выполняет команды. Если в команде встречаются данные, контроллер рассылает на все процессорные элементы команду, и эта команда выполняется на нескольких или на всех процессорных элементах. Каждый процессорный элемент имеет свою собственную память для хранения данных. Одним из преимуществ данной архитектуры считается то, что в этом случае более эффективно реализована логика вычислений. До половины логических инструкций обычного процессора связано с управлением выполнением

машинных команд, а остальная их часть относится к работе с внутренней памятью процессора и выполнению арифметических операций. В *SIMD* компьютере управление выполняется контроллером, а «арифметика» отдана процессорным элементам.

Для *графических станций* максимальная вычислительная мощность важнее, чем максимальное быстродействие процессора. Повышение пропускной способности магистралей процессор — память и память — периферийные устройства практически невозможно (или же крайне дорого) при сохранении приемлемой IBM-совместимости. Эта причина (в числе других, не менее важных) и привела к созданию целых семейств профессиональных компьютеров. Они могут быть узкоспециализированными (например, серверы баз данных) или универсальными, но их объединяет то, что совместимость в них обеспечивается на программном, а не на аппаратном уровне, как в PC.

Разница по скорости между самыми современными персональными компьютерами определяется в основном уже не наличием более быстрого процессора, а сбалансированной и быстродействующей периферией, быстрой памятью, мощным 3D-акселератором.

"Сердцем" графической станции, как правило, являются *RISC*-процессоры.

*RISC (Restricted (reduced) instruction set compute* — компьютер с упрощенным набором команд) — архитектура процессора, в которой быстродействие увеличивается за счёт упрощения команд, чтобы их декодирование было проще, а время выполнения — короче. Первые *RISC*-процессоры даже не имели команд умножения и деления. Это также облегчает повышение тактовой частоты и делает более эффективной *суперскалярность* (распараллеливание команд между несколькими исполнительными блоками).

В них жестко "защит" минимально необходимый набор команд, которые выполняются очень быстро, обычно за один



такт процессора; иногда выполняются и несколько команд в один такт. Необходимое для работы ЭВМ количество команд обеспечивается микропрограммно, т. е. часть команд на процессорном уровне подразделяется на микропрограммные, состоящие в свою очередь из некоторого количества элементарных команд. Для программиста эти команды практически неразличимы. RISC-архитектура процессора обеспечивает большую производительность и гибкость.

Комплекс, предназначенный для формирования изображений, которые выводятся на экран монитора - это *видеосистема компьютера*.

*Видеосистема* современного компьютера состоит из обязательной графической (формирующей изображение программно) и дополнительной подсистемы обработки видеоизображений. Обе эти составляющие части обычно используют общий монитор, а соответствующие аппаратные средства системного блока могут располагаться на отдельных картах или объединяться на одном комбинированном адаптере, который можно назвать адаптером дисплея (*Display Adapter*).

Графический адаптер служит для программного формирования графических и текстовых изображений и является промежуточным элементом между монитором и шиной компьютера. Изображение строится по программе, исполняемой центральным процессором, в чем ему могут помогать графические акселераторы и сопроцессоры. В монитор адаптер посылает сигналы управления яркостью лучей RGB и синхросигналы строчной и кадровой разверток.

Все компоненты дисплейного адаптера могут размещаться на одной карте расширения, а зачастую они устанавливаются прямо на системной плате, используя преимущества локального подключения к системной шине.

### *Принципы вывода изображений*

Видеосистема РС ориентирована на *растровый метод вывода изображения*. Растровый метод подразумевает, что некий рисующий инструмент, способный оставлять видимый след, сканирует всю поверхность, на которую выводится изображение. Траектория движения инструмента постоянна и не зависит от выводимого изображения, но инструмент может рисовать, а может и не рисовать отдельные точки траектории. Видимым изображением являются оставленные им точки.

Для качественного изображения экран должен иметь как можно больше точек матрицы разложения - то есть строк в кадре и точек на строке. Частотные параметры видеосистемы определяются исходя из желаемой частоты кадров, разрешения экрана и режима развертки. Заботясь о зрении пользователя, частоту кадров стремятся повышать. При низкой частоте экран начинает мерцать, что особо заметно на больших белых полях изображения. Разрешение экрана стремятся увеличить - чем оно выше, тем больше информации можно уместить на экране.

Поскольку размер экрана постоянно увеличивается - 17" монитор является уже нормой для многих видов деятельности, - потребность в разрешении, скажем, 1600x1200 вполне реальна. Но для этого уже требуется полоса 120 МГц! (Кадровая частота 50 Гц - это отнюдь не идеал.) Заметим, что чем выше частота развертки, тем ниже производительность графической системы при построении изображений.

### *Графический режим*

В графическом режиме имеется возможность индивидуального управления свечением каждой точки экрана монитора независимо от состояния остальных. В графическом режиме каждой точке экрана - пикселу - соответствует ячейка специальной памяти, которая сканируется схемами адаптера синхронно с движением луча монитора. Эта постоянно циклически сканируемая (с кадровой частотой) память



называется видеопамью (Video Memory), или VRAM (Video RAM). Процесс постоянного сканирования видеопамью называется регенерацией изображения, и, к счастью, этого же сканирования оказывается достаточно для регенерации информации микросхемам динамической памяти, применяемой в этом узле. Для программно-управляемого построения изображений к видеопамью также должен обеспечиваться доступ и со стороны системной магистрали компьютера, причем как по записи, так и по чтению. Количество бит видеопамью, отводимое на каждый пиксел, определяет возможное число состояний пиксела - цветов, градаций яркости или иных атрибутов (например, мерцание). Так, при одном бите на пиксел возможны лишь два состояния - светится или не светится. Два бита на пиксел - можно иметь одновременно четыре цвета на экране. Четыре бита на пиксел (16 цветов), обеспечиваемые адаптером EGA, были достаточны для многих графических приложений (например, графики в САПРах). Пределом мечтаний в свое время было 256 цветов (8 бит на пиксел) адаптера VGA. Сейчас остановились на режимах High Color (15 бит - 32768 цветов или 16 бит - 65536 цветов), а для профессионалов - True Color - "истинный цвет" (24 бит - 16,7 миллиона цветов), реализуемых современными адаптерами и мониторами SVGA. 15 и 24 бита распределяются между базисными цветами R:G:B поровну (5:5:5 и 8:8:8), 16 бит - с учетом особенностей цветовосприятия неравномерно (5:6:5 или 6:6:4).

Под интеллектом графического адаптера подразумевается наличие на его плате собственного процессора, способного формировать растровое изображение в видеопамью (bitmap) по командам, полученным от центрального процессора. Команды ориентируются на наиболее часто используемые методы описания изображений, которые строятся из отдельных графических элементов более высокого уровня, чем пикселы.

Команды рисования (Drawing Commands) обеспечивают построение графических примитивов - точки, отрезка прямой, прямоугольника, дуги, эллипса. Примитивы такого типа в командах описываются в векторном виде, что гораздо компактнее, чем их растровый образ. Таким образом, удается значительно сократить объем передаваемой графической информации за счет применения более эффективного способа описания изображений.

Современные графические адаптеры берут на себя и многие функции построения трехмерных изображений. Трехмерное изображение должно состоять из ряда поверхностей различной формы. Эти поверхности "собираются" из отдельных элементов полигонов, чаще треугольников, каждый из которых имеет трехмерные координаты вершин и описание поверхности (цвет, узор). Перемещение объектов (или наблюдателя) приводит к необходимости пересчета всех координат. Для создания реалистичных изображений учитывается перспектива - пространственная и атмосферная (дымка или туман), освещенность поверхностей и отражение света от них, прозрачность и многие факторы.

#### *Текстовый режим*

В символьном или текстовом режиме формирование изображения происходит несколько иначе. Если в графическом режиме (APA) каждой точке экрана соответствует своя ячейка видеопамью, то в текстовом режиме ячейка видеопамью хранит информацию о символе, занимающем на экране знакоместо определенного формата. Знакоместо представляет собой матрицу точек, в которой может быть отображен один из символов из определенного набора. В ячейке видеопамью хранится код символа, определяющий его индекс в таблице символов, и атрибуты символа, определяющие вид его отображения. К атрибутам



относится цвет фона, цвет символа, инверсия, мигание и подчеркивание символа.

В текстовом режиме экран организуется в виде матрицы знакомест, образованной горизонтальными линиями и вертикальными колонками. Этой матрице соответствует аналогичным образом организованная видеопамять. Адаптер, работающий в текстовом режиме, имеет дополнительный блок - знакогенератор. Во время сканирования экрана выборка данных из очередной ячейки видеопамати происходит при подходе к соответствующему знакоместу, причем одна и та же ячейка видеопамати будет выбираться при проходе по всем строкам растра, образующим линию знакомест. Считанные данные попадают в знакогенератор, который вырабатывает построчную развертку соответствующего символа.

Данный режим при работе с графикой не используется.

#### *Дисплей*

Самым главным устройством вывода визуальной информации является дисплей (display - устройство отображения). Дисплей может быть основан на различных физических принципах: здесь применимы электронно-лучевые трубки, газоплазменные матрицы, жидкокристаллические индикаторы и другие приборы.

#### *Параметры монитора*

Мониторы подразделяются на монохромные (Monochrome или Mono) и цветные (Colour или Color). Монохромные мониторы могут быть как черно-белыми, так и черно-зелеными или черно-желтыми. Черно-белые мониторы до сих пор используются с адаптерами классов VGA и SVGA. Эти мониторы сочетают высокую разрешающую способность (у них отсутствуют трехцветные зерна люминофора) с низкой ценой. Высокая четкость изображения при режимах высокого разрешения позволяет длительно работать с текстом, не

утомляя глаза. Цветные мониторы получили наибольшее распространение.

Главным параметром монитора является размер диагонали экрана Screen Size, который принято измерять в дюймах. По умолчанию считается, что ширина экрана больше его высоты и соотношение этих размеров составляет 4:3.

#### *Дисплейные адаптеры*

Дисплейным адаптером условимся называть блок компьютера, к которому подключается дисплей. Поскольку адаптер предназначен для подключения монитора, его обязательным элементом будет контроллер ЭЛТ (CRT Controller). В задачу этого контроллера входит согласованное формирование сигналов сканирования видеопамати (адрес и стробы чтения) и сигналов вертикальной и горизонтальной синхронизации монитора. Строб - в системах с синхронизацией по тактовой частоте - период времени, выделенный для осуществления определенной операции, а также сигнал о необходимости выполнения такой операции. Контроллер ЭЛТ должен обеспечивать требуемые частоты развертки и режимы сканирования видеопамати, которые зависят от режима отображения (графический или текстовый) и организации видеопамати, о чем говорилось ранее. Опорной частотой для работы контроллера является - частота вывода пикселей в графических режимах или точек разложения символов в текстовом режиме.

Видеопамать является специальной областью памяти, из которой организуется циклическое чтение содержимого для регенерации изображения. Необходимый объем видеопамати определяется желаемым графическим режимом. Трактровка данных видеопамати зависит от используемого видеорежима.



## 2.2. Программное обеспечение автоматизированного выполнения графических изображений

Вывод изображения на экран дисплея и разнообразные действия с ним, в том числе и визуальный анализ, требуют от пользователя известной геометрической грамотности. Геометрические понятия, формулы и факты, относящиеся, прежде всего к плоскому и трехмерному случаям, играют в задачах компьютерной графики особую роль.

Геометрические фигуры представлены в памяти ЭВМ числовыми моделями, содержащими параметры формы и параметры положения фигуры, а также геометрические условия формирования фигуры.

В начертательной геометрии рассматриваются способы описания фигур с помощью параметров. Минимальное количество параметров, необходимых для описания фигуры, называется ее параметрическим числом.

Точка задается двумя параметрами на плоскости и тремя в пространстве. Прямая задается двумя параметрами на плоскости и четырьмя в пространстве. Плоскость тремя параметрами. Точки, прямые и плоскости описываются параметрами положения. Отрезок прямой на плоскости задается четырьмя и в пространстве шестью параметрами, из которых один (длина отрезка) является параметром формы. Окружность задается тремя параметрами, из которых один (радиус окружности) является параметром формы.

Числовые модели фигур могут быть заданы аналитически в виде неявного уравнения либо параметрически в виде параметрического уравнения.

## 2.3. Устройства для ввода графической информации

Устройствами ввода графической информации, называются устройства, предназначенные для преобразования геометрографической информации, находящейся на твердых носителях, в компьютерное представление. К ним относятся: клавиатура, манипулятор «мышь», трекбол, трекпоинт, джойстик, световое

перо, графический планшет с указкой – карандашом, сенсорный дисплей, сканер, цифровая фото- и видеокамера, аналого-цифровые преобразователи и системы датчиков.

*Компьютерная клавиатура* - одно из основных устройств ввода информации от пользователя в компьютер.

*Манипулятор «мышь»* - одно из указательных устройств ввода, обеспечивающее интерфейс пользователя с компьютером.

*Трекбол* - указательное устройство ввода информации об относительном перемещении для компьютера. Аналогично мыши по принципу действия и по функциям. Трекбол функционально представляет собой перевернутую механическую (шариковую) мышь.

*Трекпоинт* - устройство, позволяющее выбирать данные на дисплее, вводить графические данные. Представляет собой кнопку, которая отслеживает на экране дисплея с помощью курсора направление давления пальца на нее. Обычно устанавливается в центре клавиатуры. Также миниатюрный тензометрический джойстик, применяемый в ноутбуках как замена мыши.

*Джойстик* - устройство ввода информации, которое представляет собой манипулятор, посредством которого можно задавать экранные координаты графического объекта; также может выполнять функции клавиатуры.

*Световое перо* - один из инструментов ввода графических данных в компьютер, разновидность манипуляторов.

*Дигитайзер (или графический планшет)* – это устройство, облегчающее ввод графической информации, например при рисовании, так как устройство ввода выполнено в форме шариковой ручки или пера.

*Сканер* – это устройство, для ввода графической информации с твердого носителя (бумаги), служит для преобразования графической информации в компьютерное представление.



#### 2.4. Устройства вывода графической информации

К устройствам вывода графической информации относятся: монитор, принтер, плоттер (графопостроитель), установка быстрого прототипирования (трехмерный принтер).

*Растровый дисплей* (монитор с электронно-лучевой трубкой или жидкокристаллический) служит для представления изображения на экране.

Альтернатива растровым устройствам - *векторные устройства вывода* изображения. В этих устройствах инструмент прорисовывает только изображаемые фигуры и его траектория движения определяется выводимым изображением. Изображение состоит из *графических примитивов*, которыми могут быть отрезки прямых - векторы (откуда и название метода вывода), дуги, окружности.

*Принтер* (от английского print — печать; печатающее устройство) — периферийное устройство компьютера, предназначенное для перевода текста или графики на физический (твердый) носитель из электронного вида малыми тиражами (от единиц до сотен) без создания печатной формы.

По возможности печати графической информации принтеры делятся на алфавитно-цифровые (с возможностью печати ограниченного набора символов) и графические.

По принципу переноса изображения на носитель принтеры делятся на:

- матричные;
- лазерные (также светодиодные принтеры);
- струйные;
- сублимационные
- твердочернильные.

По количеству цветов печати — на чёрно-белые (монохромные) и цветные.

*Матричные* принтеры — старейшие из ныне применяемых типов принтеров. Изображение формируется печатающей головкой, которая состоит из набора иголок (игольчатая матрица), приводимых в действие электромагнитами. Головка

передвигается построчно вдоль листа, при этом иголки ударяют по бумаге через красящую ленту, формируя точечное изображение.

*Лазерные* принтеры используют в работе процесс ксерографической печати, при котором изображение формируется путем сканирования лазерным лучом фоточувствительных элементов принтера. Центральным печатающим механизмом является металлическая трубка, покрытая пленкой из органического фоточувствительного проводника. Ее называют фотовалом.

*Достоинства лазерных принтеров:*

- Обладают большой скоростью печати, так как лазерный луч движется намного быстрее ПГ струйного принтера
- За счет точности лазера имеют более высокое разрешение печати
- Экономичней струйного по расходникам.

*Недостатки лазерных принтеров:*

- Высокая стоимость (особенно цветной)
- Массивность – занимают много места
- Дорогие картриджи
- Окись углерода в составе соединений тонера при большой концентрации в воздухе вызывает головную боль, слабость, сонливость и учащение пульса
- Высокое пиковое энергопотребление.

В *струйных* принтерах используется матрица, печатающая жидкими красителями, а изображение на носителе формируется из точек. Основной и самой дорогой деталью таких принтеров является печатающая головка (кратко - ПГ). У некоторых принтеров она встроена в картридж (в основном характерна для принтеров компании HP).

*Достоинства струйных принтеров:*

- Дешев
- Занимает мало места



- Возможность установки системы непрерывной подачи чернил, с помощью которой можно существенно экономить на печати, особенно фото
- Возможность заправлять картриджи самостоятельно
- Большой выбор в типе печатных носителей
- Полноценная печать фотографий высокого качества.

#### *Недостатки струйных принтеров:*

- Медленная печать
- При длительном простое могут высохнуть чернила
- Высокая стоимость оригинальных расходников при их малом ресурсе.

Основное применение *термосублимационных* принтеров в настоящее время – фотопечать. Термосублимация (возгонка) — это быстрый нагрев красителя, когда минует жидкая фаза. Из твердого красителя сразу образуется пар. Чем меньше порция, тем больше фотографическая широта (динамический диапазон) цветопередачи. Пигмент каждого из основных цветов, а их может быть три или четыре, находится на отдельной (или на общей многослойной) тонкой лавсановой ленте. Печать окончательного цвета происходит в несколько проходов: каждая лента последовательно протягивается под плотно прижатой термоголовкой, состоящей из множества термоэлементов. Эти последние, нагреваясь, возгоняют краситель. Точки, благодаря малому расстоянию между головкой и носителем, стабильно позиционируются и получают весьма малого размера.

*Термопринтеры* - процесс печати состоит в формировании изображения термической печатной головкой на специальной термочувствительной бумаге, которая чернеет в местах нагрева, образуя символы.

#### *Термопринтеры:*

- Просты и дешёвы
- Не требуют красящего вещества
- Качество печати невысокое
- Цветопередача только чёрно-белая.

*Плоттеры и графопостроители* служат для вывода чертежей, графиков, качественных изображений на бумагу или синтетическую пленку. К векторным устройствам вывода статических изображений относятся перьевые плоттеры.

*Установка быстрого прототипирования (трехмерный принтер)* – это устройство, предназначенное для преобразования компьютерной модели объекта в физический макет (прототип) этого объекта.

### 3. ГЕОМЕТРИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ЕГО ЗАДАЧИ

Вывод изображения на экран дисплея и разнообразные действия с ним, в том числе и визуальный анализ, требуют от пользователя известной геометрической грамотности. Геометрические понятия, формулы и факты, относящиеся, прежде всего к плоскому и трехмерному случаям, играют в задачах компьютерной графики особую роль.

Первой основой для описания предметов можно считать Евклидову геометрию, допускающую однозначное представление материальных объектов на плоскости. Евклидово построение предполагает определенный набор инструментов (линейка, циркуль) и множество допустимых операций, которые можно выполнить с их помощью. Дальнейшее развитие идей Евклидовых построений сформировало методы начертательной геометрии и проекционного черчения.

Введение систем координат Декартом позволило соединить геометрию с аналитической математикой. Так, открылась возможность получать новые геометрические объекты путем решения алгебраических уравнений.

Геометрические фигуры представлены в памяти ЭВМ числовыми моделями, содержащими параметры формы и параметры положения фигуры, а также геометрические условия формирования фигуры.



В геометрии рассматриваются способы описания фигур с помощью параметров. Минимальное количество параметров, необходимых для описания фигуры, называется ее параметрическим числом.

Каждой точке  $M$  ставится в соответствие упорядоченная пара чисел  $(x, y)$  ее координат (рис. 3.1).

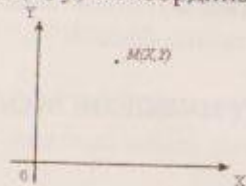


Рис. 3.1

Точка на плоскости задается двумя числами  $(x, y)$ , определяющими ее положение относительно начала координат и тремя в пространстве.

Прямая линия описывается уравнением  $y=kx+b$  и задается двумя параметрами на плоскости ( $k$  и  $b$ ) и четырьмя в пространстве.

Отрезок прямой на плоскости задается четырьмя (координаты  $x_1$  и  $x_2$  начала и конца отрезка) и в пространстве шестью параметрами, из которых один (длина отрезка) является параметром формы.

Плоскость задается тремя параметрами.

Точки, прямые и плоскости описываются параметрами положения.

*Кривая второго порядка*

Формула кривой второго порядка может выглядеть, например, так:

$$x^2+a_1y^2+a_2xy+a_3x+a_4y+a_5=0.$$

Как видно, пяти параметров вполне достаточно для описания бесконечной кривой второго порядка.

К кривым второго порядка относятся параболы, гиперболы, эллипсы, окружности и другие линии, уравнения которых не содержат степеней выше второй. *Окружность*

задается тремя параметрами, из которых один (радиус окружности) является параметром формы.

Отличаются кривые второго порядка тем, что не имеют точек перегиба.

Числовые модели фигур могут быть заданы аналитически в виде неявного уравнения либо параметрически в виде параметрического уравнения.

Для эффективного использования этих известных формул в задачах компьютерной графики более удобной является их матричная запись.

В начертательной геометрии есть понятие определителя фигуры, который состоит из частей: постоянной и переменной. Первая геометрическая, вторая алгоритмическая части определителя. Алгоритмическая часть определителя задает алгоритм воспроизводства всего множества точек, принадлежащих фигуре, которая имеет данную геометрическую часть определителя. Все эти понятия являются фундаментальными для компьютерной графики.

*Аффинные преобразования*

Аффинное преобразование это такое преобразование, которое сохраняет параллельность линий, но не обязательно углы или длины.

Наиболее часто применяются следующие приемы компьютерной графики: *перенос, масштабирование, поворот, отражение.*

Под *переносом* понимается смещение примитивов вывода на один и тот же вектор.

*Масштабирование* это увеличение или уменьшение всего изображения либо его части. При масштабировании координаты точек изображения умножаются на некоторое число.

Под *поворотом* понимается вращение примитивов вывода вокруг заданной оси. ( В плоскости чертежа вращение происходит вокруг точки.)



Под *отражением* понимают получение зеркального отображения изображения относительно одной из осей (например  $X$ ).

К изложенным фундаментальным преобразованиям сводятся следующие:

*прокручивание* - перемещение окна на поверхности визуализации (если перемещение ограничено только направлениями вверх и вниз, то оно называется вертикальным прокручиванием);

*трансфокация* - постепенное изменение масштаба изображения;

*кувыркание* - динамическое изображение примитивов вывода, вращающихся вокруг некоторой оси, ориентация которой непрерывно изменяется в пространстве;

*панорамирование* постепенный перенос изображения с целью создания зрительного ощущения движения.

Любое аффинное преобразование в трехмерном пространстве может быть представлено в виде суперпозиции вращений, растяжений, отражений и переносов. Очевидно, что для изменения положения некоторого тела в пространстве необходимо выполнить преобразование координат для каждой его точки.

Процесс создания моделей геометрических объектов, содержащих информацию о геометрических параметрах изделия, функциональную и вспомогательную информацию называют геометрическим моделированием.

Системы автоматизированного проектирования (САПР), или CAD-системы (*Computer-Aided Design*), - программные пакеты, предназначенные для создания чертежей, трехмерных моделей, конструкторской, технологической и других видов документации. Чертежно-конструкторская документация создается в различных системах проектирования, использующих векторные геометрические редакторы (EUCLID, CATIA, Solid Works, AutoCAD, КОМПАС и др.).

При разработке реального проекта, требующего принятия в государственных структурах, решающее значение имеет соблюдение государственных стандартов (ГОСТ).

Поэтому целесообразнее всего использовать векторный редактор, поддерживающий ГОСТ.

Основные преимущества чертежно-конструкторской документации, выполненной с применением компьютерных технологий, следующее: возможность неоднократного редактирования, высококачественная печать, создание неограниченного количества копий, возможность создания архивов.

Растровые графические редакторы не предназначены для выполнения чертежно-конструкторской документации. В основном они используются в художественной компьютерной графике.

### 3.1. Оформление чертежно-конструкторской документации

Под 2D (2-dimension) - графикой понимается создание и редактирование *плоских двухмерных геометрических изображений*.

Основная цель моделирования на плоскости - формирование контуров. Контур представляет собой последовательность отрезков прямых, дуг окружностей и сплайнов. При построении контуров могут быть использованы многогранники и кривые Безье. Точки используются для задания начала и конца контура и в качестве опорных точек пересечения соседних элементов.

Разработка конструкторской документации с помощью ЭВМ в режиме использования её в качестве «электронного кульмана», то есть создание плоских изображений - одна из возможностей использования графических редакторов.

Системы, одно из назначений которых - создание чертежно-конструкторской документации в электронном виде, относятся к векторным геометрическим редакторам.



В настоящее время получили достаточно широкое распространение различные системы проектирования, базирующиеся на геометро-графических редакторах (T - FLEX, Solid Works, AutoCAD, КОМПАС и др.). Одной из сильных сторон российской разработки КОМПАС является полная поддержка стандартов ЕСКД. При выборе векторного геометро-графического редактора для создания чертежно-конструкторской документации определяющим фактором является возможность соблюдение стандартов.

Связь между геометро-графическими редакторами осуществляется с помощью возможности использования различных форматов графических файлов. Форматы графических файлов BMP, JPEG, PNG и т.п. позволяет осуществлять связь между графическими редакторами.

Многие графические редакторы сохраняют файлы в собственном формате, но существует и множество универсальных форматов, способных восприниматься большинством программ. Если чертеж создан в одном векторном графическом редакторе, а возникла необходимость отредактировать его в другом, то наиболее рациональным действием будет сохранение файла в формате, поддерживаемом обоими редакторами.

### 3.2. Создание 3D-моделей объектов средствами компьютерной графики

Создание трехмерных моделей изделий с целью создания конструкторской и технологической документации, необходимой для их выпуска, - это 3D (3-dimension) моделирование.

Метод поверхностного моделирования наиболее эффективен при проектировании сложных криволинейных поверхностей.

Для описания формы поверхностей трехмерных объектов в системах компьютерной графики могут использоваться разнообразные методы.

Трехмерная модель геометрического объекта, в которой поверхность объекта описывается массивами вершин, ребер и многоугольных плоских граней (полигонов), называется *векторной полигональной (поверхностной) моделью*. Это модель более высокого уровня, чем каркасная модель.

*Воксельная модель* - это трехмерный растр, трехмерная модель геометрического объекта, в которой тело описывается массивом объемных элементов определенного размера. Подобно тому, как пиксели располагаются на плоскости 2D-изображения, так и воксели (объемные элементы) образуют трехмерные объекты в определенном объеме.

*Каркасная модель* геометрического объекта описывается точками и линиями (ребрами и вершинами). В каркасной модели нельзя отличить видимые грани объекта от невидимых (скрытых).

*Аналитической моделью* является описание поверхности математическими формулами.

*Физическая модель* объекта - это его макет, прототип, образец, созданный тем или иным способом для изучения его физических свойств.

Типом трехмерной модели геометрического объекта является *твердотельная модель*.

Во всех современных системах трехмерного моделирования построение твердотельной модели выполняется по общему принципу, который заключается в последовательном выполнении над объемными элементами *булевых операций: объединения, пересечения и вычитания*.

Логической операцией для формирования твердотельной модели из базисных тел является *вычитание*.

Неоспоримым преимуществом твердотельной модели геометрического объекта кроме построения графических изображений является возможность посчитать массу, центр тяжести, момент инерции объекта (т.е. рассчитать его *массинерционные* характеристики).



Системы трехмерного геометрического моделирования по сформированной 3D-модели объекта позволяют в автоматическом (или полуавтоматическом) режиме получать *любой проекционный вид* объекта.

После построения 3D-модели или сборки конструктор может получить её чертеж, избежав, таким образом, рутинного создания видов средствами плоского черчения. Плоский чертеж будет создан автоматически (либо полуавтоматически) и с абсолютной точностью, независимо от сложности модели.

В системах трехмерного моделирования построение нового объемного элемента выполняется (как правило) по принципу, который заключается *в перемещении в пространстве некоторого плоского элемента по определенному закону*. Такой способ образования объемных элементов является основным и называется *кинематическим*.

Вторым способом, имеющим менее широкие возможности для получения новых (нестандартных) элементов, является формирование объемных элементов из уже имеющихся геометрических примитивов.

В большинстве современных систем геометрического моделирования, к сожалению, отсутствует возможность по вычерченным стандартным видам в автоматическом режиме получить объемную модель элемента.

Способ поиска элемента в базе данных при решении поставленной задачи (построения нового объемного элемента) в общем случае не дает положительного результата, так как любые базы данных объемных элементов не обладают такой полнотой, чтобы в них можно было найти любую требуемую форму.

### 3.3. Программное обеспечение автоматизированного выполнения графических изображений

Наиболее распространенными программными комплексами машинной графики являются КОМПАС и AutoCAD.

Основные компоненты КОМПАС-3D – собственно система трехмерного твердотельного моделирования, чертежно-графический редактор и модуль проектирования спецификаций.

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

Чертежно-графический редактор (КОМПАС-ГРАФИК) предназначен для автоматизации проектно-конструкторских работ в различных отраслях деятельности. Он может успешно использоваться в машиностроении, архитектуре, строительстве, составлении планов и схем – везде, где необходимо разрабатывать и выпускать чертежную и текстовую документацию.

Совместно с любым компонентом КОМПАС-3D может использоваться модуль проектирования спецификаций, позволяющий выпускать разнообразные спецификации, ведомости и прочие табличные документы.

Документ-спецификация может быть ассоциативно связан со сборочным чертежом (одним или несколькими его листами) и трехмерной моделью сборки.

При разработке функций и интерфейса КОМПАС-3D учитывались приемы работы, присущие машиностроительному проектированию.

Элемент рабочего окна программы КОМПАС, изображенный на рисунке, называется *инструментальной панелью геометрии*.





Элемент рабочего окна программы КОМПАС, изображенный на рисунке, называется *инструментальной панелью редактирования*.

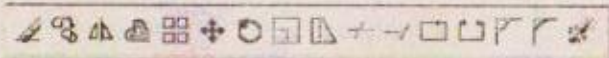


Операции *редактирования* являются операциями геометрического преобразования уже сформированных объектов.

**AutoCAD** (Computer-Aided Design) — двух- и трехмерная система автоматизированного проектирования и черчения, разработанная компанией Autodesk. AutoCAD является наиболее распространенной САПР в мире благодаря средствам черчения.

Компания Autodesk занимается разработкой системы автоматизированного проектирования AutoCAD с 1982 года. За это время были созданы тысячи дополнений и специализированные решения от сторонних фирм и самой компании Autodesk. На данный момент в мире насчитывается около шести миллионов пользователей AutoCAD.

С помощью элемента рабочего окна программы AutoCAD, изображенного на рисунке, выполняется редактирование графических примитивов



Фактические параметры геометрических примитивов, предназначенных для формирования изображения в векторных геометро-графических редакторах задаются пользователем в процессе работы.

Геометро-графический редактор **Microsoft Paint** не является векторным графическим редактором и не предназначен для выпуска чертежно-конструкторской документации. Microsoft Paint — простейший графический растровый редактор, поставляемый вместе с операционной системой Windows.

Геометро-графический редактор **Corel Draw** является векторным, но не предназначен для выпуска чертежно-конструкторской документации. Corel Draw — мощный двумерный векторный редактор. В настоящее время основная сфера его применения — подготовка файлов для графопостроителей и режущих плоттеров. Его форматы стали стандартом обмена плоскими векторными изображениями. Часто используется при создании наружной рекламы.

#### 4. ГРАФИЧЕСКИЕ ОБЪЕКТЫ

Графические объекты — это объекты, с помощью которых осуществляется вывод на экран изображений. Их использование в приложениях для создания графических изображений возможно на различных уровнях.

Графических объектов не так уж и много.

Рассмотрим геометрические объекты. По размерности параметрических моделей, необходимых для представления геометрических объектов, они делятся на нульмерные, одномерные, двумерные и трехмерные. Нульмерные и одномерные классы геометрических объектов могут моделироваться как в двух координатах (2D) на плоскости, так и в трех координатах (3D) в пространстве. Двухмерные и трехмерные объекты могут моделироваться только в пространстве.

В таблице приведена классификация базовых геометрических объектов.



Размерность объекта	Размерность пространства	Вид объекта
Нульмерные	На плоскости 2D	Точки на плоскости
		Точки на линии
	В пространстве 3D	Точки в пространстве
		Точки на поверхности
Одномерные	На плоскости 2D	Прямые
		Окружности
		Контуры
		Сплаины
		Кривые 2-го порядка
	В пространстве 3D	Векторы
		Прямые
		Сплаины
		Параметрическая кривая поверхность
		Линии пересечения поверхностей
		Проекция линии на поверхность
Двухмерные	В пространстве	Плоскости
		Поверхности вращения
		Линейчатые поверхности
Трехмерные	В пространстве	Тело вращения
		Тело сдвига
		Тело цилиндрическое
		Тело коническое
		Тело сферическое
		Тело призматическое

## 5. ПРИМИТИВЫ И ИХ АТТРИБУТЫ

Любой плоский графический объект можно представить как совокупность простых графических элементов, называемых **примитивами**. Каждая графическая система имеет собственный набор таких графических примитивов, из которых строятся более сложные объекты. Вместе с тем ряд примитивов является общим для всех систем. К ним относятся отрезки прямых линий, окружности и их дуги, кольца, эллипсы, прямоугольники, многоугольник, текстовые строки, кривые линии.

Простые примитивы:

- Точка
- Отрезок прямой линии, ломаная линия
- Круг (окружность)
- Дуга
- Прямоугольник
- Многоугольник
- Эллипс
- Сплайн
- Текст

К сложным примитивам относятся: полилиния, мультилиния, мультитекст, размер, выноска, допуск, штриховка, вхождение блока или внешней ссылки, атрибут и растровое изображение.

Каждый графический примитив имеет набор характеристик (**аттрибутов**), которые могут быть изменены пользователем. К основным атрибутам примитивов относятся:

- координаты опорных точек
- цвет
- толщина линий
- тип линий
- тип и размер шрифта (для текстовых строк).



## 6. ПРИМЕНЕНИЕ ИНТЕРАКТИВНЫХ ГРАФИЧЕСКИХ СИСТЕМ ДЛЯ ВЫПОЛНЕНИЯ И РЕДАКТИРОВАНИЯ ИЗОБРАЖЕНИЙ И ЧЕРТЕЖЕЙ

**Интерактивный режим (Interactive Mode)** – диалоговый режим взаимодействия пользователя с программой. Позволяет пользователю вводить *команды* и *данные* во время выполнения программы, управляя её работой с учетом выводимых программой результатов.

Программы массового применения, такие как текстовые и графические редакторы, электронные таблицы, обучающие программы, компьютерные игры, экспертные системы и др., работают в интерактивном режиме.

Реализованные в программе средства взаимодействия в интерактивном режиме называют *интерфейсом* пользователя (в современных программах обычно реализован графический интерфейс).

Интерактивный режим реализует принцип «вижу то, что делаю». В интерактивном режиме работы графических систем для указания конкретной точки в области формирования графического объекта используется специальный указатель – курсор.

Элегантный интерфейс пользователя стал неотъемлемой частью всякого успешного программного продукта. Достижения в создании новых мониторов, эргономике (изучении человеческого фактора) и в разработке привели к всеобщему распространению интерактивных методов и средств, прокладывающих себе дорогу с семидесятых. Многооконные системы позволяют выполнять одновременно несколько работ, мышь позволяет быстро указывать на требуемый элемент, меню ускоряет возможность выбора, значки представляют важные понятия, рисунки демонстрируют информацию визуально, кнопки выполняют стандартные операции.

Альтернативный режим является «пакетным», реализующий принцип «вижу только конечный результат».

### 6.1. Оконные функции

Под окном, обычно, понимают прямоугольную область, которая может быть отображена на экране монитора. Ею может быть или часть плоского графического объекта, или часть проекции пространственного объекта. Размеры окна могут быть не пропорциональны рабочей области (области отображения) экрана, однако вывод окна всегда производится без искажения пропорций объекта: окружность всегда останется окружностью, а квадрат – квадратом.

Каждая графическая система обладает гибкой системой оконных функций, позволяющих, с одной стороны, обеспечить необходимый уровень проработки каждой детали формируемого объекта, а с другой – оценивать весь объект целиком, оперировать с его отдельными частями. Благодаря наличию оконных функций становится возможным формировать достаточно большие по размеру сложные и насыщенные графические объекты, используя даже небольшую рабочую область экрана дисплея.

Функции:

- **Показать все.** Функция выводит в рабочую область экрана весь графический объект. При этом многие его мелкие детали могут быть не показаны, или сливаться с другими деталями. Тем не менее, эта функция очень полезная: она позволяет оценить весь объект целиком или перейти к заданию части объекта, с которой предполагается дальнейшая работа.

- **Задание окна просмотра (задание окна в окне).** Окно просмотра задается внутри текущего окна – той части графического объекта, которая в данный момент отображена на экране монитора. Для задания окна просмотра необходимо определить любые противоположные углы прямоугольной области, после чего окно будет автоматически отображено на экране.



- **Возврат к предыдущему окну.** Эта функция удобна при отработке мелких деталей графического объекта, расположенных в разных частях текущего окна.

- **Сдвиг окна просмотра.** Функция позволяет передвинуть текущее окно в пространстве графического объекта в любом направлении, не меняя его размеры. Эта функция полезна при вводе и редактировании графических элементов, попавших на границу текущего окна, либо при вводе длинных и узких объектов.

## 6.2. Операции над графическими объектами

Каждая графическая система имеет систему редактирования сформированного объекта. Эти функции предназначены не только для устранения ошибок, но и позволяют ускорить процесс его формирования. Работа большинства функций редактирования основана на выделении – пометке – части объекта (группы примитивов), подвергаемой той или иной процедуре. Пометка выполняется или заданием прямоугольной области вокруг части графического объекта (в этом случае помеченными становятся попавшие в окно примитивы), или непосредственным указанием курсора тех графических примитивов, которые должны быть включены в группу.

Операции:

- **Откатка.** Наиболее простая из функций редактирования. Она заключается в последовательной отмене последних действий, выполненных пользователем. Эта функция не требует задания группы редактирования. Каждое её выполнение приводит к удалению графического примитива, введенного последним, либо отмене любой другой функции редактирования.

- **Удаление (стирание).** Функция удаления стирает примитивы, включенные в группу редактирования. Эта функция используется не только для устранения допущенных

при формировании объекта ошибок, но может быть использована и для других целей.

Часто, при создании сложных объектов, необходимо выполнять предварительные вспомогательные построения, которые в дальнейшем необходимо удалять.

- **Сдвиг.** Используется для плоскопараллельного перемещения помеченной группы примитивов в пространстве графического объекта. Описав группу, можно переместить ее в любое место пространства объекта. Как правило, выделив графические примитивы, входящие в группу сдвига, необходимо также определить «точку привязки группы», так как при сдвиге группы сохраняется не только состав примитивов и их атрибуты, но и расположение примитивов относительно друг друга.

Эта функция полезна не только при исправлении ошибок, связанных с неточностями при состыковке отдельных частей графического объекта. Часто опытные пользователи описывают сложные части объекта отдельно, в стороне от других частей, а затем, используя функцию сдвига, вставляют описанную часть в нужное место графического объекта.

- **Копирование.** Функция позволяет создать копию «помеченных» примитивов и переместить ее в произвольное место пространства графического объекта.

При задании группы копирования необходимо определить «точку привязки группы», так как при копировании также сохраняется расположение примитивов относительно друг друга.

Эта функция используется не только для ввода повторяющихся частей объекта. Ею часто пользуются при задании его «похожих» частей. Скопировав группу примитивов, её можно затем отредактировать – «подправить». Такой режим намного эффективней, чем описание группы «с нуля». Опытные пользователи легко выделяют такие группы в описываемом объекте, тем самым значительно сокращая время работы над объектом



- **Размножение.** Функция используется при описании объектов с многократно повторяющимися группами примитивов. Описав группу один раз, можно многократно копировать её в любое место пространства объекта.

- **Поворот.** Эта функция используется для поворота группы «помеченных» примитивов в плоскости проекций. Для выполнения этой функции необходимо, кроме задания собственно группы, определить точку, относительно которой будет совершен поворот, а также угол, на который будет повернута группа относительно заданной точки. Те части графического объекта, которые находятся под углом, удобно описывать в не повернутом, привычном положении, а затем, выполнив функцию поворота. Расположить их требуемым образом.

- **Симметричное отображение.** Для графических объектов или их частей, имеющих ось (или плоскость) симметрии, нет необходимости в их полном описании. Достаточно описать лишь его половинку, а затем, воспользовавшись функцией симметричного отображения, автоматически получить объект целиком.

- **Растяжение, сжатие.** Эта функция позволяет производить линейное искажение группы примитивов вдоль одной или сразу двух координатных осей. Коэффициент искажения вдоль каждой оси задается или непосредственно в виде действительного числа или в процентах.

- **Сохранение в виде файла на диске.** Любая графическая система обладает средствами записи и чтения графических объектов. Объекты хранятся в виде файлов на устройствах с файловой структурой. Как правило, это магнитные или лазерные диски и электронные устройства хранения данных (карты памяти).

При формировании объекта на любой стадии его можно сохранить, пусть и в незавершенном виде. Необходимым атрибутом любого графического объекта является его имя, которое задается при первом его сохранении

- **Чтение из файла на диске.** Сложные графические объекты, как правило, невозможно описать за один сеанс работы пользователя с графической системой. Пользователь завершает работу с системой, выполняя, как правило, функцию «сохранить». Новый сеанс работы начинается с чтения записанного ранее объекта. Для того, чтобы прочитать именно тот объект, пользователю необходимо помнить его имя и ввести его в ответ на соответствующий запрос графической системы.

## 7. РЕШЕНИЕ ЗАДАЧ ГЕОМЕТРИЧЕСКОГО МОДЕЛИРОВАНИЯ

Наибольшее преимущества, по сравнению с ручными способами геометрических построений, можно добиться при использовании вычислительной техники именно при работе с пространственными объектами.

Пространственные объекты формируются из отсеков различных поверхностей. Каждая поверхность задается определителем, который, как известно, имеет геометрическую и алгоритмическую часть. Если для задания поверхности можно сформировать определитель или записать ее уравнением (формализовать задание поверхности), то такая поверхность называется регулярной.

Алгоритмические части обычно определяют отдельные команды графической системы, отвечающие за создание регулярных поверхностей. Для каждой такой команды необходимо задать графические объекты, необходимые для ее выполнения. Эти объекты и составляют геометрическую часть определителя, хотя в графических системах термин «геометрическая часть определителя» не используется.



## 8. СТАНДАРТЫ КОМПЬЮТЕРНОЙ ГРАФИКИ

Стандартизация в компьютерной графике направлена на обеспечение мобильности и переносимости прикладных программ, унификацию взаимодействия с графическими устройствами и обеспечение возможности обмена графической информацией между различными подсистемами. Использование стандартов позволяет сократить сроки разработки графических систем и увеличить их жизненный цикл. Сегодня в практике использования средств компьютерной графики применяется большое количество стандартов, различающихся по назначению и функциональным возможностям. Они имеют разную степень официальности - от фактических до международных стандартов.

Наиболее заметный след в двадцатипятилетней истории графических стандартов оставили:

- Core - Core Graphics System;
- GKS - Graphical Kernel System;
- MGKS - Minimal GKS;
- GKS-N - New Graphical Kernel System;
- CKS-3D - 3D Graphical Kernel System;
- PHIGS P - programmer's Hierarchical Interactive Graphics System;
- VDI - Virtual Device Interface;
- CGI - Computer Graphics Interface;
- PS - Adobe Systems PostScript Language;
- DPS - Adobe Systems Display Postscript System;
- NeWS - SUN Microsystems Network Extensible Window System;
- X Window - MIT X Window System;
- Windows - Microsoft Windows System;
- XGKS - X+ GKS;
- PEX - PHIGS+ X;
- OpenGL - SGI Graphical Language;
- VRML Virtual Reality Modeling Language;
- PREMO PResentation Environment for Multimedia Objects.

В основе разработки графических стандартов лежит принцип виртуальных ресурсов, позволяющий разделить графическую систему на несколько слоев - прикладной, базисный и аппаратно зависимый. При этом каждый слой является виртуальным ресурсом для верхних слоев и может использовать возможности нижних слоев с помощью стандартизованных программных интерфейсов. Кроме того, графические системы могут обмениваться информацией с другими системами или подсистемами с помощью стандартизованных файлов или протоколов.

В соответствии с этими соображениями первоначально были выделены *три основных направления стандартизации*:

- *базисные графические системы,*
- *интерфейсы виртуального устройства,*
- *форматы обмена графическими данными.*

Стандартизация *базисных графических систем* направлена на обеспечение мобильности прикладных программ и основана на концепции ядра, содержащего универсальный набор графических функций, общих для большинства применений.

Наиболее известными проектами по стандартизации базисных систем являются Core System, GKS, GKS-3D, PHIGS, PHIGS+. Основное направление развития этих проектов заключалось в усилении изобразительных возможностей для визуализации геометрических объектов (2D, 3D, удаление скрытых линий и граней, полутоновая закрашка и пр.). Стандарт на базисную графическую систему включает в себя функциональное описание и спецификации графических функций для различных языков программирования.

*Интерфейс виртуального устройства* разделяет аппаратно-зависимую и аппаратно-независимую части графической системы. Он обеспечивает заменяемость графических устройств (терминальную независимость), а также возможность работы с несколькими устройствами одновременно. Интерфейс виртуального устройства может



существовать в форме программного интерфейса и/или протокола взаимодействия двух частей графической системы. Эта концепция нашла применение в проектах CGI, X Window и MS Windows (а также NeWS и Display Postscript), обеспечивших удобные средства для манипулирования растровыми изображениями.

#### *Стандарты обмена данными. Метафайлы*

Стандарты обмена графическими данными можно условно разделить на следующие группы:

- графические метафайлы,
- проблемно-ориентированные протоколы,
- растровые графические файлы.

*Графический метафайл* представляет собой описание изображения в функциях виртуального графического устройства (в терминах примитивов и атрибутов). Он обеспечивает возможность запоминать графическую информацию единым образом, передавать ее между различными системами и интерпретировать для вывода на различные устройства. Таким образом, метафайл структура данных, которая содержит различные информационные записи: информацию об изображении, об устройстве, создавшем изображение, и об устройстве, на котором изображение должно быть восстановлено. Характеристики метафайла определяются его функциональными возможностями и способом кодирования информации. Метафайл обычно разрабатывается как составная часть какой либо графической системы. При этом его функциональные возможности однозначно соответствуют возможностям этой системы. Метафайл Windows представляет собой набор команд Windows GDI (интерфейс графических устройств graphics device interface набор функций, которые управляют выводом на монитор компьютера и другие устройства), которые участвуют в создании изображения. Эти команды

описания векторов, информация о цвете, растровые и текстовые данные записываются в виде инструкций, используемых для создания рисунка в Windows. В зависимости от того каким редактором будет прочитан файл, команды будут интерпретироваться по-разному: в векторном редакторе получим векторный рисунок, в растровом редакторе растровый рисунок.

В зависимости от выбранного способа кодирования метафайл может использоваться в качестве средства хранения и передачи изображений, протокола взаимодействия отдельных подсистем, языка описания изображений.

GKSM - GKS Metafile;

CGM - Computer Graphics Metafile;

NAPLPS - North American Presentation Level Protocol Syntax;

HPG - Hewlett Packard Graphics Language;

PostScript - Adobe Systems Language;

WMF - Microsoft Windows MetaFile;

GEM - GEM Draw File Format;

PIC - Lotus Graphics File Format;

SLD - AutoCad Slide File Format.

Как видно, наименование большинства из них происходит от соответствующих графических стандартов.

Сегодня в части стандартизации прикладных графических протоколов наиболее проработанной является область машиностроительных и электронных САПР. Здесь уже имеется ряд отраслевых и международных стандартов:

IGES - Initial Graphics Exchange Specification;

SET - Standard d'Exchange et de Transfert;

PDDI - Product Data Definition Interface;

MAP - Manufacturing Automation Protocol;

VDAFS - Verband der Deutschen Automobilindustrie-Flachen-Schnittstelle;

PDES - Product Data Exchange Standard;

STEP - Standard for Exchange Product Model Data;

EDIF - Electronic Design Interchange Format;

DXF - Autocad Data eXchange Format;



В других отраслях существуют пока только локальные стандарты, используемые в рамках одной или нескольких организаций.

### 8.1. Форматы графических файлов

Документы можно хранить в виде файлов, имеющих определенные форматы. Графическая информация может быть сохранена в файле одного из следующих форматов.

*Векторные форматы* используют для сохранения векторных изображений в сочетании с растровой графикой и текстами. Наибольшее распространение получили следующие векторные форматы: AI, CDR, CMX, WMF.

*Растровые форматы* файлов предназначены исключительно для растровых изображений: BMP, PCX, TIFF, CPT, PSD, GIF, JPEG. Форматы GIF и JPEG применяются для работы в Internet, а CPT и PSD применяются для сохранения многослойных изображений.

Растровые графические файлы стали активно применяться для хранения и транспортировки графической информации, в системах обработки данных и подготовки научно-технической документации, использующих персональные компьютеры, а также лазерные и струйные печатающие устройства. Основными характеристиками растровых файлов являются метод упаковки (сжатия) информации и тип поддерживаемой цветовой модели.

Первоначально растровые файлы содержали только статические изображения. В последнее время появились проекты по стандартизации форматов динамических (анимационных) изображений. Сегодня используется уже большое количество разнообразных форматов растровых файлов. Некоторые из них (например GIF, TIFF, PCX) получили широкое распространение и поддержку, другие ждут общественного признания, третьи поддерживаются только их разработчиками.

*Универсальные форматы* файлов (PS, EPS, PDF) используются языком PostScript и обеспечивают сохранение информации любого типа, как и векторные форматы, и открываются в любых графических форматах.

GIF - CompuServe Graphics Interchange Format;  
TIFF - Aldus & Microsoft Tag Image File Format;  
PCX - ZSoft PC Paintbrush format;  
RLE - CompuServe & Teletext Run Length Encoded;  
BMP - Microsoft Windows BitMaP;  
LBM - Deluxe Paint format;  
PIC - Pictor/PC Paint format;  
MAC - MacPaint format;  
IMG - Gem Paint format;  
CUT - Dr. Halo Cut files;  
TGA - Targa format;  
JPEG - Joint Photographic Experts Group;  
MEPG - Moving Pictures Experts Group;  
FLC - Autodesk Animator;  
AVI - Microsoft Animation Video.



## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Методические указания по изучению раздела «Компьютерная графика» дисциплины «Инженерная и компьютерная графика» для студентов специальности 200401 «Биотехнические и медицинские аппараты и системы» очной формы обучения / ГОУВПО «Воронежский государственный технический университет»; сост. Е.А. Балаганская, Е.К. Лахина. Воронеж, 2010. 24 с.

2. Методические указания по изучению раздела «Компьютерная графика» дисциплины «Начертательная геометрия. Инженерная графика» для студентов специальности 150202 «Оборудование и технология сварочного производства» очной формы обучения / ГОУВПО «Воронежский государственный технический университет»; сост. Е.А. Балаганская, Е.К. Лахина. Воронеж, 2010. 34 с.

3. <http://computer-graphics2.ru/topics/t3r1part1.html>

4. <http://kafedra-7.narod.ru/graf/1.htm#s111>

## СОДЕРЖАНИЕ

Введение .....	1
1. Понятие о компьютерной графике .....	2
1.1 Векторная графика.....	3
1.2 Растровая графика.....	6
1.3 Понятие о фрактальной графике.....	9
1.4 Цвет.....	10
1.5 Графические языки.....	14
2. Технические средства компьютерной графики.....	18
2.1 Графическая станция.....	19
2.2. Программное обеспечение автоматизированного выполнения графических изображений.....	28
2.3. Устройства для ввода графической информации.....	29
2.4 Устройства вывода графической информации.....	30
3. Геометрическое моделирование и его задачи .....	33
3.1 Оформление чертежно-конструкторской документации .....	37
3.2 Создание 3D-моделей объектов средствами компьютерной графики .....	38
3.3 Программное обеспечение автоматизированного выполнения графических изображений .....	40
4. Графические объекты .....	43
5. Примитивы и их атрибуты .....	45
6. Применение интерактивных графических систем для выполнения и редактирования изображений и чертежей .....	46
6.1 Оконные функции .....	47
6.2 Операции над графическими объектами .....	48
7. Решение задач геометрического моделирования ....	51
8. Стандарты компьютерной графики .....	52
8.1. Форматы графических файлов.....	56
Библиографический список .....	58