

МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕГО
ОБРАЗОВАНИЯ
РЕСПУБЛИКИ УЗБЕКИСТАН

ТАШКЕНТСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ АБУ РАЙХАНА
БЕРУНИЙ

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
КОНСПЕКТ ЛЕКЦИИ

2-часть

(для магистров технических специальностей)

Ташкент - 2007

Информационные технологии, конспект лекций
Составители: М.В.Сагатов, Р.М.Ирмухамедова, Равилов
Ш.М., Д.К.Каримова, ТГТУ, Ташкент, 2007, 149 стр.

Данный конспект лекции предназначен для магистров Ташкентского государственного технического университета - технических специальностей. В конспекте изложены материалы по современным средствам автоматизации обработки документов, средствам автоматизации научно-исследовательских работ, особенностям работы с системой MathCad, MathLab, использованию сервисов Интернет в научной деятельности, вопросам компьютерной безопасности, методы её реализации, языкам программирования в Интернете, HTML, PHP, JAVASCRIPT.

В конспекте имеется 16 графических изображений и 2 таблицы.

Издано на основе решения научно методического совета Ташкентского Государственного Технического Университета имени Абу Райхона Беруний. (пр. №4 от 04.06.2007)

Рецензенты: Каримов М.М - Ташкентский Университет Информационных Технологий, проректор по науке, д.т.н., проф.,
Арипов М.М. – Национальный университет Узбекистана, д.ф.м.н., проф.

Лекция № 1.

Автоматизация обработки данных.

Содержание:

1. Преобразование документов в электронную форму
2. Сканирование документа
3. Работа с программой FineReader
4. Средства автоматизация перевода документов

Компьютер предназначен для работы с документами, имеющими электронную форму. В то же время, нам часто приходится иметь дело с бумажными изданиями и документами: журналами, книгами, письмами, служебными записками и т. д. Чтобы в работе с информацией такого рода тоже можно было использовать компьютер, необходимы средства *преобразования бумажных документов в электронную форму*.

Если предполагается, что документ содержит в основном текстовую информацию, то можно выделить следующие основные этапы такого преобразования:

- в ходе *сканирования* при помощи устройств оцифровки изображения производится создание электронного образа (изображения) документа;
- процесс *распознавания* позволяет преобразовать электронное изображение в текстовые данные (с сохранением элементов форматирования оригинала или без них);
- для документов, исполненных на иностранном языке

применяют дополнительные средства *автоматизированного перевода на другой язык*.

Преобразование документов в электронную форму. Сканирование документов.

Процесс создания электронного изображения бумажного документа напоминает его фотографирование и требует применения соответствующего устройства. Сегодня в качестве такого устройства выступает *сканер*. Такие устройства, как *цифровые камеры*, пока не могут обеспечить для документов стандартного формата качество изображения, которое гарантировало бы их надежное распознавание.

Хорошие фотокамеры довольно дороги, а сканеры успешно завоевывают рынок товаров массового потребления и вполне доступны. С помощью сканера можно ввести в компьютер картинки из газет, журналов, книг и фотографий как целиком, так и частями, которые послужат вам строительным материалом для будущих композиций. Вы можете создавать эскизы и заготовки сначала на бумаге, а затем вводить их в компьютер посредством сканера и дорабатывать с помощью графических редакторов. Наконец, сканер просто незаменим, когда необходимо превратить бумажный печатный документ в текстовый, чтобы можно было открыть его в текстовом (а не в графическом) редакторе (например, в MS Word) для просмотра и редактирования.

Сканер является устройством для ввода изображений в компьютер. Исходные изображения (оригиналы) обычно находятся на непрозрачных (бумага) или прозрачных (слайды, фотопленка) носителях. Обычно это — рисунки, фотографии, слайды и/или тексты, но могут быть и объемные предметы. По

существу сканер является устройством, которое воспринимает оптическую информацию, доступную для нашего зрения, и сначала преобразует ее в электрическую форму, а затем приводит к цифровому виду, пригодному для ввода в компьютер. Таким образом, процесс сканирования оригинала состоит в его оцифровке. Оцифрованное изображение (на жаргоне — «скан») в дальнейшем может быть обработано в компьютере с помощью графического редактора (например, Photoshop), если это рисунок, или с помощью программы распознавания символов (например, FineReader), если это текст.

Существует множество моделей сканеров, отличающихся как техническими характеристиками и возможностями, так и ценой. Совсем не факт, что вам нужен самый могущественный и самый дорогой сканер. Новички, как правило, испытывают затруднения при выборе модели сканера и, в дальнейшем, при его использовании. Ошибка в выборе сканера выражается либо в том, что вы недоплатили чуть-чуть, либо слишком переплатили. Выбирая сканер, следует исходить из задач, которые вы собираетесь решать с его помощью. Сканеры могут использоваться для текущих задач офисов, домашнего коллекционирования фотографий и профессиональной работы с графикой. Для Web-дизайна, например, вы можете обходиться и самыми дешевыми сканерами. Но для работ, предназначенных, в конечном счете, для полиграфии, вам, возможно, потребуется более мощное устройство.

Чтобы ориентироваться среди множества параметров сканеров, следует понимать, на что они практически влияют и от чего зависят. В этой главе мы попытаемся помочь в разрешении этих проблем. Сначала нужно получить общее представление о принципах

построения и функционирования сканеров. Это совсем не трудно и не потребует много времени, но очень важно. Затем следует разобраться в основных параметрах (технических характеристиках) и освоить несколько типовых приемов использования сканеров. Наконец, необходимо узнать, как корректировать отсканированные изображения в графических и других редакторах.

Как устроены и работают сканеры

Для офисных и домашних задач, а также для большинства работ по компьютерной графике лучше всего подходят так называемые планшетные сканеры. Различные модели этого типа шире других представлены в продаже. Поэтому начнем с рассмотрения принципов построения и функционирования сканеров именно этого типа. Уяснение этих принципов позволит лучше понять значение технических характеристик, которые учитываются при выборе сканеров.

Планшетный сканер (Flatbed scanner) представляет собой прямоугольный пластмассовый корпус с крышкой. Под крышкой находится стеклянная поверхность, на которую помещается оригинал, предназначенный для сканирования. Через это стекло можно разглядеть кое-что из внутренностей сканера. В сканере имеется подвижная каретка, на которой установлены лампа подсветки и система зеркал. Каретка перемещается посредством так называемого шагового двигателя. Свет лампы отражается от оригинала и через систему зеркал и фокусирующих линз попадает на так называемую матрицу, состоящую из датчиков, вырабатывающих электрические сигналы, величина которых определяется интенсивностью падающего на них света. Эти датчики основаны на светочувствительных элементах, называемых приборами с

зарядовой связью (ПЗС, Couple Charged Device — CCD). Точнее говоря, на поверхности ПЗС образуется электрический заряд, пропорциональный интенсивности падающего света. Далее нужно только преобразовать величину этого заряда в другую электрическую величину — напряжение. Несколько ПЗС располагаются рядом на одной линейке.

Электрический сигнал на выходе ПЗС является аналоговой величиной (т.е. ее изменение аналогично изменению входной величины — интенсивности света). Далее происходит преобразование аналогового сигнала в цифровую форму с последующей обработкой и передачей в компьютер для дальнейшего использования. Эту функцию выполняет специальное устройство, называемое аналого-цифровым преобразователем (АЦП, Analog-to-digital Converter — ADC). Таким образом, на каждом шаге перемещения каретки сканер считывает одну горизонтальную полосу оригинала, разбитую на дискретные элементы (пиксели), количество которых равно количеству ПЗС на линейке. Все отсканированное изображение состоит из нескольких таких полос.

В цветных сканерах сейчас используются, как правило, трехрядная матрица ПЗС и подсветка оригинала калиброванным белым светом. Каждый ряд матрицы предназначен для восприятия одной из базовых цветовых составляющих света (красной, зеленой и синей). Чтобы разделить цвета, используют либо призму, разлагающую луч белого света на цветные составляющие, либо специальное фильтрующее покрытие ПЗС. Однако существуют цветные сканеры и с однорядной матрицей ПЗС, в которых оригинал по очереди подсвечивается тремя лампами базовых цветов. Однорядная технология с тройной подсветкой считается устаревшей.

Выше мы описали принципы построения и работы так называемых однопроходных сканеров, которые сканируют оригинал за один проход каретки. Однако еще встречаются, хотя больше и не выпускаются промышленностью, трехпроходные сканеры. Это сканеры с однорядной матрицей ПЗС. В них при каждом проходе каретки вдоль оригинала используется один из базовых цветных светофильтров: за каждый проход снимается информация по одному из трех цветовых каналов изображения. Эта технология также устарела.

Кроме CCD-сканеров, основанных на матрице ПЗС, имеются CIS-сканеры (Contact Image Sensor), в которых применяется фотоэлементная технология.

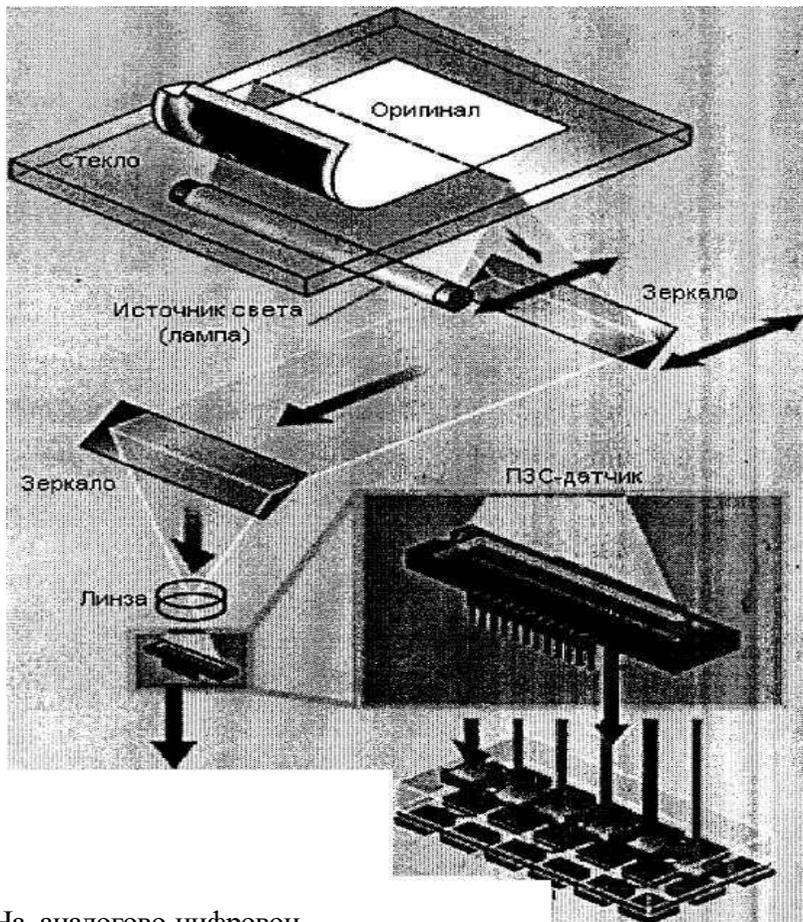
Светочувствительные матрицы, выполненные по этой технологии, воспринимают отраженный оригиналом свет непосредственно через стекло сканера без использования оптических систем фокусировки. Это позволило уменьшить размеры и вес планшетных сканеров более чем в два раза (до 3—4 кг). Однако такие сканеры хороши только для исключительно плоских оригиналов, плотно прилегающих к стеклянной поверхности рабочего поля. При этом качество получаемого изображения существенно зависит от наличия посторонних источников света (крышка CIS-сканера во время сканирования должна быть закрыта). В случае объемных оригиналов качество оставляет желать лучшего, в то время как ССО-сканеры дают неплохие результаты и для объемных (до нескольких см в глубину) предметов.

Планшетные сканеры могут быть снабжены дополнительными устройствами, такими как слайд-адаптер, автоподатчик оригиналов и др. Для одних моделей эти устройства предусмотрены, а для других нет.

Слайд-адаптер (Transparency Media Adapter, TMA) —

специальная приставка, позволяющая сканировать прозрачные оригиналы. Сканирование прозрачных материалов происходит с помощью проходящего, а не отраженного света. Иначе говоря, прозрачный оригинал должен находиться между источником света и светочувствительными элементами. Слайд-адаптер представляет собой навесной модуль, снабженный лампой, которая движется синхронно с кареткой сканера. Иногда просто равномерно освещают некоторый участок рабочего поля, чтобы не перемещать лампу. Таким образом, главная цель применения слайд-адаптера заключается в изменении положения источника света. Если же у вас есть цифровая камера (цифровой фотоаппарат), то слайд-адаптер, скорее всего, вам не нужен.

Если сканировать прозрачные оригиналы без использования слайд-адаптера, то нужно понимать, что при облучении оригинала количества отраженного и проходящего света не равны друг другу. Так, оригинал пропустит какую-то часть падающего света, которая затем отразится от белого покрытия крышки сканера и снова пройдет через оригинал. Какая-то часть света отразится от оригинала. Соотношение между частями проходящего и отраженного света зависит от степени прозрачности участка оригинала.



На аналогово-цифровой
преобразователь

Элементы ПЗС-датчик

Рис. 1.1. Схема устройства и работы планшетного сканера на основе ПЗС (CCD): свет лампы отражается от оригинала и через оптическую систему попадает на матрицу светочувствительных элементов, а затем на аналого-цифровой преобразователь (АЦП)

Таким образом, светочувствительные элементы матрицы сканера получают свет, дважды прошедший через оригинал, а также свет, отраженный от оригинала. Многократность прохода света через оригинал ослабляет

его, а взаимодействие отраженного и проходящего пучков света (интерференция) вызывает искажения и побочные видеоэффекты.

Автоподатчик — устройство, подающее оригиналы в сканер, которое очень удобно использовать при потоковом сканировании однотипных изображений (когда не нужно часто перенастраивать сканер), например, текстов или чертежей приблизительно одинакового качества.

Кроме планшетных, есть и другие типы сканеров: ручные, листопротяжные, барабанные, слайдовые, для сканирования штрих-кодов, скоростные для потоковой работы с документами.

Ручной сканер (Handheld Scanner) — портативный сканер, в котором сканирование осуществляется путем его ручного перемещения по оригиналу. По принципу действия такой сканер аналогичен планшетному. Ширина области сканирования — не более 15 см. Первые сканеры для широкого применения появились в продаже в 80-х годах XX века. Они были ручными и позволяли сканировать изображения в оттенках серого цвета. Теперь такие сканеры нелегко найти.

Листопротяжный или роликовый сканер (Sheetfed Scanner) — сканер, в котором оригинал протягивается мимо неподвижной линейной CCD- или CIS-матрицы, разновидность такого сканера — факс-аппарат.

Барабанный сканер (Drum Scanner) — сканер, в котором оригинал закрепляется на вращающемся барабане, а для сканирования используются фотоэлектронные умножители. При этом сканируется точечная область изображения, а сканирующая головка движется вдоль барабана очень близко от оригинала.

Слайдовый сканер (Film-scanner) — разновидность планшетного сканера, предназначенная для сканирования

прозрачных материалов (слайдов, негативных фотопленок, рентгеновских снимков и т. п.). Обычно размер таких оригиналов фиксирован. Заметим, что для некоторых планшетных сканеров предусмотрена специальная приставка (слайд-адаптер), предназначенная для сканирования прозрачных материалов (см. выше).

Сканер штрих-кодов (Bar-code Scanner) — сканер, предназначенный для сканирования товарных штрих-кодов. По принципу действия он сходен с ручным сканером и подключается к компьютеру, либо к специализированной торговой системе. При наличии соответствующего программного обеспечения распознавать штрих-коды может любой сканер.

Скоростной сканер для работы с документами (Document Scanner) — разновидность листопротяжного сканера, предназначенная для высокопроизводительного многостраничного ввода. Сканеры могут быть оборудованы приемными и выходными лотками объемом свыше 1000 листов и вводить информацию со скоростью свыше 100 листов в минуту. Некоторые модели этого класса обеспечивают двустороннее (дуплексное) сканирование, подсветку оригинала разными цветами для отсечки цветного фона, компенсацию неоднородности фона, имеют модули динамической обработки разнотипных оригиналов.

Итак, для дома и офиса лучше всего подходит планшетный сканер. Если вы хотите заниматься графическим дизайном, то лучше выбрать ССD-сканер (на основе ПЗС-матрицы), поскольку он позволяет сканировать и объемные предметы. Если вы собираетесь сканировать слайды и другие прозрачные материалы, то следует выбрать сканер, для которого предусмотрен слайд-адаптер. Обычно собственно сканер и подходящий к

нему слайд-адаптер продаются отдельно. Если не получается приобрести слайд-адаптер одновременно со сканером, то при необходимости вы сможете сделать это позже. Необходимо также определить максимальные размеры сканируемых изображений. В настоящее время типовым является формат А4, соответствующий обычному листу писчей бумаги. Большинство бытовых сканеров ориентированы именно на этот формат. Для сканирования чертежей и другой конструкторской документации обычно требуется формат А3, соответствующий двум листам формата А4, соединенным по длинной стороне. В настоящее время цены однотипных сканеров для форматов А4 и А3 сближаются. Можно предположить, что оригиналы, не превышающие по размерам формат А4, будут лучше обрабатываться сканером, ориентированным на формат А3.

Перечисленные выше параметры далеко не исчерпывают весь список, но на данном этапе нашего рассмотрения мы пока можем использовать только их. При выборе сканера решающими являются три аспекта: аппаратный интерфейс (способ подключения), оптико-электронная система и программный интерфейс (так называемый TWAIN-модуль). Далее мы рассмотрим их более подробно.

Разные модели сканеров понимают разные управляющие команды. Чтобы избежать разнобоя, был принят универсальный стандарт взаимодействия сканера и приложений. Этот стандарт называется *TWAIN*. Приложение посылает команды драйвера *TWAIN*, который преобразует их в инструкции, распознаваемые сканером. Таким образом, для приложения перестает иметь значение конкретная модель сканера. Операционная система Windows поддерживает интерфейс *TWAIN*, а все современные сканеры совместимы с ним и предоставляют необходимые драйверы нижнего уровня

Сканирование через посредство интерфейса *TWAIN* осуществляется следующим образом. Сначала следует включить сканер. Команда сканирования располагается в приложении в меню Файл (например, в программе Imaging соответствующий пункт и называется — Сканировать). После выбора этой команды открывается диалоговое окно драйвера *TWAIN*, вид которого зависит от модели сканера. В окне задают параметры сканирования: черно-белый или цветной режим, разрешающая коррекция яркости и контрастности. Большинство сканеров позволяют также известить предварительное черновое сканирование с низким разрешением и по результатам точно задать область сканирования — часть страницы документа».

Этап *распознавания* документа состоит в преобразовании электронного изображения (фактически набора цветных или черно-белых точек) в текстовый документ. Ранее для описания этого процесса обычно использовался термин *OCR (Optical Character Recognition)*, который соответствует одному из используемых методов. При таком подходе происходит «сравнение» элемента изображения с эталонными вариантами начертания символов, после чего выбирается наиболее подходящий символ. Этот подход требует использования специального комплекта шрифтов, но дает на нем \ наилучшие результаты. Современные алгоритмы распознавания не привязаны к конкретному начертанию символов, так же, как человек способен узнавать буквы при любых начертаниях (и даже при значительных искажениях).

В ходе распознавания сначала в изображении выделяются крупные элементы текста: колонки, абзацы, отдельные текстовые блоки (например, подрисуночные подписи), ячейки таблиц. Этот этап называют *сегментацией*, он может выполняться автоматически или вручную. После этого выполняется автоматический этап распознавания:

блоки разбиваются на строки, строки — на отдельные символы, каждый из которых распознается независимо и помещается в итоговый текстовый документ.

Работа с программой FineReader

Все операции, необходимые в ходе преобразования бумажного документа в электронную форму, могут быть выполнены с помощью программы **FineReader**

Эта программа способна выполнять сканирование и распознавание текстовых языках, в том числе и смешанных двуязычных текстов. С ее помощью выполнять пакетную обработку многостраничных документов, а также Гня вать режим распознавания для улучшения соответствия электронной!^ бумажному оригиналу при плохом качестве последнего или использований шрифтов, далеких от стандартных.

Основные операции обработки бумажного документа в программе RneReaden няются с помощью панели инструментов Scan&Read. С точки зрения этой процесс обработки документа состоит из пяти этапов:

- сканирование документа (кнопка Сканировать);
- сегментация документа (кнопка Сегментировать);
- распознавание документа (кнопка Распознать);
- редактирование и проверка результата (кнопка

Проверить);

- сохранение документа (кнопка Сохранить).

Сканирование документа. На этапе *сканирования* производится получение и-жения при помощи сканера и сохранение их в виде, удобном для последу обработки. Чтобы начать сканирование, надо включить сканер и щелкнуть на к Сканировать на панели инструментов Scan&Read.

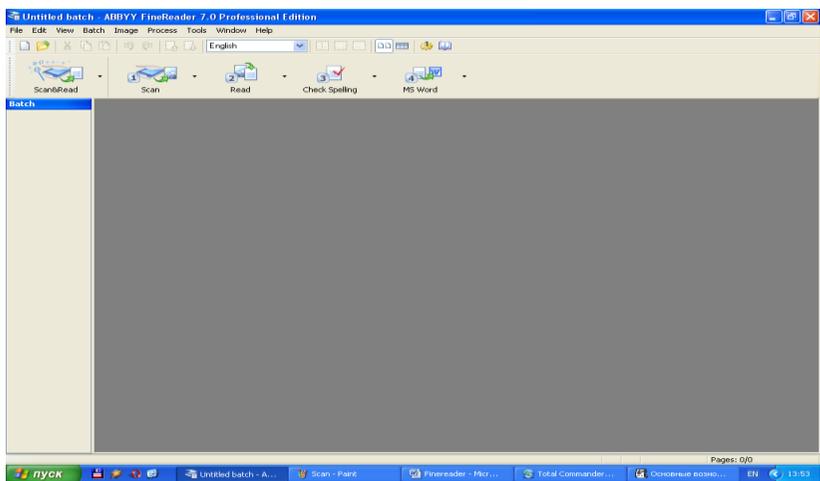


Рис. 1.2. рабочее окно программы **FineReader**

В программе FineReader сканирование может производиться как через драйвер **TWAIN**, так и в обход его. Первый способ используют, когда требуется точная настройка параметров сканирования, когда документ включает цветные иллюстрации, которые необходимо сохранить, а также когда разные страницы многостраничного документа сильно различаются по качеству. Второй вариант обеспечивает максимальную скорость и удобств! нирования.

Выбор используемого варианта осуществляется при помощи флажка: **Показывать диалог TWAIN-драйвера сканера (Сервис ► Опции ► Сканирование**. Процесс сканирования осуществляется автоматически и требует от пользователя, только вспомогательных операций, таких, как смена сканируемой страницы. Возможность вмешательства в работу программы заблокирована размещением на экране специального диалогового окна, уведомляющего о том, что идёт сканирование позволяющего прервать это процесс.

По завершении сканирования значки всех обработанных страниц отображаю окне **Пакет**. В основной

части рабочей области появляется окно **Изображение**, содержащее изображение текущей страницы. Добавлять страницы в пакет можно не только путем сканирования, но и путем открытия файлов с изображениями, имеющихся на компьютере.

Сегментация документа. Второй этап работы - *сегментация*, разбиение страницы на блоки текста. Естественный порядок распознавания - по строкам, расположенным на странице сверху вниз и идущим от левого края до правого. Если страница содержит колонки, иллюстрации, врезки, подрисуночные подписи или таблицы, то порядок распознавания требует коррекции.

Содержимое страницы разбивается на *блоки*, внутри каждого из которых, распознавание осуществляется в естественном порядке. Блоки нумеруются, исходя из порядка включения их в документ. При *автоматической сегментации* (кнопка **Сегментировать** на панели инструментов Scan&Read) определение границ блоков осуществляется автоматически. При этом учитываются поля документа, просветы между колонками, рамки.

Если структура страницы очень сложная, удобнее использовать *ручную сегментацию* или ручное редактирование результатов автоматической сегментации. Блоки отображаются в виде цветных прямоугольников с номером в левом верхнем углу. Новый блок создают протягиванием мыши по диагонали прямоугольника. Текущий; блок помечается выделенной линией, а его углы — прямоугольными маркерами. С помощью этих маркеров можно изменить размер или положение блока.

Команды редактирования блоков выведены на панель Инструменты. Они позволяют:

- объединить два блока в один (Добавить часть блока);
- удалить фрагмент блока (Удалить часть блока);
- изменить положение блоков (Переместить блоки);

- изменить порядок нумерации блоков (Перенумеровать блоки);

- изменить разбиение таблицы на ячейки (Добавить вертикаль, Добавить горизонталь, Удалить линии).

Разные типы блоков обрабатываются программой по-разному. Чтобы изменить тип блока, надо щелкнуть правой кнопкой мыши в его пределах и назначить новый тип с помощью меню Тип блока в контекстном меню. Программа FineReader поддерживает следующие типы блоков:

- текстовый (Текст) — на этапе распознавания преобразуется в текст;

- табличный (Таблица) — представляет собой набор ячеек, каждая из преобразуется в текст по отдельности;

- изображение (Картинка) — включается в документ без изменений как графическая иллюстрация, если формат сохранения преобразованного документа допускает вставные объекты;

- лишний (Нераспознаваемый) — игнорируется;

- содержащий штрих-код (Штрих-код) — распознается как штрих-код.

Распознавание текста. Процесс распознавания текста после сегментации на щелчка на кнопке **Распознать** и полностью автоматизирован. В ходе процесса отображается диалоговое окно **Распознавание**, позволяющее прервать процесс. Кроме того, в этом окне отображаются сообщения, указывающие на наличие проблем при распознавании. Проблемы обычно вызываются неверными настройками или качеством распознаваемого изображения. Если же дело в каких-то шрифтовых особенностях распознаваемого документа, применяют *распознавание с обучением*.

Распознавание с обучением. Распознавание с обучением состоит в формировании *эталона*, который

используется в ходе распознавания в дальнейшем. Эталон настраивается так, чтобы соответствовать определенному документу или однотипных документов. Чтобы создать эталон, используют команду **Сервис** **Редактор** **эталонов**

Новый эталон. После этого надо указать имя эталона и щелкнуть на кнопке ОК. Режим распознавания с обучением включается при настройке параметров работы программы (**Сервис** ► **Опции** ► **Распознавание**). На панели **Обучение** следует выбрать нужный эталон и установить флажок **Распознавание с обучением**.

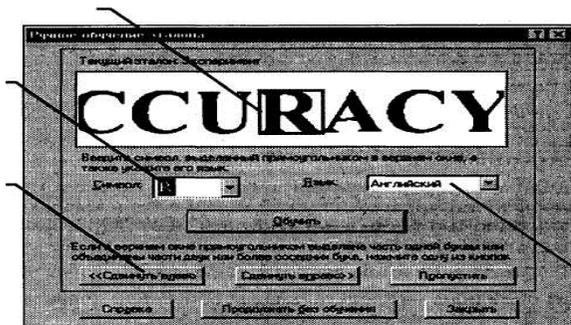
Когда в ходе распознавания с обучением программа FineReader обнаруживает символ, который не может интерпретировать однозначно, на экран выдается диалоговое окно **Ручное обучение эталона** (рис. 1.3). Программа указывает элемент изображения, вызвавший сомнения, и показывает, как именно он будет интерпретирован. Если допущена ошибка, можно указать нужный символ в поле Символ или уточнить область распознавания с помощью кнопок **Сдвинуть влево** и **Сдвинуть вправо**.

Затем надо щелкнуть на кнопке **Обучить**. Необходимые сведения сохраняются и используются при дальнейшем анализе изображения. Если число ошибок невелико, можно продолжить распознавание в обычном режиме щелчком на кнопке **Продолжить без обучения**.

**Расознаваемый
элемент**

Настройка символа

**Настройка
области
расознавания**



*Рис.1.3. Ручное «обучение» механизма
распознавания текста*

Редактирование документа. Когда распознавание данной страницы завершается, полученный текстовый документ отображается в окне Текст. Заключительные этапы работы позволяют отредактировать полученный текст с помощью средств, напоминающих текстовый редактор Word Pad (панель для форматирования открывается при помощи команды **Вид ► Панели инструментов ► Форматирование**). Провести проверку орфографии с учетом трудностей распознавания позволяет кнопка **Проверить** на панели инструментов Scan&Read.

Сохранение документа. По щелчку на кнопке **Сохранить** на панели инструментов Scan&Read запускается Мастер сохранения результатов. Он позволяет сохранить распознанный текст или передать его в другую программу (например, в Microsoft Word) для последующей обработки.

Обработка бланков. *Бланки, или формы*, представляют собой особый род документов. Они используются как анкеты, бюллетени для голосования, опросные листы и состоят из *постоянной части*, содержащей информацию, используемую в ходе заполнения бланка, и *переменной части*, куда при заполнении бланка заносятся данные. В ходе обработки бланков требуется получить внесенные в него данные и представить их в виде, удобном для дальнейшей обработки. При этом часто приходится иметь дело с тысячами однотипных бланков.

Для обработки бланков используется автономное приложение FineReader. Процесс работы с бланками несколько отличается от работы с обычными документами. Вначале подготавливается *шаблон*, который содержит все постоянные переменные зоны бланка. Этап сегментации заменяется *наложением шаблона*, то есть его совмещением с

постоянными элементами бланка. Это позволяет определить местонахождение переменных элементов бланка и провести их распознавание. Данные, полученные с отдельного бланка, рассматриваются как строка таблицы или как отдельная запись базы данных. Содержимое отдельного поля бланка соответствует ячейке таблицы.

Для создания шаблона требуется электронное изображение отдельного бланка и незаполненного. Чтобы создать шаблон, надо в приложении FineReader дать команду **Файл ► Новый**, после чего указать имя *пакета форм* и папку для хранения отсканированных бланков. Затем необходимо отсканировать или готовое изображение, которое будет использоваться в качестве основы шаблона.

Сам процесс создания шаблона состоит в *ручной сегментации бланка*. При этом кроме окна **Редактор шаблонов** открыто также диалоговое окно **Параметры**, определить как блоки, охватывающие фиксированные элементы бланка, которые содержат области, подлежащие заполнению. Блоки, соответствующие постоянным элементам, используются как приводные метки. Чтобы исключить такой блок из процесса распознавания, следует щелкнуть на нем правой: мышью и выбрать в контекстном меню команду **Тип блока ► Статический текст**.

Параметры блока задают на вкладке **Блок** диалогового окна **Параметры**. Для каждого распознаваемого блока надо установить флажок **Экспортируемый блок** указать имя поля базы данных. Информация из этого блока будет заноситься в указанное поле. После того как все нужные блоки созданы и настроены, щелкнуть на кнопке **Закреть** на панели инструментов. При этом производится проверка, обеспечивают ли заданные блоки возможность однозначного на шаблона на бланк.

В результате сканирования заполненного бланка, наложения шаблона и распознавания, полученные данные

представляются в виде формы, содержащей назван и данные, полученные при распознавании. Сохранение данных производят в формате, ориентированном на последующую обработку средствами электронных таблиц или баз данных, например, в виде электронной таблицы Excel (файл .XLS).

Средства автоматизации перевода документов

К средствам автоматизации перевода можно отнести два вида программ: электронные словари и программы перевода. *Электронные словари* представляют собой средства для перевода *отдельных слов*, отображаемых на экране или имеющих в документе. Удобство их использования состоит в возможности немедленно получить перевод неизвестного слова без поиска его в отдельном толстом томе. *Программы перевода* получают на входе текст, выполненный на одном языке или на другом языке, то есть автоматизируют перевод текста.

Электронные словари удобны для профессиональных переводчиков, которые выполняют большую часть работы по переводу вручную. Их также могут использовать лица, в целом знающие иностранный язык, если надо не обеспечить перевод документа, а просто ознакомиться с его содержанием.

Надежный и качественный автоматический перевод документов с одного языка на другой (мы будем говорить в основном о переводе с английского на русский) пока остается недостижимым идеалом. Причин для этого множество, и главная из них состоит в том, что перевод текста не сводится к переводу отдельных единиц. Преодолеть этот барьер современные программы автоматического перевода пока не могут.

Тем не менее, современные средства автоматизации перевода достигли того уровня, который позволяет эффективно использовать их на практике. Дело в том, что *технический текст*, в отличие от *художественного*, использует ограниченное количество языковых конструкций и более

ориентирован на однозначную интерпретацию. Среди используемых лексических единиц встречается большое число *терминов*, имеющих совершенно определенный смысл в рамках данной технической дисциплины. Это значительно упрощает процесс перевода и позволяет в отдельных случаях автоматически получать текст, близкий к результату подстрочного перевода.

Программы автоматического перевода имеет смысл использовать технических текстов в следующих случаях:

- при абсолютном незнании иностранного языка;
- при необходимости получить перевод *быстро*, даже ценой снижения его качества(например, это относится к переводу Web-документов);
- для перевода *на* иностранный язык (умения читать иноязычные тексты недостаточно, чтобы научиться объясняться на иностранном языке);
- для быстрого создания первоначального черновика («подстрочника»), используемого в ходе подготовки полноценного перевода.

Контрольные вопросы:

1. Назовите средства автоматизации обработки документов на компьютере.
2. Для чего предназначен сканер и какие виды сканеров вы знаете?
3. Какие этапы работ включает в себя процесс преобразования документ в электронную форму?
4. Для чего предназначена программа Fine Reader?
5. Какие программные средства используются для автоматического перевода документа на другие языки?
6. Основные инструменты программы Promt?

Лекция № 2,3.
Средства автоматизации научно-исследовательских работ. Работы с системой MathCad, MatLab.

Содержание:

1. Компьютер как инструмент научной работы.
2. Специализированная система MathCad и приемы работы с системой MathCad.
3. Специализированная система MATLAB.

Компьютер как инструмент научной работы

Вычислительная мощь компьютера позволяет использовать его как средство автоматизации научной работы. Для решения сложных расчетных задач используют программы, написанные специально. В то же время, в научной работе встречается широкий спектр задач ограниченной сложности, для решения которых можно использовать универсальные средства. К такого рода задачам относятся, например, следующие:

- подготовка научно-технических документов, содержащих текст и формулы, записанные в привычной для специалистов форме;
- вычисление результатов математических операций, в которых участвуют числовые константы, переменные и размерные физические величины;
- операции с векторами и матрицами;
- решение уравнений и систем уравнений (неравенств);
- статистические расчеты и анализ данных;

- построение двумерных и трехмерных графиков;
- тождественные преобразования выражений (в том числе упрощение), аналитическое решение уравнений и систем;
 - дифференцирование и интегрирование, аналитическое и численное;
- решение дифференциальных уравнений;
- проведение серий расчетов с разными значениями начальных условий и других параметров.

Рост сложности решаемых задач по объективным причинам ведёт к сложности алгоритмов и их реализаций на алгоритмических языках Си, Паскаль, Фортран и др. Ещё больше времени уходит на отладку кода. Эти причины привели к созданию Систем автоматизированного проектирования (САПР), в которые заложены некие алгоритмы. Такие системы появились достаточно давно, и были узкоспециализированными. Среди математических САПР наибольшую популярность приобрели MathCAD (MathSoft Inc.), Mathematica (Wolfram Research, Inc.), MATLAB (MathWorks Inc.), Maple V (Waterloo Maple Inc.).

Программы MathCad, MatLab, которые представляет собой автоматизированную систему, позволяющую динамически обрабатывать данные в числовом и аналитическом (формульном) виде.

Специализированная система MathCad и приемы работы с системой MathCad.

MathCAD является математическим редактором, позволяющим проводить разнообразные научные и инженерные расчеты, начиная от элементарной арифметики и заканчивая сложными реализациями численных методов. Пользователи MathCAD - это студенты, ученые, инженеры, разнообразные технические специалисты. Благодаря простоте применения, наглядности математических действий, обширной библиотеке

встроенных функций и численных методов, возможности символьных вычислений, а также превосходному аппарату представления результатов (графики самых разных типов, мощных средств подготовки печатных документов и Web-страниц), MathCAD стал наиболее популярным математическим приложением. MathCAD 2001, в отличие от большинства других современных математических приложений, построен в соответствии с принципом WYSIWYG ("What You See Is What You Get" - "что Вы видите, то и получите"). Поэтому он очень прост в использовании, в частности, из-за отсутствия необходимости сначала писать программу, реализующую те или иные математические расчеты, а потом запускать ее на исполнение. Вместо этого достаточно просто вводить математические выражения с помощью встроенного редактора формул, причем в виде, максимально приближенном к общепринятому, и тут же получать результат. Кроме того, можно изготовить на принтере печатную копию документа или создать страницу в Интернете именно в том виде, который этот документ имеет на экране компьютера при работе с MathCAD. Создатели MathCAD сделали все возможное чтобы пользователь, не обладающий специальными знаниями в программировании (а таких большинство среди ученых и инженеров), мог в полной мере приобщиться к достижениям современной вычислительной науки и компьютерных технологий. Для эффективной работы с редактором MathCAD достаточно базовых навыков пользователя. С другой стороны, профессиональные программисты (к которым относит себя и автор этих строк) могут извлечь из MathCAD намного больше, создавая различные программные решения, существенно расширяющие возможности, непосредственно заложенные в MathCAD.3 В соответствии с проблемами реальной

жизни, математикам приходится решать одну или несколько из следующих задач:

- ввод на компьютере разнообразных математических выражений (для дальнейших расчетов или создания документов, презентаций, Web-страниц);
- проведение математических расчетов;
- подготовка графиков с результатами расчетов;
- ввод исходных данных и вывод результатов в текстовые файлы или файлы с базами данных в других форматах;
- подготовка отчетов работы в виде печатных документов;
- подготовка Web-страниц и публикация результатов в Интернете;
- получение различной справочной информации из области математики.

Со всеми этими (а также некоторыми другими) задачами с успехом справляется MathCAD:

- математические выражения и текст вводятся с помощью формульного редактора MathCAD, который по возможностям и простоте использования не уступает, к примеру, редактору формул, встроенному в Microsoft Word;
- математические расчеты производятся немедленно, в соответствии с введенными формулами;
- графики различных типов (по выбору пользователя) с богатыми возможностями форматирования вставляются непосредственно в документы;
- возможен ввод и вывод данных в файлы различных форматов;
- документы могут быть распечатаны непосредственно в MathCAD в том виде, который пользователь видит на экране компьютера, или сохранены в формате RTF для последующего редактирования в более

мощных текстовых редакторах (например, Microsoft Word);
- возможно сохранение документов в формате Web-страницы, причем создание файлов с рисунками происходит автоматически;

- символьные вычисления позволяют мгновенно получить разнообразную справочную математическую информацию, а система помощи, Центр Ресурсов и встроенные электронные книги помогают быстро отыскать нужную справку или пример тех или иных расчетов.

Таким образом, следует хорошо представлять себе, что в состав MathCAD входят несколько интегрированных между собой компонентов - это мощный текстовый редактор для ввода и редактирования как текста, так и формул, вычислительный процессор - для проведения расчетов согласно введенным формулам, и символьный процессор, являющийся, по сути, системой искусственного интеллекта. Сочетание этих компонентов создает удобную вычислительную среду для разнообразных математических расчетов, одновременно, документирования результатов работы.

ЗНАКОМСТВО С MATHCAD

В данном разделе, несколько забегаая вперед, покажем, как быстро начать работу с MathCAD, научиться вводить математические выражения и получать первые результаты расчетов. Сведения этого раздела будут затем освещены с большей подробностью, сейчас только продемонстрируем в качестве знакомства со средой MathCAD его некоторые возможности. Поэтому, встретившись с проблемами при повторении на компьютере описываемых здесь действий, например, при вводе выражений или построении графиков, обратитесь к соответствующему материалу следующих разделов.

После того как MathCAD установлен на компьютере

и запущен на исполнение, появляется основное окно приложения, показанное на рис. 1.1. Оно имеет ту же структуру, что и большинство приложений Windows. Сверху вниз располагаются заголовки окна, строка меню, панели инструментов (стандартная и форматирования) и *рабочий лист*, или *рабочая область*, документа (worksheet). Новый документ создается автоматически при запуске MathCAD. В самой нижней части окна находится строка состояния. Не забывая о сходстве редактора MathCAD с обычными текстовыми редакторами, вы интуитивно поймете назначение большинства кнопок на панелях инструментов.

При запуске на переднем плане также появляется диалоговое окно **Tip of the Day** (Совет Дня), которое можно убрать, нажав кнопку **Close** (Заккрыть), нанее указывает курсор на рис. 2.1. Чтобы отключить опцию появления Совета Дня при последующих запусках MathCAD, снимите в его диалоговом окне флажок **Show tips on startup** (Показывать советы при запуске). Нажав кнопку **Next Tip** (Следующий совет), можно просмотреть следующий совет. При запуске MathCAD также можно наблюдать еще одно окно - **Resource Center** (Центр Ресурсов), которое является, по сути, отдельной программой путеводителем по возможностям MathCAD 2001, снабженной множеством примеров решения самых различных математических, физических и инженерных задач. В своей работе вы можете либо не обращать на него внимания, либо пользоваться как хорошим дополнением к справочной системе.

Закрыв окно **Tip of the Day**, можно приступить к вводу выражений в пустой документ. Самый простой (но, возможно, не самый лучший для новичка) способ ввода - это ввод формул с клавиатуры. Для того чтобы выполнить простые расчеты по формулам, сделайте следующее:

- определите место в документе, где должно появиться выражение, щелкнув мышью в соответствующей точке документа;
- введите левую часть выражения;
- введите знак равенства $\leq=>$.

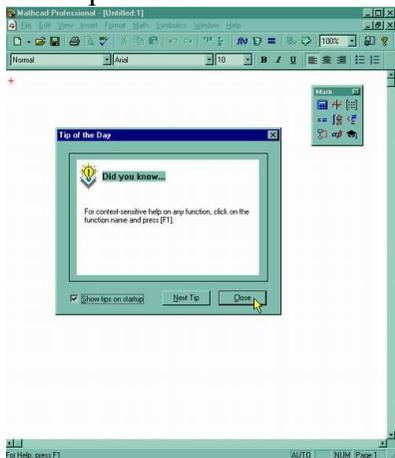


Рис. 2.1. Окно приложения MathCAD 2001 после первого запуска

Оставим пока разговор о более надежных способах ввода, и приведем пример простейших расчетов. Для вычисления синуса какого-нибудь числа достаточно ввести с клавиатуры выражение типа $\sin(1/4)=$. После того как будет нажата клавиша со знаком равенства, с правой стороны выражения, как по мановению волшебной палочки, появится результат (листинг 2.1).

Листинг 2.1. Расчет простого выражения

$$\sin\left(\frac{1}{4}\right) = 0.247$$

Подобным образом можно проводить и более сложные и громоздкие вычисления, пользуясь при этом всем арсеналом специальных функций, которые встроены в

MathCAD. Легче всего вводить их имена с клавиатуры, как в примере с вычислением синуса, но, чтобы избежать возможных ошибок в их написании, лучше выбрать другой путь. Чтобы ввести встроенную функцию в выражение:

1. Определите место в выражении, куда следует вставить функцию.

2. Нажмите кнопку с надписью $f(x)$ на стандартной панели инструментов (на нее указывает курсор на рис. 1.2).

3. В списке **Function Category** (Категория функции) появившегося диалогового окна **Insert Function** (Вставить функцию) выберите категорию, к которой принадлежит функция, - в нашем случае это категория **Trigonometric** (Тригонометрические).

4. В списке **Function Name** (Имя функции) выберите имя встроенной функции, под которым она фигурирует в MathCAD (sin). В случае затруднения с выбором ориентируйтесь на подсказку, появляющуюся при выборе функции в нижнем текстовом поле диалогового окна **Insert Function** (рис.2.2.)

5. Нажмите кнопку ОК - функция появится в документе.

6. Заполните недостающие аргументы введенной функции (в нашем случае это $1/4$).

Результатом будет введение выражения из листинга 2.1, для получения значения которого осталось лишь ввести знак равенства.

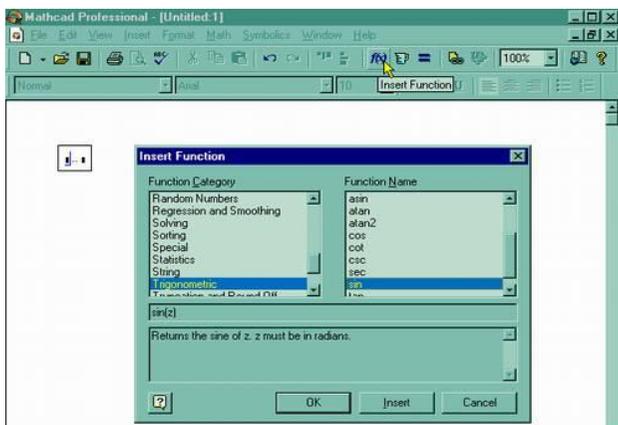


Рис. 2.2. Вставка встроенной функции

Конечно, не всякий символ можно ввести с клавиатуры. Например, неочевидно, как вставить в документ знак интеграла или дифференцирования. Для этого в MathCAD имеются специальные панели инструментов, очень похожие на средства формульного редактора Microsoft Word. Одна из них - панель инструментов **Math** (Математика) - показана на рис. 2.1. Она содержит инструменты для вставки в документы типично математических объектов (операторов, графиков, элементов программ и т. п.). Эта панель показана более крупным планом на рис. 2.3 уже на фоне редактируемого документа.

Панель содержит девять кнопок, нажатие каждой из которых приводит, в свою очередь, к появлению на экране еще одной панели инструментов. С помощью этих девяти дополнительных панелей можно вставлять в документы MathCAD разнообразные объекты. На рис. 1.3, как легко увидеть, на панели **Math** в нажатом состоянии находятся две первые сверху слева кнопки (над левой из них находится указатель мыши). Поэтому на экране присутствуют еще две панели - **Calculator** (Калькулятор) и

Graph (График). Легко догадаться, какие объекты вставляются при нажатии кнопок на этих панелях.

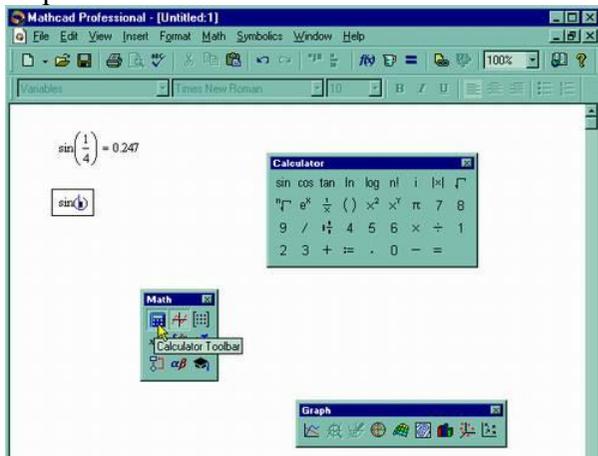


Рис. 2.3. Использование панели инструментов **Math**

К примеру, можно ввести выражение из листинга 2.1 исключительно с помощью панели **Calculator** (Калькулятор). Для этого нужно сначала нажать кнопку **sin** (самую первую сверху). Результат данного действия показан на рис. 2.3 (выражение в рамке). Теперь остается лишь набрать выражение $1/4$ внутри скобок (в *местозаполнителе*, обозначаемом черным прямоугольником). Для этого нажмите последовательно кнопки **1-4** на панели **Calculator** (Калькулятор) и затем, на ней же, кнопку **=**, чтобы получить ответ (разумеется, тот же самый, что и в предыдущей строке документа).

Как видите, вставлять в документы математические символы можно по-разному, как и во многих других приложениях Windows. В зависимости от опыта работы с MathCAD и привычек работы на компьютере, пользователь может выбрать любой из них.

Описанные действия демонстрируют использование MathCAD в качестве обычного калькулятора с

расширенным набором функций. Для математика же интерес представляет, как минимум, возможность задания переменных и операций с функциями пользователя. Нет ничего проще - в MathCAD эти действия, как и большинство других, реализованы по принципу "как принято в математике, так и вводится". Присваивание обозначается не знаком равенства, чтобы подчеркнуть его отличие от операции вычисления. Символ равенства говорит о вычислении значения слева направо, а символ ":= " - о присваивании значения справа налево.

Чтобы построить график функции, например $F(x)$, необходимо ввести параметры функции и их значения, затем ввести функцию $F(x)$ (формулу описания) и следует нажать на панели **Graph** (график) кнопку с нужным типом графика (на нее на рисунке наведен указатель мыши) и в появившейся заготовке графика определить значения, которые будут отложены по осям.

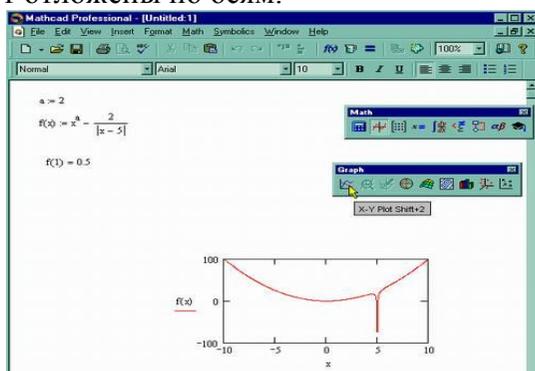


Рис. 2.4. Построение графика функции .

Одной из самых впечатляющих возможностей MathCAD являются символьные вычисления, позволяющие решить многие задачи аналитически. Фактически, по мнению автора, MathCAD "знает" математику, по крайней мере, на уровне неплохого ученого. Умелое использование интеллекта символьного процессора MathCAD избавит вас

от огромного количества рутинных вычислений, к примеру, интегралов и производных. Обратите внимание на традиционную форму написания выражений, единственная особенность заключается в необходимости применения символа символьных вычислений \rightarrow вместо знака равенства. Его, кстати, можно ввести в редакторе MathCAD с любой из панелей **Evaluation** (Выражения) или **Symbolic** (Символика), а символы интегрирования и дифференцирования - с панели **Calculus** (Вычисления).

В этом разделе была рассмотрена лишь небольшая часть вычислительных возможностей системы MathCAD. Тем не менее, несколько приведенных здесь примеров дают неплохое представление о его назначении.

ИНТЕРФЕЙС ПОЛЬЗОВАТЕЛЯ

В MathCAD интерфейс пользователя интуитивен и сходен с другими приложениями Windows. Его составные части:

- верхнее меню, или строка меню (menu bar);
- панели инструментов (toolbars) **Standard** (Стандартная) и **Formatting** (Форматирование);
- панель инструментов **Math** (Математика) и доступные через нее дополнительные математические панели инструментов;

- рабочая область (worksheet);

О строка состояния (status line, или status bar);

- всплывающие, или контекстные, меню (pop-up menus, или context menus);

П диалоговые окна, или диалоги (dialogs).

Большинство команд можно выполнить как с помощью меню (верхнего или контекстного), так и панелей инструментов или клавиатуры.

Меню

Строка меню располагается в самой верхней части

окна MathCAD. Она содержит девять заголовков, щелчок мышью на каждом из которых приводит к появлению соответствующего меню с перечнем сгруппированных по действию команд:

- **File** (Файл) - команды, связанные с созданием, открытием, сохранением, пересылкой по электронной почте и распечаткой на принтере файлов с документами;

- **Edit** (Правка) - команды, относящиеся к правке текста (копирование, вставка, удаление фрагментов и т. п.);

- **View** (Вид) - команды, управляющие внешним видом документа в окне редактора MathCAD, а также команды, создающие файлы анимации;

- **Insert** (Вставка) - команды вставки различных объектов в документы;

- **Format** (Формат) - команды форматирования текста, формул и графиков;

- **Math** (Математика) - команды управления вычислительным процессом;

- **Symbolics** (Символика) - команды символьных вычислений;

- **Window** (Окно) - команды управления расположением окон с различными документами на экране;

- **Help** (Справка) - команды вызова контекстно-зависимой справочной информации, доступа к Центру Ресурсов, опции Совета Дня и сведений о версии программы.

Чтобы выбрать нужную команду, щелкните мышью на содержащем ее меню и повторно на соответствующем элементе меню. Некоторые команды находятся не в самих меню, а в подменю, как это показано на рис. 2.5. Чтобы выполнить такую команду, например команду вызова на экран панели инструментов **Symbolic** (Символика), наведите указатель мыши на пункт **Toolbars View** (Вид) и

выберите в появившемся подменю пункт **Symbolic** (Символика).

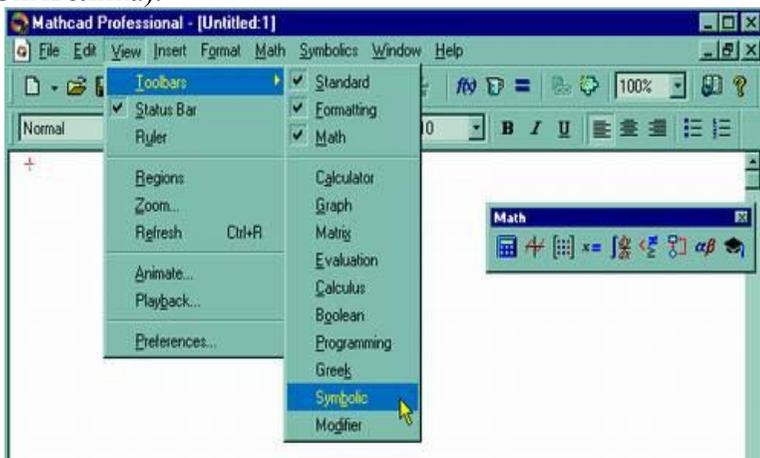


Рис. 2.5. Работа с меню

Обратите внимание, что пункты меню, которые содержат подменю, снабжены стрелками (как пункт **Toolbars** на рис. 2.5). Кроме того, некоторые пункты меню имеют (или не имеют) флажки проверки, указывающие на включение (или выключение) соответствующей опции в текущий момент. Так, на рис. 2.5 флажки проверки выставлены в пунктах **Status Bar** (Строка состояния) и имен трех панелей инструментов, что говорит о наличии в данный момент на экране строки состояния и трех панелей. Флажки же в пунктах **Ruler** (Линейка), **Regions** (Регионы) и имен математических панелей инструментов отсутствуют, т. е. в данный момент эти опции выключены.

Назначение пунктов меню, на которые наведен указатель мыши, появляется в виде подсказки слева на строке состояния (в нижней части окна MathCAD). На рис. 2.5 указатель наведен на пункт **Symbolic**, поэтому подсказка гласит "Show or hide the symbolic keyword toolbar" (Показать или скрыть панель символика). Помимо верхнего меню, схожие функции выполняют всплывающие

меню (рис. 2.6). Они появляются, как и в большинстве других приложений Windows, при нажатии в каком-либо месте документа правой кнопки мыши. При этом состав данных меню зависит от места их вызова, поэтому их еще называют контекстными. MathCAD сам "догадывается", в зависимости от контекста, какие операции могут потребоваться в текущий момент, и помещает в меню соответствующие команды. Поэтому использовать контекстное меню зачастую проще, чем верхнее, т. к. не надо вспоминать, где конкретно в верхнем меню находится нужный пункт. Как и верхнее меню, контекстное также может иметь подменю (на рис. 2.6 показан участок документа с примером изменения отображения знака умножения в формуле; примечательно, что эту операцию в MathCAD можно осуществить только при помощи контекстного меню).

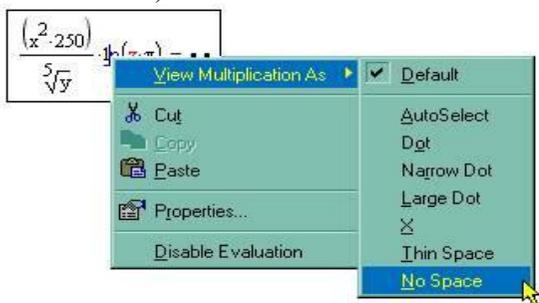


Рис. 2.6. Контекстное меню

ПАНЕЛИ ИНСТРУМЕНТОВ

Панели инструментов служат для быстрого (в один щелчок мыши) выполнения наиболее часто применяемых команд. Все действия, которые можно выполнить с помощью панелей инструментов, доступны и через верхнее меню. На рис. 2.7 изображено окно MathCAD с тремя

основными панелями инструментов, расположенными непосредственно под строкой меню. Кнопки в панелях сгруппированы по сходному действию команд:

- **Standard** (Стандартная) - служит для выполнения большинства операций, таких как действия с файлами, редакторская правка, вставка объектов и доступ к справочным системам;

- **Formatting** (Форматирование) - для форматирования (изменения типа и размера шрифта, выравнивания и т. п.) текста и формул;

- **Math** (Математика) - для вставки математических символов и операторов в документы.

Группы кнопок на панелях инструментов разграничены по смыслу вертикальными линиями - *разделителями*. При наведении указателя мыши на любую из кнопок рядом с кнопкой появляется *всплывающая подсказка* - короткий текст, поясняющий назначение кнопки (см. рис. 2.3 и 2.4). Наряду со всплывающей подсказкой, более развернутое объяснение готовящейся операции можно отыскать на строке состояния.



Рис. 2.7. Основные панели инструментов

Панель **Math** (Математика) предназначена для вызова на экран еще девяти панелей (рис. 2.8), с помощью которых, собственно, и происходит вставка математических операций в документы. В прежних

версиях MathCAD эти математические панели инструментов назывались *палитрами* (palettes) или *наборными панелями*. Чтобы показать какую-либо из них, нужно нажать соответствующую кнопку на панели **Math** (см. рис. 2.3). Перечислим назначение математических панелей:

- **Calculator** (Калькулятор) - служит для вставки основных математических операций, получила свое название из-за схожести набора кнопок с кнопками типичного калькулятора;

- **Graph** (График) - для вставки графиков;

- **Matrix** (Матрица) - для вставки матриц и матричных операторов;

- **Evaluation** (Выражения) - для вставки операторов управления вычислениями;

- **Calculus** (Вычисления) - для вставки операторов интегрирования, дифференцирования, суммирования;

- **Boolean** (Булевы операторы) - для вставки логических (булевых) операторов;

- **Programming** (Программирование) - для программирования средствами MathCAD;

- **Greek** (Греческие символы) - для вставки греческих символов;

- **Symbolic** (Символика) - для вставки символьных операторов.

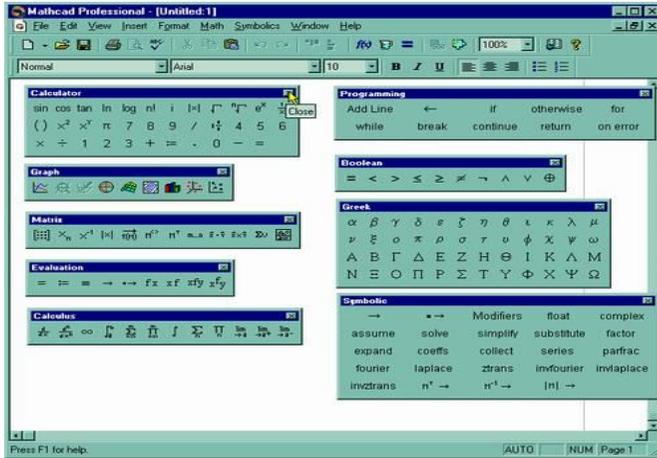


Рис. 2.8. Математические панели инструментов

При наведении указателя мыши на многие из кнопок математических панелей появляется всплывающая подсказка, содержащая еще и сочетание "горячих клавиш", нажатие которых приведет к эквивалентному действию. Ввод действий с клавиатуры часто удобнее нажатия кнопок панелей инструментов, но требует большего опыта.

Настройка панелей инструментов

В MathCAD, подобно другим программам Windows, пользователь может настроить внешний вид панелей инструментов наиболее оптимальным для него образом.

Вы можете:

- показывать или скрывать панели; перемещать панели в любое место экрана и изменять их форму; делать панели плавающими, и наоборот; настраивать основные панели, т. е. определять набор их кнопок.

Присутствие панелей на экране

Вызвать любую панель на экран или скрыть ее можно с помощью меню **View** (Вид) / **Toolbars** (Панели инструментов), выбирая в открывающемся подменю имя нужной панели (см. рис. 2.5). Убрать любую панель с экрана можно еще и посредством контекстного меню,

которое вызывается щелчком правой кнопкой мыши в любом месте панели (например, на любой кнопке). В контекстном меню, следует выбрать пункт **Hide** (Скрыть). Кроме того, если панель *плавающая*, т. е. не прикреплена к основному окну (как, например, все панели на рис. 2.8), то ее можно отключить кнопкой закрытия (на рис. 2.8 указатель мыши наведен на эту кнопку панели **Calculator**). Математические панели, в отличие от основных, можно вызвать или скрыть нажатием соответствующей кнопки панели **Math** (Математика). Присутствие или отсутствие математических панелей показано в виде нажатой (или отжатой) соответствующей кнопки (см. рис. 2.3, 2.4 или 2.8).

Создание плавающих панелей

Чтобы открепить любую из панелей от границ окна MathCAD:

1. Поместите указатель мыши над первым (см. рис. 2.7) или последним разделителем панели (первый разделитель имеет характерный объемный вид, а последний - обычный).

2. Нажмите и удерживайте левую кнопку мыши - вы увидите характерный профиль очертаний панели.

3. Не отпуская кнопку, перетащите панель (для чего переместите указатель мыши в любое место экрана, ориентируясь на перемещение профиля панели).

4. Отпустите кнопку мыши - панель станет плавающей и переместится туда, где находился ее профиль.

Результат перетаскивания основных панелей показан на рис. 2.9. Обратите внимание, что у плавающих панелей инструментов появляется заголовок с названием панели. Чтобы снова прикрепить панель к окну, перетащите ее за этот заголовок к границе окна. При подведении панели на некоторое расстояние к границе

можно увидеть, что панель "притягивается" ею. Следует отпустить в этот момент кнопку мыши, и панель перестанет быть плавающей. Можно прикреплять панели не только к строке меню в верхней части окна, а к любой его границе.

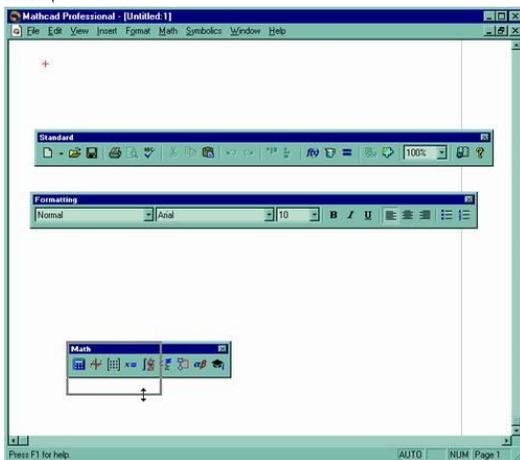


Рис. 2.9. Изменение расположения и размера панелей инструментов

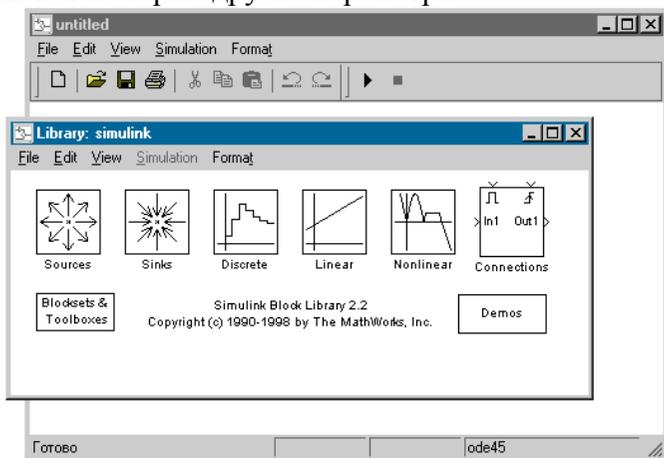
Имеется также и более простой способ открепить панель от границ окна MathCAD. Для этого просто щелкните дважды на ее первом или последнем разделителе. Чтобы прикрепить панель к окну, достаточно двойного щелчка на ее заголовке.

Специализированная система MATLAB Модели Simulink.

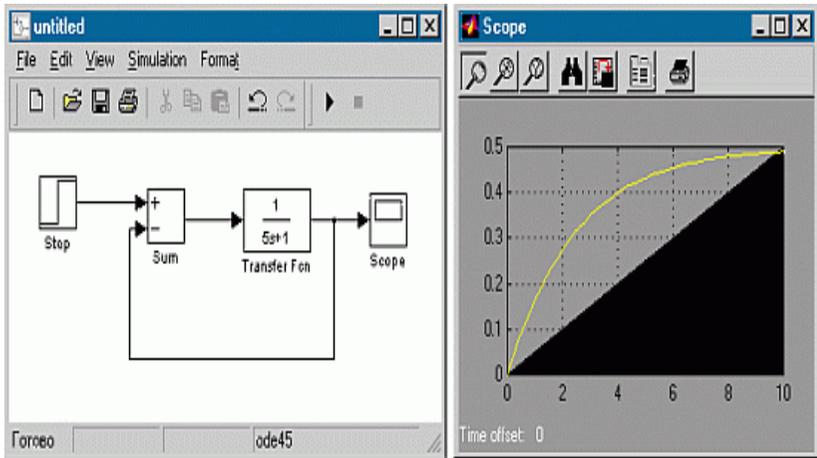
При инсталляции MATLAB можно установить пакет визуального моделирования систем, процессов — Simulink. Чтобы создать новую модель, щёлкните мышкой по пиктограмме  или наберите команду **simulink** . Перед

вами появится два окна. В первом, можно из библиотеки выбрать модель, а во втором, как из кирпичиков, вы собираете общую модель системы:

Двойным щелчком, например по пиктограмме Source (Источники), вы откроете библиотеку источников. Открывая соответствующие библиотеки и перетягиванием мышкой выбранной модели в окно создания моделей системы вы можете набрать модель реальной системы. Далее, запустив модель, вы получите решение. В настройках можно выбрать время работы системы, метод решения и некоторые другие параметры.



Рассмотрим реакцию систему с единичной ООС и аperiodическим звеном в прямой цепи на единичное воздействие. Для её набора нам понадобятся библиотеки модели: Source|Step, Sinks|Scope, Linear|Transfer Fcn и Linear|Sum. Перетянув эти элементарные модели в окно редактирования Simulink, изменяем исходные параметры в соответствии с нашей системой (двойной щелчок мышью) и соединяем их.



Чтобы можно было использовать полученное решение в сценарии MATLAB нужно добавить компонент Connections|Out. Теперь вы имеете возможность работать с переменной. Данные, которые поступают на этот порт доступны вам из среды MATLAB. Чтобы запустить модель, вам достаточно набрать команду **sim** с необходимыми параметрами (наберите HELP SIM для подробной информации).

В последние годы в ВУЗах ряда стран при изучении студентами разных предметов в виде лекционных, лабораторных и практических занятий, при выполнении курсовых работ, проектов, выпускных работ и диссертаций все более широкое применение находит система моделирования и анализа с применением ряда программных комплексов. Среди программных комплексов пакет «Matlab», отличается своей простотой, компактностью, надежностью, расширенными возможностями, отвечающими современным требованиям, базирующимся на новые современных технологиях процессу обучения.

«Matlab» является продуктом фирмы «The Mathwoks Inc» Первая версия пакета «Matlab» была

разработана уже более 20 лет тому назад. Развитие и совершенствование этого пакета происходило одновременно с развитием средств вычислительной техники. Название пакета «Matlab» происходит от словосочетания Matrix Laboratory, он ориентирован, в первую очередь, на обработку массивов данных (матриц и векторов). Именно поэтому, несмотря на достаточно высокую скорость смены поколений вычислительной техники, «Matlab» успевал впитывать все наиболее ценное от каждого из них.

Как следует из названия пакета, он ориентирован, в первую очередь, на обработку массивов данных (матриц и векторов). Это позволило его разработчикам существенно повысить эффективность процедур, работающих с указанными типами данных.

В результате к настоящему времени «Matlab» представляет собой богатейшую библиотеку функций, моделей (более 800). Единственная проблема при работе, с ними заключается в умении быстро отыскать те модели и функции из них, которые нужны для решения поставленной задачи.

Именно в сфере математического моделирования «Matlab» позволяет наиболее полно использовать все современные достижения компьютерных технологий, в том числе средства визуализации и аудификации (озвучивания) данных, а также возможности обмена данными через Интернет. Кроме того, пользователь имеет возможность создавать средствами «Matlab» собственный графический интерфейс, отвечающий как его вкусам, так и требованиям решаемой задачи.

Для облегчения работы с пакетом специалистами различных областей науки и техники вся библиотека функций разбита на разделы. Те модели и функции из них, которые носят более общий характер, входят в состав

основного ядра пакета «Matlab». Те же модели и функции, которые являются специфическими для конкретной области, включены в состав пакетов расширения (Toolboxes).

Таким образом, «Matlab» - в первую очередь, это средство математического моделирования, обеспечивающее проведение исследований с точки зрения анализа и синтеза, практически во всех известных областях техники и науки. При этом структура пакета позволяет эффективно сочетать оба основных подхода к созданию модели: аналитический и имитационный.

SIMULINK

Особое место среди наборов инструментов занимает система визуального моделирования Simulink. В определенном смысле Simulink можно рассматривать как самостоятельный продукт фирмы Math Works (который даже в некоторых случаях продается в «именной» упаковке). Однако он работает только при наличии ядра «Matlab» и использует многие функции, входящие в его состав.

Следует обратить внимание, что пакеты «Matlab», Simulink и их пакеты расширения (Toolboxes, Blocksets) постоянно развиваются и совершенствуются.

Simulink – это уникальный инструмент исследования, не требует от пользователя знания навыков программирования, позволяет приобрести мощнейшие ассоциации и уяснить аналогии между структурой математического описания объекта и структурой блок-схемы, реализующей данную модель.

Simulink – интерактивный инструмент для моделирования, имитации и анализа динамических систем. Он дает возможность строить графические блок-диаграммы, имитировать динамические системы,

исследовать их работоспособность систем и совершенствовать проекты.

Simulink – полностью интегрирован с «Matlab», обеспечивая немедленный доступ к широкому спектру инструментов анализа и проектирования

Сразу становятся ясными причинно-следственные отношения с цепочкой математических понятий: независимая переменная, дифференциальное уравнение, вынуждающая функция, начальные условия, решение дифференциального уравнения и с соответствующей цепочкой понятий: «аналоговое» моделирование – время моделирования, входной сигнал, структурная схема объекта, начальные условия на интеграторах, выходной сигнал и т. д. При моделировании в среде «Matlab», Simulink - средств измерения, регулирования и систем управления у студентов, слушателей и исследователей прививаются:

- навыки выбора, наладки, эксплуатации более современных средств автоматизации;
- навыки использования современных технологий в процессе обучения;
- четкое определение уравнения статики и динамики изучаемого объекта;
- умение анализировать работу изучаемого объекта;
- умение синтезировать изучаемый объект с желаемой характеристикой.

Пакет «Matlab» располагает широким набором виртуальных элементов, модулей, функций, представленных в виде условных обозначений, которые обладают основными свойствами реальных простых физических элементов, а в совокупности простых устройств, модулей, агрегатов, преобразователей, приборов, регуляторов, систем контроля и регулирования и т.д.

Таким образом, «собрав» на экране монитора из соответствующих элементов требуемую виртуальную лабораторную работу, то - есть определенную структурную схему изучаемого объекта, можно выполнить ее полный анализ, изучить его в установившихся и переходных режимах и произвести его синтез с желаемыми характеристиками. При этом можно быть уверенным, что при корректной сборке схемы, (корректном выборе типовых преобразователей, установке необходимых коэффициентов типовых звеньев - элементов, правильном их конфигурировании между собой, с учетом их принципиальной схемы) и умелом проведении экспериментов, результаты исследований совпадут с результатами исследований в реальной схеме, а по точности превзойдет их. В этом - суть программных пакетов «Matlab», «Simulink» и их несомненные превосходства и достоинства.

Пакеты «Matlab», «Simulink» полезны, не только для студентов в процессе изучения различных предметов при всех формах обучения, но и инженерам, научным работникам, которые проектируют и испытывают новейшие устройства, ибо эти испытания можно провести на виртуальной модели. Это избавляет студентов и исследователей от необходимости строить физическую модель изучаемого устройства (выбирать необходимые устройства, их монтировать, заполнять необходимыми веществами, настраивать и эксплуатировать) и, следовательно, резко сокращает как материальные, так и временные затраты.

Пакеты «Matlab», «Simulink» дают возможность более углубленного исследования схем, аппаратов и систем, которые изучаются студентами, исследователями, поскольку обладают, основными широкими типовыми элементарными наборами звеньев, функций и т. д..

Следует также учитывать тенденцию к унификации лабораторных стендов, когда один и тот же стенд позволяет обеспечить выполнение ряда лабораторных работ. В таких стендах элементы и устройства упрятаны вглубь стенда, а на первый план при необходимости выходят структурные схемы анализируемого объекта, его дифференциальное уравнение, переходная функция, весовая функция, передаточная функция или же его мнемосхемы.

Выполнение виртуальных лабораторных работ может оказаться для студентов и исследователей даже более показательным и более ясным, чем выполнение работ на иных современных многоцелевых лабораторных стендах, так как студент – исследователь лабораторный стенд представляет как совокупность элементарных типовых звеньев с определенными связями между собой. Действительно, студент – исследователь, шаг за шагом осмысленно извлекая нужный ему элемент из базы данных, согласно принципиальной схемы изучаемого объекта и понимая происходящие преобразования на каждом простом элементе и представляя себе его уравнение, самостоятельно собирает структурную схему изучаемого объекта на мониторе. Исследовательские возможности пакетов «Matlab», «Simulink» неизмеримо больше, чем у любого современного многоцелевого унифицированного стенда. Наконец, пакеты «Matlab», «Simulink» могут быть полезной при дистанционном обучении.

Контрольные вопросы:

1. Какие средства автоматизации научной работы вы знаете?
2. Для каких процессов предназначен система MathCad?
3. Каким образом осуществляется простые расчеты в

- системе MathCad?
4. С помощью какого инструмента строятся графики функций?
 5. Перечислите возможности системы MatLab?
 6. Система визуального моделирования Simulink.
 7. Преимущества использования пакетов «Matlab» и «Simulink» в учебном процессе?

Лекция №4. Системы автоматизированного проектирования

Содержание:

- 1. Введение в автоматизированное проектирование**
- 2. Стадии проектирования.**
- 3. Классификация моделей и параметров, используемых при автоматизированном проектировании**
- 4. Структура САПР.**
- 5. Разновидности САПР**
- 6. Функции, характеристики и примеры CAE/CAD/CAM-систем.**
- 7. Понятие о CALS-технологии**

Проектирование технического объекта — создание, преобразование и представление в принятой форме образа этого еще не существующего объекта. Образ объекта или его составных частей может создаваться в воображении человека в результате творческого процесса или генерироваться в соответствии с некоторыми алгоритмами в процессе взаимодействия человека и ЭВМ. В любом случае инженерное проектирование начинается при

наличии выраженной потребности общества в некоторых технических объектах, которыми могут быть объекты строительства, промышленные изделия или процессы. Проектирование, при котором все проектные решения или их часть получают путем взаимодействия человека и ЭВМ, называют *автоматизированным*, в отличие от *ручного* (без использования ЭВМ) или *автоматического* (без участия человека на промежуточных этапах). Система, реализующая автоматизированное проектирование, представляет собой *систему автоматизированного проектирования* (в англоязычном написании *CAD System — Computer Aided Design System*).

Проектирование сложных объектов основано на применении идей и принципов, изложенных в ряде теорий и подходов. Наиболее общим подходом является системный подход, идеями которого пронизаны различные методики проектирования сложных систем.

Основной общий принцип системного подхода заключается в рассмотрении частей явления или сложной системы с учетом их взаимодействия. *Системный подход включает в себя выявление структуры системы, типизацию связей, определение атрибутов, анализ влияния внешней среды*

Стадии проектирования

Стадии проектирования — наиболее крупные части проектирования как процесса, развивающегося во времени. В общем случае выделяют стадии научно-исследовательских работ (НИР), эскизного проекта или опытно-конструкторских работ (ОКР), технического, рабочего проектов, испытаний опытных образцов или опытных партий. Стадию НИР иногда называют предпроектными исследованиями или стадией технического предложения. Очевидно, что по мере перехода от стадии к стадии степень подробности и тщательность проработки проекта возрастают, и рабочий проект уже должен быть вполне достаточным для изготовления опытных или серийных образцов. Близким к определению стадии, но менее четко оговоренным понятием, является понятие этапа проектирования.

Стадии (этапы) проектирования подразделяют на составные части, называемые *проектными процедурами*. Примерами проектных процедур могут служить подготовка детализированных чертежей, анализ кинематики, моделирование переходного процесса, оптимизация параметров и другие проектные задачи. В свою очередь, проектные процедуры можно расчленить на более мелкие компоненты, называемые *проектными операциями*, например, при анализе прочности детали сеточными методами операциями могут быть построение сетки, выбор или расчет внешних воздействий, собственно моделирование полей напряжений и деформаций, представление результатов моделирования в графической и текстовой формах. Проектирование сводится к выполнению некоторых последовательностей проектных процедур — *маршрутов проектирования*.

Иногда разработку ТЗ на проектирование называют *внешним* проектированием, а реализацию ТЗ — *внутренним* проектированием.

Классификация моделей и параметров, используемых при автоматизированном проектировании

В автоматизированных проектных процедурах вместо еще не существующего проектируемого объекта оперируют некоторым квазиобъектом — *моделью*, которая отражает некоторые интересующие исследователя свойства объекта. Модель может быть *физическим* объектом (макет, стенд) или *спецификацией*. Среди моделей-спецификаций различают упомянутые выше функциональные, поведенческие, информационные, структурные модели (описания). Эти модели называют *математическими*, если они формализованы средствами аппарата и языка математики.

Математическая функциональная модель в общем случае представляет собой алгоритм вычисления вектора выходных параметров Y при заданных векторах параметров элементов X и внешних параметров Q .

Математические модели могут быть

символическими и численными. При использовании *символических* моделей оперируют не значениями величин, а их символическими обозначениями (идентификаторами). *Численные* модели могут быть *аналитическими*, т. е. их можно представить в виде явно выраженных зависимостей выходных параметров Y от параметров внутренних X и внешних Q , или *алгоритмическими*, в которых связь Y , X и Q задана неявно в виде алгоритма моделирования. Важнейший частный случай алгоритмических моделей — *имитационные*, они отображают процессы в системе при наличии внешних воздействий на систему. Другими словами, имитационная модель — это алгоритмическая • поведенческая модель.

Классификацию математических моделей выполняют также по ряду других признаков. Так, в зависимости от принадлежности к тому или иному иерархическому уровню выделяют модели уровней системного, функционально-логического, макроуровня (сосредоточенного) и микроуровня (распределенного).

По характеру используемого для описания математического аппарата различают модели лингвистические, теоретико-множественные, абстрактно-алгебраические, нечеткие, автоматные и т. п.

Кроме того, введены понятия полных моделей и макромоделей, моделей статических и динамических, детерминированных и стохастических, аналоговых и дискретных.

Полная модель объекта в отличие от *макромодели* описывает не только процессы на внешних выводах моделируемого объекта, но и внутренние для объекта процессы.

Статические модели описывают статические состояния, в них не присутствует время в качестве независимой переменной. *Динамические* модели отражают поведение системы, т. е. в них обязательно используется время.

Стохастические и *детерминированные* модели различают в зависимости от учета или неучета случайных факторов.

В *аналоговых* моделях фазовые переменные —

непрерывные величины, в *дискретных* — дискретные, в частном случае дискретные модели являются *логическими (булевыми)*, в них состояние системы и ее элементов описывается булевыми величинами. В ряде случаев полезно применение *смешанных* моделей, в которых одна часть подсистем характеризуется аналоговыми моделями, другая — логическими.

Информационные модели относятся к информационной страте автоматизированных систем, их используют для описания связей между единицами информации и прежде всего при инфологическом проектировании баз данных (БД)

Структура САПР

Как и любая сложная система, САПР состоит из подсистем (рис. 4.1). Различают подсистемы проектирующие и обслуживающие.

Проектирующие подсистемы непосредственно выполняют проектные процедуры. Примерами проектирующих подсистем могут служить подсистемы геометрического трехмерного моделирования механических объектов, изготовления конструкторской документации, схемотехнического анализа, трассировки соединений в печатных платах.

Обслуживающие подсистемы обеспечивают функционирование проектирующих подсистем, их совокупность часто называют системной средой (или оболочкой) САПР. Типичными обслуживающими подсистемами являются подсистемы управления проектными данными (PDM — Product Data Management), управления процессом проектирования (DesPM — Design Process Management), пользовательского интерфейса для связи раз-работников с ЭВМ, CASE (Computer Aided Software Engineering) для разработки и сопровождения программного обеспечения САПР, обучающие подсистемы для освоения пользователями технологий, реализованных в САПР.



Рис 4.1. Структура программного обеспечения САПР

Структурирование САПР по различным аспектам обуславливает появление *видов обеспечения САПР*. Принято выделять семь видов обеспечения-

- *техническое* (ТО), включающее различные аппаратные средства (ЭВМ, периферийные устройства, сетевое коммутационное оборудование, линии связи, измерительные средства);
- *математическое* (МО), объединяющее математические методы, модели и алгоритмы для выполнения проектирования;
- *программное* (ПО), представляемое компьютерными программами САПР;
- *информационное* (ИО), состоящее из баз данных (БД), систем управления базами данных (СУБД), а также других данных, используемых при проектировании; отметим, что вся совокупность используемых при проектировании данных называется информационным фондом САПР, а БД вместе с СУБД носит название банка данных (БД),
- *лингвистическое* (ЛО), выражаемое языками общения между проектировщиками и ЭВМ, языками программирования и языками обмена данными между техническими средствами

САПР;

- *методическое* (МО), включающее различные методики проектирования, иногда к МО относят также математическое обеспечение;
- *организационное* (ОО), представляемое штатными расписаниями, должностными инструкциями и другими документами, регламентирующими работу проектного предприятия.

Разновидности САПР

Классификацию САПР осуществляют по ряду признаков, например, по приложению, целевому назначению, масштабам (комплексности решаемых задач), характеру базовой подсистемы — ядра САПР.

По *приложениям* наиболее представительными и широко используемыми являются следующие группы САПР.

1. САПР для применения в отраслях общего машиностроения. Их часто называют машиностроительными САПР или MCAD (Mechanical CAD) системами.

2. САПР для радиоэлектроники. Их названия — ECAD (Electronic CAD) или EDA (Electronic Design Automation) системы.

3. САПР в области архитектуры и строительства.

По *целевому назначению* различают САПР или подсистемы САПР, обеспечивающие разные аспекты (страты) проектирования. Так, в составе MCAD появляются CAE/CAD/CAM-системы:

1) САПР функционального проектирования, иначе САПР-Ф или CAE (Computer Aided Engineering) системы;

2) *конструкторские* САПР общего машиностроения — САПР-К, часто называемые просто CAD-системами;

3) *технологические* САПР общего машиностроения — САПР-Т, иначе называемые автоматизированными системами технологической подготовки производства АСТПП или системами САМ (Computer Aided Manufacturing)

По *масштабам* различают отдельные программно-методические комплексы (ПМК) САПР, например, комплекс анализа прочности механических изделий в

соответствии с методом конечных элементов (МКЭ) или комплекс анализа электронных схем, системы ПМК, системы с уникальными архитектурами не только программного (software), но и технического (hardware) обеспечений

По *характеру базовой подсистемы* различают следующие разновидности САПР

1 САПР на базе подсистемы машинной графики и геометрического моделирования Эти САПР ориентированы на приложения, где основной процедурой проектирования является конструирование, т е определение пространственных форм и взаимного расположения объектов Поэтому к этой группе систем относится большинство графических ядер САПР в области машиностроения

2 САПР на базе СУБД Они ориентированы на приложения, в которых при сравнительно несложных математических расчетах перерабатывается большой объем данных Такие САПР преимущественно встречаются в технико-экономических приложениях, например, при проектировании бизнес-планов, но имеют место также при проектировании объектов, подобных щитам управления систем автоматики САПР на базе конкретного прикладного пакета Фактически это автономно используемые программно-методические комплексы, например, имитационного моделирования производственных процессов, расчета прочности по методу конечных элементов, синтеза и анализа систем автоматического управления и т п Часто такие САПР относятся к системам САЕ Примерами могут служить программы логического проектирования на базе языка VHDL, математические пакеты типа MathCAD

3 Комплексные (интегрированные) САПР, состоящие из совокупности подсистем предыдущих видов Характерными примерами комплексных САПР являются САЕ/CAD/CAM-системы в машиностроении или САПР БИС Так, САПР БИС включает в себя СУБД и подсистемы проектирования компонентов, принципиальных, логических и функциональных схем, топологии кристаллов, тестов для проверки годности

изделий Для управления столь сложными системами применяют специализированные *системные среды*

Функции, характеристики и примеры САЕ/CAD/CAM-систем

Функции САD-систем в машиностроении подразделяют на функции двухмерного (2D) и трехмерного (3D) проектирования К функциям 2D относятся черчение, оформление конструкторской документации, к функциям 3D — получение трехмерных моделей, метрические расчеты, реалистичная визуализация, взаимное преобразование 2D и 3D моделей

Среди САD-систем различают «легкие» и «тяжелые» системы Первые из них ориентированы преимущественно на 2D графику, сравнительно дешевы и менее требовательны в отношении вычислительных ресурсов Вторые ориентированы на геометрическое моделирование (3D), более универсальны, дороги, оформление чертежной документации в них обычно осуществляется с помощью предварительной разработки трехмерных геометрических моделей

Основные функции САМ-систем разработка технологических процессов, синтез управляющих программ для технологического оборудования с числовым программным управлением (ЧПУ), моделирование процессов обработки, в том числе построение траекторий относительного движения инструмента и заготовки в процессе обработки, генерация постпроцессоров для конкретных типов оборудования с ЧПУ (NC — Numerical Control), расчет норм времени обработки

Функции САЕ-систем довольно разнообразны, так как связаны с проектными процедурами анализа, моделирования, оптимизации проектных решений. В состав машиностроительных САЕ-систем прежде **всего** включают программы для следующих процедур-

- моделирование полей физических величин, в том числе анализ прочности, который чаще всего выполняется в соответствии с МКЭ,
- расчет состояний и переходных процессов на

макроуровне,

- имитационное моделирование сложных производственных систем на основе моделей массового обслуживания и сетей Петри.

Для удобства адаптации САПР к нуждам конкретных приложений, для ее развития целесообразно иметь в составе САПР инструментальные средства адаптации и развития. Эти средства представлены той или иной CASE-технологией, включая языки расширения. В некоторых САПР применяют оригинальные инструментальные среды.

Понятие о CALS-технологии

CALS-технология — это технология комплексной компьютеризации сфер промышленного производства, цель которой — унификация и стандартизация спецификаций промышленной продукции на всех этапах ее жизненного цикла. Основные спецификации представлены проектной, технологической, производственной, маркетинговой, эксплуатационной документацией. В CALS-системах предусмотрены хранение, обработка и передача информации в компьютерных средах, оперативный доступ к данным в нужное время и в нужном месте. Соответствующие системы автоматизации назвали автоматизированными логистическими системами или CALS (Computer Aided Logistic Systems). Поскольку под логистикой обычно понимают дисциплину, посвященную вопросам снабжения и управления запасами, а функции CALS намного шире и связаны со всеми этапами жизненного цикла промышленных изделий, применяют и более соответствующую предмету расшифровку аббревиатуры CALS — Computer Acquisition and LifeCycle Support.

Применение CALS позволяет существенно сократить объемы проектных работ, так как описания многих составных частей оборудования, машин и систем, проектировавшихся ранее, хранятся в базах данных сетевых серверов, доступных любому пользователю технологии CALS. Существенно облегчается решение проблем ремонтпригодности, интеграции продукции в различного рода системы и среды, адаптации к меняющимся условиям эксплуатации, специализации проектных организаций и т. п. Ожидается, что успех на рынке сложной технической

продукции будет немислим вне технологии CALS

Развитие CALS-технологии должно привести к появлению так называемых *виртуальных производств*, при которых процесс создания спецификаций с информацией для программно управляемого технологического оборудования, достаточной для изготовления изделия, может быть распределен во времени и пространстве между многими организационно автономными проектными студиями. Среди несомненных достижений CALS-технологии следует отметить легкость распространения передовых проектных решений, возможность многократного воспроизведения частей проекта в новых разработках и др.

Построение открытых распределенных автоматизированных систем для проектирования и управления в промышленности составляет основу современной CALS-технологии. Главная проблема их построения — обеспечение единообразного описания и интерпретации данных независимо от места и времени их получения в общей системе, имеющей масштабы вплоть до глобальных. Структура проектной, технологической и эксплуатационной документации, языки ее представления должны быть стандартизованными. Тогда становится реальной успешная работа над общим проектом разных коллективов, разделенных во времени и пространстве и использующих разные CAE/CAD/CAM-системы. Одна и та же конструкторская документация может быть использована многократно в разных проектах, а одна и та же технологическая документация адаптирована к разным производственным условиям, что позволяет существенно сократить и удешевить общий цикл проектирования и производства. Кроме того, упрощается эксплуатация систем.

Следовательно, информационная интеграция является неотъемлемым свойством CALS-систем. Поэтому в основу CALS-технологии положен ряд стандартов, обеспечивающих такую интеграцию.

Важные проблемы, требующие решения при создании комплексных САПР — управление сложностью проектов и интеграция ПО. Эти проблемы включают вопросы декомпозиции проектов, распараллеливания

проектных работ, целостности данных, межпрограммных интерфейсов и др.

Комплексные автоматизированные системы

Известно, что частичная автоматизация зачастую не дает ожидаемого повышения эффективности функционирования предприятий. Поэтому предпочтительным является внедрение интегрированных САПР, автоматизирующих все основные этапы проектирования изделий. Дальнейшее повышение эффективности производства и конкурентоспособности выпускаемой продукции возможно за счет интеграции систем проектирования, управления и документооборота.

Такая интеграция лежит в основе создания *комплексных систем автоматизации*, в которых помимо функций собственно САПР реализуются средства для автоматизации функций управления проектированием, документооборота, планирования производства, учета и т. п.

В основу CALS-технологии положен ряд стандартов и прежде всего это стандарты STEP, а также Parts Library, Mandate, SGML (Standard Generalized Markup Language), EDIFACT (Electronic Data Interchange For Administration, Commerce, Transport) и др. Стандарт SGML устанавливает способы унифицированного оформления документов определенного назначения — отчетов, каталогов, бюллетеней и т. п., а стандарт EDIFACT — способы обмена подобными документами.

Одна из наиболее известных реализаций CALS-технологии разработана фирмой Cot-putervision. Эта технология названа EPD (Electronic Product Definition) и ориентирована на поддержку процессов проектирования и эксплуатации изделий машиностроения.

В CALS-системах на всех этапах жизненного цикла изделий используется документация, полученная на этапе проектирования. Поэтому естественно, что составы подсистем в CALS и комплексных САПР в значительной мере совпадают.

Технологию EPD реализуют

CAD — система автоматизированного проектирования;

CAM — автоматизированная система технологической

подготовки производства (АСТПП).

CAE — система моделирования и расчетов,

CAPE (Concurrent Art-to-Product Environment) — система поддержки параллельного проектирования (concurrent engineering),

PDM — система управления проектными данными, представляющая собой специализированную СУБД (DBMS — Data Base Management System),

3D Viewer — система трехмерной визуализации,

CADD — система документирования,

CASE — система разработки и сопровождения программного обеспечения, методики исследования и анализа функционирования предприятий

Основу EPD составляют системы CAD и PDM, в качестве которых используются CADD5 и Ortegra соответственно

Управление данными в САПР

В большинстве автоматизированных информационных систем применяют СУБД, поддерживающие реляционные модели данных. Среди общих требований к СУБД можно отметить: 1) обеспечение целостности данных (их полноты и достоверности); 2) защита данных от несанкционированного доступа и от искажений из-за сбоев аппаратуры; 3) удобство пользовательского интерфейса; 4) в большинстве случаев важна возможность распределенной обработки в сетях ЭВМ.

Первые два требования обеспечиваются ограничением прав доступа, запрещением одновременного использования одних и тех же обрабатываемых данных (при возможности их модификации), введением контрольных точек (checkpoints) для защиты от сбоев и т. п.

Банк данных в САПР является важной обслуживающей подсистемой, он выполняет функции информационного обеспечения и имеет ряд особенностей. В нем хранятся как редко изменяемые данные (архивы, справочные данные, типовые проектные решения), так и сведения о текущем состоянии различных версий выполняемых проектов. Как правило, БД работает в многопользовательском режиме, с его помощью осуществляется информационный интерфейс (взаимодействие) различных подсистем САПР. Построение БД САПР — сложная задача, что обусловлено следующими особенностями САПР.

1. Разнообразие проектных данных, фигурирующих в процессах обмена как по своей семантике (многоаспектность), так и по формам представления. В частности, значительна доля графических данных.

2. Нередко обмены должны производиться с высокой частотой, что предъявляет жесткие требования к быстродействию средств обмена (полагают, что СУБД должна работать со скоростью обработки тысяч сущностей в секунду).

3. В САПР проблема целостности данных оказывается более трудной для решения, чем в большинстве других систем, поскольку проектирование является процессом взаимодействия многих проектировщиков, которые не только считают данные, но и изменяют их, причем в значительной мере работают параллельно. Из этого факта вытекают следствия во-первых, итерационный характер проектирования обычно приводит к наличию по каждой части проекта нескольких версий, любая из них может быть принята в дальнейшем в качестве основной, поэтому нужно хранить все версии с возможностью возврата к любой из них, во-вторых, нельзя допускать использования неутвержденных данных, поэтому проектировщики должны иметь свое рабочее пространство в памяти и работать в нем автономно, а моменты внесения изменений в общую БД должны быть согласованными и не порождать для других пользователей неопределенности данных

4. Транзакции могут быть длительными и трудоемкими. Транзакцией называют последовательность операций по удовлетворению запроса. В САПР внесение изменений в некоторую часть проекта может вызвать довольно длинную и разветвленную сеть изменений в других его частях из-за существенной взаимозависимости компонентов проекта (многошаговость реализации запросов). В частности, транзакции могут включать в себя такие трудоемкие операции, как верификация проектного решения с помощью математического моделирования. В результате транзакции могут длиться даже несколько часов и более. Одна из трудностей заключается в отображении взаимозависимости (ассоциативности) данных. При хранении компонентов проекта во внешней памяти затраты

времени на обработку запросов оказываются значительно выше, чем в большинстве других автоматизированных систем, с менее выраженными взаимозависимостями данных.

5. Иерархическая структура проектных данных и, следовательно, отражение наследования в целях сокращения объема базы данных.

В определенной мере названные особенности учитываются в СУБД третьего поколения, в которых стали применяться черты объектно-ориентированных (объектных) СУБД. В них наборы данных характеризующих состояние предметной области (состояние проекта в случае САПР) помещаются в отдельные файлы. Интерпретация семантики данных осуществляется с помощью специальных процедур (методов) сопровождающих наборы.

Отличительные особенности СУБД третьего поколения расширенный набор возможных типов данных (это абстрактные типы, массивы, множества, записи, композиции разных типов, отображение величин с значениями разных типов), открытость (доступность из разных языков программирования возможность обращения к прикладным системам из СУБД), непроцедурность языка (общепринятым становится язык запросов SQL), управление асинхронными параллельными процессами, состояние которых отражает БД. Последнее свойство позволяет говорить о тесной взаимосвязи СУБД и подсистемы управления проектами DesPM.

Программные средства управления проектированием в САПР

В зависимости от степени автоматизации управляющих функций можно выделить несколько уровней управления проектированием:

1) компонентный; на этом уровне пользователь должен знать специфические особенности каждой конкретной программы, используемой в маршруте проектирования; при организации маршрута он должен позаботиться об информационных интерфейсах

используемых программ; другими словами, системная среда лишь предоставляет сведения об имеющихся программах и их интерфейсах;

2) ресурсный; пользователь по-прежнему оперирует программами при компиляции маршрута проектирования, но системная среда позволяет скрыть специфику каждой программы, так как общение унифицировано;

3) заданный; пользователь составляет маршрут проектирования не из отдельных программ, а из отдельных проектных процедур; покрытие маршрута программами выполняет системная среда;

4) проблемный; пользователь формулирует задания в форме «что нужно сделать», а не «как это сделать», т. е. не определяет маршрут проектирования, а ставит проектную проблему.

В системных средах САПР управление проектированием возлагается на подсистему CAPE, в некоторых системах обозначаемую как DesPM (Design Process Manager). DesPM должна включать в себя компоненты: комплексы базовых знаний по тем предметным областям, которые определяются объектом проектирования, а также знаний о языках представления характеристик и ограничений; средства для генерации плана (маршрута проектирования), определения наличия средств и ресурсов для реализации плана, средства выполнения плана; средства оценки результатов DesPM позволяет выбирать объекты проектирования, проводить декомпозицию моделей, для каждого компонента выбирать проектные процедуры из имеющегося набора.

По каждому объекту DesPM выдает сообщения, примерами которых могут быть: «объект проектируется другим разработчиком», «проектирование преждевременно, не выполнены предшествующие процедуры», «не подготовлены исходные данные». Одной из важнейших функций DesPM является помощь в реализации параллельного проектирования. Желательно в DesPM предусмотреть возможности создания «суперпроцедур» — командных файлов для выполнения часто повторяющихся фрагментов маршрутов проектирования.

Расширение возможностей управления

проектированием и адаптация системной среды к конкретным САПР связано с применением языков расширения. Язык расширения — это язык программирования, позволяющий адаптировать и настраивать системную среду САПР на выполнение новых проектов. Язык расширения должен обеспечивать доступ к различным компонентам системной среды, объединять возможности базового языка программирования и командного языка, включать средства процедурного программирования. Для большинства языков расширения базовыми являются ЛИСП или С.

Так, язык Skill из Design Framework-2 фирмы Cadence или язык CCL (CASE Comment Language) фирмы Matra Datavision являются ЛИСП-подобными, а язык AMPLE из Falcon Framework фирмы Mentor Graphics базируется на языках С и ПАСКАЛЬ.

Управление процессом проектирования включает в себя большое число действий и условий, поддерживающих параллельную работу многих пользователей над общим проектом. Управление выполняется на основе моделей вычислительных процессов. Используются спецификации моделей, принятые в CASE-системах, например, диаграммы потоков данных, ориентированные графы. Сначала модели составляют для заданного уровня, а затем система осуществляет их покрытие. Применяют также описания на языках расширения или 4GL. В системной среде Ulyses спецификации даны в виде набора модулей с указанием условий их активизации, что близко к представлению моделей в системах, управляемых знаниями.

Инструментальные среды разработки программного обеспечения.

CASE-системы часто отождествляют с инструментальными средами разработки ПО, называемыми средами быстрой разработки приложений (RAD — Rapid Application Development). Примерами широко известных инструментальных сред RAD являются VB (Visual Basic), Delphi, PowerBuilder фирм Microsoft, Borland, PowerSoft соответственно. Применение инструментальных сред существенно сокращает объем

ручной работы программистов особенно при проектировании интерактивных частей программ.

Большое практическое значение имеют инструментальные среды для разработки ПП, предназначенных для работы под управлением операционных систем Windows, в связи с широкой распространенностью последних

Простейшая система для написания Windows-программ на языке C++, позволяющая сократить объем кода, создаваемого пользователем вручную, основана на библиотеке DLL (Dynamic Link Library), которая содержит модули, реализующие функции API (Application Programming Interface) для связи прикладных программ с ОС MS Windows.

Эта система получила развитие в MFC (Microsoft Foundation Classes) представляющей собой библиотеку классов для автоматического создания каркасов ПО многоуровневых приложений. В библиотеке имеются средства поддержки оконного интерфейса работы с файлами и др.

В средах быстрой разработки приложения RAD обычно реализуется способ программирования, называемый управлением событиями. При этом достигается автоматическое создание каркасов программ, существенно сокращается объем ручного кодирования. В этих средах пользователь может работать одновременно с несколькими экранами (окнами). Типичными являются окна из следующего списка:

1. Окно меню с пунктами «File», «Edit», «Window» и т.п.

2. Окно формы, на котором собственно и создается прототип экрана будущей прикладной программы.

3. Палитра инструментов – набор изображений объектов пользовательского интерфейса, из которых можно компоновать содержимое окна формы.

4. Окно свойств и событий, с помощью которого ставятся в соответствие друг другу объекты окна формы события и обработчики событий. Событием в прикладной программе является нажатие клавиши или установка курсора мыши в объект формы. Каждому событию должна соответствовать событийная процедура (обработчик события), которая проверяет код клавиши и вызывает

нужную реакцию. В RAD имеются средства для удобства разработки обработчиков событий.

5. Окно редактора кода, в котором пользователь записывает создаваемую вручную часть кода.

6. Окно проекта — список модулей и форм в создаваемой программе.

Для написания событийных процедур в Visual Basic используется язык и текстовый редактор одноименного языка, в Delphi — язык и редактор языка Object Pascal. В CASE-системе фирмы IBM, включающей части visualage (для клиентских приложений) и visualgen (для серверных приложений), базовым языком выбран smalltalk. В среде разработки приложений клиент-сервер sqlwindows оригинальные фрагменты программ пишутся на специальном языке SAL. Нужно заметить, что для реализации вычислительных процедур и, в частности, для написания миниспецификаций используется обычная для 3GL технология программирования.

Обычно после написания ПП на базовом языке компилятор системы переводит программ на промежуточный р-код. Вместе с интерпретатором р-кода эта программа рассматривается, как EXE-файл.

Инструментальные средства концептуального проектирования CASE-системы

В современных информационных технологиях важное место отводится инструментальным средствам и средам разработки АС, в частности, системам разработки и сопровождения их ПО. Эти технологии и среды образуют системы, называемые CASE-системами.

Используется двойное толкование аббревиатуры CASE, соответствующее двум направлениям использования CASE-систем. Первое из них — Computer Aided System Engineering — подчеркивает направленность на поддержку концептуального проектирования сложных систем, преимущественно слабоструктурированных. Далее CASE-системы этого направления будем называть системами CASE для концептуального проектирования.

Второе направление было рассмотрено выше, его название Computer Aided Software engineering переводится, как автоматизированное проектирование программного обеспечения, соответствующие CASE-системы называют инструментальными CASE или инструментальными средами разработки ПО (одно из близких к этому названий — RAD — Rapid Application Development).

Среди систем CASE для концептуального проектирования различают системы функционального, информационного или поведенческого проектирования. Наиболее известной методикой функционального проектирования сложных систем является методика SADT (Structured Analysis and Design Technique), предложенная в 1973 г. Р. Россом и впоследствии ставшая основой международного стандарта IDEF0 (Integrated DEFinition 0).

Системы информационного проектирования реализуют методики инфологического проектирования БД. Широко используются язык и методика создания информационных моделей приложений, закрепленные в международном стандарте IDEF1X. Кроме того, развитые коммерческие СУБД, как правило, имеют в своем составе совокупность CASE-средств проектирования приложений.

Основные положения стандартов IDEF0 и IDEF1X использованы также при создании комплекса стандартов ISO 10303, лежащих в основе технологии STEP для представления в компьютерных средах информации, относящейся к проектированию и производству в промышленности.

Поведенческое моделирование сложных систем используют для определения динамики функционирования сложных систем. В его основе лежат модели и методы имитационного моделирования систем массового обслуживания, сети Петри, возможно применение конечно-автоматных моделей, описывающих поведение системы,

как последовательности смены состояний.

Применение инструментальных CASE-систем ведет к сокращению затрат на разработку ПО благодаря уменьшению числа итераций и числа ошибок, к улучшению качества продукта в результате лучшего взаимопонимания разработчика и заказчика, к облегчению сопровождения готового ПО.

Среди инструментальных CASE-систем различают интегрированные комплексы инструментальных средств для автоматизации всех этапов жизненного цикла ПО (такие системы называют Workbench) и специализированные инструментальные средства для выполнения отдельных функций (Tools). Средства CASE по своему функциональному назначению принадлежат к одной из следующих групп: 1) средства программирования, 2) средства управления программным проектом; 3) средства верификации (анализа) программ; 4) средства документирования.

К первой группе относятся компиляторы с алгоритмических языков; построители диаграмм потоков данных; планировщики для построения высокоуровневых спецификаций и планов ПО (возможно на основе баз знаний, реализованных в экспертных системах); интерпретаторы языков спецификации и языков четвертого поколения; прототайпер для разработки внешних интерфейсов — экранов, форм выходных документов, сценариев диалога; генераторы программ определенных классов (например, конверторы заданных языков, драйверы устройств программного управления, постпроцессоры); кросс-средства; отладчики программ. При этом под языками спецификаций понимают средства укрупненного описания разрабатываемых алгоритмов и программ, к языкам 4GL относят языки для компиляции программ из набора готовых модулей, реализующих

типовые функции достаточно общих приложений (чаще всего это функции технико-экономических систем).

Основной средств управления программным проектом является репозиторий — БД проекта. Именно в репозитории отражена история развития программного проекта, содержатся все созданные версии (исходный программный код, исполняемые программы, библиотеки, сопроводительная документация и т. п.) с помощью репозитория осуществляется контроль и отслеживание вносимых изменений.

Средства верификации служат для оценки эффективности исполнения разрабатываемых программ и определения наличия в них ошибок и противоречий. Различают статические и динамические анализаторы. В статических анализаторах ПО исследуется на наличие неопределенных данных, бесконечных циклов, недопустимых передач управления и т. п. Динамический анализатор функционирует в процессе исполнения проверяемой программы; при этом исследуются трассы, измеряются частоты обращений к модулям и т. п. Используемый математический аппарат — сети Петри, теория массового обслуживания.

В последнюю из перечисленных групп входят документаторы для оформления программной документации, например, отчетов по данным репозитория; различные редакторы для объединения, разделения, замены, поиска фрагментов программы и других операций редактирования.

Контрольные вопросы:

- 1.Какие системы называются автоматизированными системами проектирования?
- 2.Классификация моделей автоматизированного

- проектирования.
3. Основные признаки классификации САПР.
 4. Понятие о CALS-технологии
 5. Программные средства управления проектированием в САПР.

Лекция № 5.

Компьютерные сети. Техническое обеспечение сетей. Локальная и глобальная сеть

Содержание:

- 1. Компьютерные сети – основные понятия.**
- 2. Локальные сети. Топология локальных сетей.**
- 3. Техническое и программное обеспечение локальных сетей.**

В современном мире ни одну крупную технологическую проблему нельзя решить без переработки значительных объемов информации и коммуникационных процессов. Наряду с энерго и фондовооруженностью современному производству необходима и информационная вооруженность, определяющая степень применения прогрессивных технологий. Особое место в организации новых информационных технологий занимают компьютерные сети.

Компьютерные (информационно-вычислительные) сети представляют собой совокупность компьютеров, объединенных посредством телефонной сети, а также специализированными сетями передачи данных. Передача информации на значительные расстояния осуществляется с помощью проводных, кабельных, радиорелейных и

спутниковых линий связи. В настоящее время широкое применение находит оптическая связь по стекловолоконным кабелям. В общем случае, для создания компьютерных сетей необходимо специальное аппаратное обеспечение (сетевое оборудование) и специальное программное обеспечение (сетевые программные средства). Простейшее соединение двух компьютеров для обмена данными называется прямым соединением. Для создания прямого соединения компьютеров, работающих в ОС Windows(начиная с ОС Windows-98), не требуется ни специального аппаратного, ни программного обеспечения. В этом случае аппаратными средствами являются стандартные порты ввода/вывода, а в качестве программного обеспечения используется стандартное средство, имеющееся в составе операционной системы (**Пуск► Программы► Стандартные► Связь► Прямое кабельное соединение**).

Основной задачей, решаемой при создании компьютерных сетей, является обеспечение совместимости оборудования по электрическим и механическим характеристикам и обеспечение совместимости информационного обеспечения (программ и данных) по системе кодирования и формату данных. Решение этой задачи относится к области стандартизации и основано на так называемой модели OSI (Model of Open System Interconnections –модель взаимодействия открытых систем). Она создана на основе технических предложений Международного института стандартов ISO (International Standards Organization).

Согласно модели ISO/OSI архитектуру компьютерных сетей следует рассматривать на разных уровнях (общее число уровней – до семи).

Для обеспечения необходимой совместимости на каждом из семи возможных уровней архитектуры

компьютерной сети действуют специальные стандарты, называемые протоколами. Они определяют характер аппаратного взаимодействия компьютеров сети (аппаратные протоколы) и характер взаимодействия программ и данных (программные протоколы). Физически функции поддержки протоколов исполняют аппаратные устройства (интерфейсы) и программные средства (программы поддержки протоколов). Программы, выполняющие поддержку протоколов, также называют протоколами.

В соответствии с используемыми протоколами компьютерные сети принято разделять на локальные (LAN – Local Area Network) и глобальные (WAN – Wide Area Network).

Локальные сети – это сети, объединяющие компьютеры, находящиеся рядом (в одной комнате, на одном этаже, здании, в нескольких близко расположенных зданиях). При этом для соединения компьютеров используются выделенные линии связи, принадлежащие той же организации, что и компьютеры. Глобальные сети, как правило, характеризуются увеличенными геометрическими размерами. Здесь нет никаких ограничений для расстояния между компьютерами. Глобальные сети могут объединять как отдельные компьютеры, так и отдельные локальные сети, в том числе использующие различные протоколы.

Компьютерная сеть, объединяя компьютеры, обеспечивает совместное использование ими информации и ресурсов. Информация – это данные и программы, находящиеся в файлах на дисках, а ресурсы – это жесткие диски, принтеры, модемы и другие устройства, установленные на компьютерах, объединенных в сети.

Назначение всех видов компьютерных сетей определяется двумя функциями:

- обеспечение совместного использования аппаратных и программных ресурсов сети;
- обеспечение совместного доступа к ресурсам данных.

Для локальных сетей характерна высокая скорость передачи информации между компьютерами, достигающая 100 Мбит в секунду. Скорость передачи данных в глобальных сетях значительно ниже.

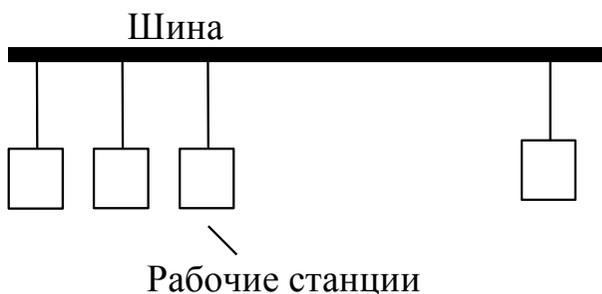
Обеспечение коллективного доступа к информационным ресурсам может осуществляться двумя путями:

- организация работы пользователей с одним компьютером в режиме разделения времени;
- организация локальной сети. В этом случае один из компьютеров выделяется для работы в качестве сервера. Сервер – это может быть как отдельный компьютер с определенным набором функций, выделенный для совместного использования участниками сети, так и программа, выполняющая определенные операции. Абонентом или рабочей станцией называется компьютер, обращающийся по мере необходимости к серверу.

Компьютерные сети, в которых нет выделенного сервера, а все локальные компьютеры могут общаться друг с другом на равных правах (обычно это небольшие сети), называются одноранговыми. У участников компьютерной сети могут быть разные права для доступа к общим ресурсам сети. Совокупность приемов разделения и ограничения прав участников сети называется *политикой сети*. Управление сетевыми политиками (их может быть несколько в сети) называется *администрированием сети*. Лицо, управляющее организацией работы участников локальной компьютерной сети, называется *системным администратором*.

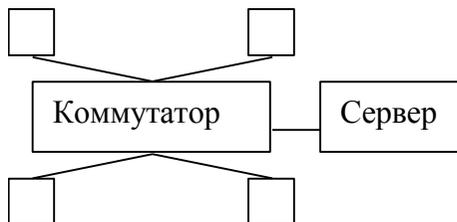
Подключение компьютера к локальной сети может осуществляться разными способами. В зависимости от схемы подключения компьютеров (топологии) локальные сети могут иметь следующую топологию:

- шинная топология – все компьютеры с помощью сетевых адаптеров подключаются к общей магистрали (шине). В процессе работы сети вся информация от передающей рабочей станции поступает на адаптеры всех рабочих станций, воспринимается же адаптером только той рабочей станцией, которой она предназначена.



- топология «звезда» - каждый компьютер через сетевой адаптер подключается к центральному коммутатору. Недостатки: подключение большого числа пользователей требует значительных аппаратных затрат. В современных звездообразных сетях функции коммутации компьютеров и управления сетью разделены между сетевым сервером и коммутатором.

Рабочие станции



- топология кольцо – характеризуется наличием однонаправленного канала передачи данных в виде кольца или петли. Информация передается последовательно между адаптерами рабочих станций до тех пор, пока не будет принята получателем. Недостаток: разрыв кольца обрывает сеть.



Создание локальных сетей характерно для отдельных предприятий или отдельных подразделений предприятия. Отдельные локальные сети могут объединяться в глобальные сети. В этом случае для связи между сетями, работающими по разным протоколам, служат специальные средства, называемые *шлюзами*. Шлюзы могут быть как аппаратными, так и программными. Например, это может быть специальный компьютер (*шлюзовый сервер*), а может быть и компьютерная программа. При подключении локальной сети к глобальной сети на первый план выступают вопросы сетевой безопасности. В частности, должен быть ограничен

доступ в локальную сеть для посторонних лиц извне, а также ограничен выход за пределы локальной сети пользователям, не имеющим соответствующих прав. Для обеспечения сетевой безопасности между локальной и глобальной сетями устанавливают так называемые *брандмауэры*. Это может быть специальный компьютер и компьютерная программа, препятствующая несанкционированному перемещению данных между сетями.

Все пользователи самостоятельно решают, какие данные на своем компьютере сделать общедоступным по сети. На сегодняшний день одноранговые сети бесперспективны. Если к сети подключено более 10 пользователей, то одноранговая сеть, где компьютеры выступают в роли и клиентов, и серверов, может оказаться недостаточно производительной. Поэтому большинство сетей используют выделенные серверы. Выделенным называется такой сервер, который функционирует только как сервер (исключая функции клиента или рабочей станции). Они специально оптимизированы для быстрой обработки запросов от сетевых клиентов и для управления защитой файлов и каталогов. Сети на основе сервера стали промышленным стандартом. Существуют и комбинированные типы сетей, совмещающие лучшие качества одноранговых сетей и сетей на основе сервера.

В производственной практике ЛВС играют очень большую роль. Посредством ЛВС в систему объединяются персональные компьютеры, расположенные на многих удаленных рабочих местах, которые используют совместно оборудование, программные средства и информацию. Рабочие места сотрудников перестают быть изолированными и объединяются в единую систему. Все ЛВС работают в

одном стандарте принятом для компьютерных сетей - в стандарте Open Systems Interconnection (OSI).

Техническое обеспечение ЛВС. Сетевые устройства и средства коммуникаций.

Для объединения компьютеров в локальную сеть требуется вставить в каждый подключаемый к сети компьютер сетевой контроллер, который позволяет компьютеру получать информацию из локальной сети и передавать данные в сеть, а также соединить компьютеры кабелями, по которым происходит передача данных между компьютерами, а также другими подключенными к сети устройствами (принтерами, сканерами и т.д.). В некоторых типах сетей кабели соединяют компьютеры непосредственно, в других соединение кабелей осуществляется через специальные устройства - концентраторы (или хабы), коммутаторы и др. В небольших сетях обычно компьютеры сети соединяются кабелями с концентратором, который и передает сигналы от одних подключенных к нему компьютеров к другим.

В качестве средств коммуникации наиболее часто используются витая пара, коаксиальный кабель и оптоволоконные линии. При выборе типа кабеля учитывают следующие показатели:

- Стоимость монтажа и обслуживания;
- Скорость передачи информации;
- Ограничения на величину расстояния передачи информации (без дополнительных усилителей-повторителей (репитеров));
- Безопасность передачи данных.

Главная проблема заключается в одновременном обеспечении этих показателей, например, наивысшая

скорость передачи данных ограничена максимально возможным расстоянием передачи данных, при котором еще обеспечивается требуемый уровень защиты данных. Легкая наращиваемость и простота расширения кабельной системы влияют на ее стоимость и безопасность передачи данных.

Рабочие станции присоединяются к шине посредством устройств ТАР(англ. Terminal Access Point – точка подключения терминала). ТАР представляет собой специальный тип подсоединения к коаксиальному кабелю. Зонд игольчатой формы внедряется через наружную оболочку внешнего проводника и слой диэлектрика к внутреннему проводнику и присоединяется к нему.

В ЛВС с модулированной широкополосной передачей информации различные рабочие станции получают, по мере надобности, частоту, на которой эти рабочие станции могут отправлять и получать информацию. Пересылаемые данные модулируются на соответствующих несущих частотах, т.е. между средой передачи информации и рабочими станциями находятся соответственно модели для модуляции и демодуляции. Техника широкополосных сообщений позволяет одновременно транспортировать в коммуникационной среде довольно большой объем информации. Для дальнейшего развития дискретной транспортировки данных не играет роли, какая первоначальная информация подана в модем (аналоговая или цифровая), так как она все равно в дальнейшем будет преобразована.

Основные характеристики трех наиболее типичных типов вычислительных сетей.

Характеристики	Топологии		
	Звезда	Кольцо	Шина
Стоимость расширения	Незначительная	Средняя ,	Средняя
Присоединение абонентов	Пассивное	Активное	Пассивное
Защита от отказов	Незначительная	Незначительная	Высокая
Размеры системы	Любые	Любые	Ограничен
Защищенность от прослушивания	Хорошая	Хорошая	Незначительная
Стоимость подключения	Незначительная	Незначительная	Высокая
Поведение системы при высоких нагрузках	Хорошее	Удовлетворительное	Плохое
Возможность работы в реальном режиме времени	Очень хорошая	Хорошая	Плохая
Разводка кабеля	Хорошая	Удовлетворительная	Хорошая
Обслуживание	Очень	Среднее	Среднее

Типы построения сетей по методам передачи информации.

Локальная сеть Token Ring.

Этот стандарт разработан фирмой IBM. В качестве передающей среды применяется неэкранированная или экранированная витая пара (англ. UTP или STP) или оптоволокно. Скорость передачи данных 4 Мбит/с или 16 Мбит/с. В качестве метода управления доступом станций к передающей среде используется метод - маркерное кольцо (англ. Token Ring).

Основные положения этого метода:

- устройства подключаются к сети по топологии кольцо;

- все устройства, подключенные к сети, могут передавать данные, только получив разрешение на передачу (маркер);

- в любой момент времени только одна станция в сети обладает таким правом.

Типы пакетов:

- пакет «управление/данные» (англ. Data/Command Frame) - выполняется передача данных или команд управления работой сети;

- пакет «маркер» (англ. Token) - станция может, начать передачу данных только после получения такого пакета. В одном кольце может быть только один маркер и соответственно, только одна станция с правом передачи данных;

- пакет «сброса» (англ. Abort) - посылка такого пакета вызывает прекращение любых передач. В сети можно подключать компьютеры по топологии «звезда» или «кольцо».

Локальная сеть ArcNet.

ArcNet (англ. Attached Resource Computer Network) - простая, недорогая, надежная и достаточно гибкая архитектура локальной сети.

Разработана корпорацией Datapoint в 1997 году. В качестве передающей среды используются витая пара, коаксиальный кабель (RC-62) с волновым сопротивлением 93 Ом и оптоволоконный кабель. Скорость передачи данных - 2,5 Мбит/с, существует также расширенная версия - ArcNetplus - поддерживает передачу данных со

скоростью 20 Мбит/с. При подключении устройств в ArcNet применяют топологии «шина» и «звезда». Метод управления доступом станций к передающей среде - маркерная шина (англ. Token Bus). Этот метод предусматривает следующие правила:

- все устройства, подключенные к сети, могут передавать данные только получив разрешение на передачу (маркер) в любой момент времени только одна станция в сети обладает таким правом;
- данные, передаваемые одной станцией, доступны всем станциям сети.

Основные принципы работы.

Передача каждого байта в ArcNet выполняется специальной посылкой ISU (англ. Information Symbol Unit - единица передачи информации), состоящей из трех служебных старт/стоповых битов и восьми битов данных. В начале каждого пакета передается начальный разделитель АВ (англ. Alert Burst), который состоит из шести служебных битов. Начальный разделитель выполняет функции преамбулы пакета.

В ArcNet определены 5 типов пакетов:

1. Пакет ИТТ (англ. Information to Transmit) - приглашение к передаче. Эта посылка передает управление от одного узла сети к другому. Станция, принявшая этот пакет, получает право на передачу данных.
2. Пакет FBE (англ. Free Buffer Enquiries) - запрос о готовности к приему данных. Этим пакетом проверяется готовность узла к приему данных.
3. Пакет данных. С помощью этой посылки производится передача данных.
4. Пакет АСК (англ. ACKnowledgments) - подтверждение приема. Подтверждение

готовности к приему данныхх или подтверждение приема пакета даннх без ошибок, т.е. в ответ на FBE и пакет данных.

5. Пакет NAK (Negative Acknowledgments) - неготовность к приему. Неготовность узла к приему данных (ответ на FBE) или принят пакет с ошибкой.

В сети ArcNet можно использовать две топологии: «звезда» и «шина».

Локальная сеть Ethernet.

Спецификацию Ethernet в конце семидесятых годов предложила компания Xerox Corp.

Основные принципы работы. На логическом уровне в Ethernet применяется топология шина.

- все устройства, подключенные к сети, равноправны, т.е. любая станция может начать передачу в любой момент времени (если передающая среда свободна);

- данные, передаваемые одной станцией, доступны всем станциям сети.

Сетевые операционные системы для локальных сетей.

Основное направление развития современных Сетевых Операционных Систем (англ. Network Operation System - NOS) - перенос вычислительных операций на рабочие станции, создание систем с распределенной обработкой данных. Это в первую очередь связано с ростом вычислительных возможностей персональных компьютеров и все более активным внедрением мощных многозадачных операционных систем: OS/2, Windows NT, Windows 9X и современные версии ОС. Кроме этого внедрение объектно-ориентированных

технологий (OLE, ActiveX, ODBC и т.д.) позволяет упростить организацию распределенной обработки данных. В такой ситуации основной задачей NOS становится объединение неравноценных операционных систем рабочих станций и обеспечение транспортного уровня для широкого круга задач: обработка баз данных, передача сообщений, управление распределенными ресурсами сети.

В современных NOS применяют три основных подхода к организации управления ресурсами сети.

Первый - это Таблицы Объектов (англ. Bindery). Используется в сетевых операционных системах NetWare 286 и NetWare 3.XX. Такая таблица находится на каждом файловом сервере сети. Она содержит информацию о пользователях, группах, их правах доступа к ресурсам сети (данным, сервисным услугам, печати через сетевой принтер и т.п.). Такая организация работы удобна, если в сети только один сервер. В этом случае требуется определить и контролировать только одну информационную базу. При расширении сети, добавлении новых серверов объем задач по управлению ресурсами сети резко возрастает. Администратор системы вынужден на каждом сервере сети определять и контролировать работу пользователей. Абоненты сети, в свою очередь, должны точно знать, где расположены те или иные ресурсы сети, а для получения доступа к этим ресурсам - регистрироваться на выбранном сервере.

Второй подход используется в LANServer и Windows NT Server - Структура Доменов (англ. Domain). Все ресурсы сети и пользователи объединены в группы. Домен можно рассматривать как аналог таблиц объектов, только здесь такая таблица является общей для нескольких серверов, при этом ресурсы серверов являются общими для всего домена. Поэтому

пользователю для того чтобы получить доступ к сети, достаточно подключиться к домену, после этого ему становятся доступны все ресурсы домена, ресурсы всех серверов и устройств, входящих в состав домена. Однако и с использованием этого подхода также возникают проблемы при построении информационной системы с большим количеством пользователей серверов и, соответственно, доменов. Здесь эти проблемы уже связаны с организацией взаимодействия и управления несколькими доменами, хотя по содержанию они такие же, как и в первом случае.

Третий подход - Служба Наименований Директорий или Каталогов (англ. Directory Name Services - DNS) лишен этих недостатков. Все ресурсы сети: сетевая печать, хранение данных, пользователи, серверы и т.п. рассматриваются как отдельные ветви или директории информационной системы. Таблицы, определяющие DNS, находятся на каждом сервере. Это, во-первых, повышает надежность и живучесть системы, а во-вторых, упрощает обращение пользователя к ресурсам сети. Зарегистрировавшись на одном сервере, пользователю становятся доступны все ресурсы сети. Управление такой системой также проще, чем при использовании доменов, так как здесь существует одна таблица, определяющая все ресурсы сети, в то время как при доменной организации необходимо определять ресурсы, пользователей, их права доступа для каждого домена отдельно.

В настоящее время наиболее распространенными сетевыми операционными системами являются NetWare 3.XX и 4.XX (Novell Inc.), Windows NT Server 3.51 и 4.00 (Microsoft Corp.) и LAN Server (IBM Corp.).

Контрольные вопросы:

1. Какие виды существуют компьютерных сетей?
2. Топология локальных сетей.
3. Что входит в техническое обеспечение ЛВС?
4. Основные различительные характеристики трёх топологий вычислительных сетей?
5. Какие существуют типы построения сетей по методам передачи информации.?
6. Чем характеризуются основные подходы к организации управления ресурсами сети в современных операционных системах для компьютерных сетей?

Лекция № 6.

Глобальная сеть Интернет. Использование сервисов Интернет в научной деятельности.

Содержание:

1. Краткая история Интернет.
2. Основные протоколы Интернет.
3. Службы Интернет.
4. Поисковые системы и алгоритм поиска информации в Интернете
5. Навигация на Web-странице.
6. Использование электронной почты.

С начала 60-х годов основной областью исследований Агентства по Новейшим Исследованиям (ARPA) при Министерстве Обороны США стало военное использование компьютерных технологий. Персональных компьютеров в то время не было и в помине, и крупные университеты могли позволить себе иметь 1–2 больших компьютера. Соответственно, и машинное время этих компьютеров стоило очень дорого, поэтому компьютеры эксплуатировались круглосуточно. И тогда появилась идея объединить между собой компьютеры разных университетов, чтобы сделать возможным удаленное использование любого свободного в данный момент компьютера. Этот проект получил название ARPANET. К концу 1969 года компьютеры 4-х университетов были объединены между собой, и появилась первая компьютерная сеть.

В 1972 году, когда ARPANET соединял уже 23 компьютера, была написана первая программа для обмена электронной почтой по сети. К этому времени ряд государственных организаций и корпораций создали собственные сети. Однако эти сети обладали одним общим недостатком: они могли соединять только ограниченное число однотипных компьютеров. Кроме того, эти сети были несовместимы между собой.

В середине 70-х годов для ARPANET были разработаны новые стандарты передачи данных, которые позволили объединять между собой сети произвольной архитектуры, тогда же было придумано слово «Интернет». Именно эти стандарты, получившие впоследствии название протокола TCP / IP, заложили основу для роста глобальной компьютерной сети путем объединения уже существующих сетей. В 1983 году сеть ARPANET перешла на новый протокол. В 1988 году Интернет насчитывал уже 56 тысяч компьютеров, но оставался преимущественно университетской сетью. Настоящий расцвет Интернет начался в 1992 году, когда была изобретена новая служба, получившая название «Всемирная паутина» (World Wide Web

или WWW или просто Web). WWW позволял любому пользователю Интернет публиковать свои текстовые и графические материалы в привлекательной форме и предоставляя удобную систему навигации. Постепенно Интернет стал выходить за рамки академических институтов и стал превращаться из средства переписки и обмена файлами в гигантское хранилище информации. В 1999 году Интернет объединял уже около 60 млн. компьютеров и более 275 миллионов пользователей, и каждый день в нем появлялось полтора миллиона новых Web-документов.

Таким образом, Интернет – это не совокупность прямых соединений между компьютерами. Например, если два компьютера, находящиеся на разных континентах, обмениваются данными в Интернете, это не означает, что между ними действует одно прямое соединение. Данные, которые они посылают друг другу, разбиваются на пакеты, и даже в одном сеансе связи разные пакеты одного сообщения могут пройти разными маршрутами. При этом данные, отправленные позже, могут придти раньше, что не помешает правильной сборке документа, поскольку каждый пакет имеет свою маркировку.

Итак, Интернет представляет собой как бы «пространство», внутри которого осуществляется непрерывная циркуляция данных. В Интернете информация перемещается между компьютерами, составляющими узлы сети, и какое-то время хранится на их жестких дисках. *Хостом* в сети Интернет называют компьютеры, работающие в многозадачной операционной системе, поддерживающие протоколы TCP/IP и предоставляющие пользователю какие-либо сетевые услуги.

Протоколы TCP/IP. Согласно рекомендациям Международного института стандартизации ISO системы компьютерной связи принято рассматривать на семи разных уровнях. В современном понимании TCP/IP – это не один сетевой протокол, а два протокола, лежащих на разных уровнях.

Протокол TCP (Transmission Control Protocol) – протокол транспортного уровня. Он управляет тем, как происходит передача информации, осуществляет контроль целостности передаваемой информации. Согласно этому протоколу, отправляемые данные «нарезаются» на небольшие пакеты, после чего каждый пакет маркируется таким образом, чтобы в нем были данные, необходимые для правильной сборки документа на компьютере получателя.

Протокол IP (Internet Protocol) – адресный. Он принадлежит сетевому уровню и определяет, куда происходит передача. Его суть состоит в том, что у каждого участника Всемирной сети должен быть свой уникальный адрес (IP- адрес, выражаемый четырьмя байтами (32 разрядное число), например 356.15.67.11). Ни одно число не должно превышать 255. Адрес Разделен на две части. Первая часть идентифицирует физическую сеть, к которой подключен компьютер, вторая – адрес конкретного компьютера в сети. Без этого нельзя говорить о точной доставке TCP – пакетов на нужное рабочее место. Структура IP-адреса организована так, что каждый компьютер, через который проходит какой-либо TCP-пакет, может по этим четырем числам определить, кому из ближайших «соседей» надо переслать пакет, чтобы он оказался быстрее у получателя. Решением вопроса, каким путем отправлять пакеты, занимаются *маршрутизаторы*– специализированные компьютеры или программа, работающая на узлом сервере сети.

Службы Интернет

Интернет – это в первую очередь средство обмена информацией, и когда говорят об использовании Интернет или работе в Интернет, то подразумевают одну или несколько его служб. В зависимости от конкретных целей клиенты сети используют те или иные службы.

Разные службы имеют разные протоколы. Для того, чтобы воспользоваться какой-то из служб Интернета, надо установить на компьютере программу, работающую по протоколу данной службы. Такие программы называют *клиентскими* или просто *клиентами*. Интернет с самого начала обладал открытой архитектурой. Это означает, что новые службы могут возникать по мере необходимости. Некоторые из них пользуются широкой популярностью, другие отвечают потребностям ограниченного круга пользователей, третьи вытесняются более совершенными конкурирующими службами. Наиболее известны и широко используются следующие службы:

- **Telnet** – исторически одна из ранних служб удаленного управления компьютером. В прошлом эту службу широко использовали для проведения сложных расчетов на удаленных вычислительных центрах. В настоящее время эта служба используется для дистанционного управления техническими объектами (телескопами, промышленными роботами) и системными администраторами.
- **E-Mail** – электронная почта. Выполняет те же функции, что и обычная почта, только быстрее, надежнее, дешевле. Исторически одна из ранних служб Интернет. Ее обеспечением в Интернете занимаются специальные *почтовые серверы* – программы. Существует большое разнообразие клиентских почтовых программ, например Microsoft Outlook 2000, входящий в состав пакета Microsoft Office 2000. Из специализированных почтовых программ можно указать The Bat! и Eudora Pro.
- **MailList** – списки рассылки. Это специальные тематические серверы, собирающие информацию по определенным темам и переправляющие ее подписчикам в виде сообщений электронной почты. Эта служба позволяет пользователям самим вносить себя в список рассылки (подписываться на него) и удалять себя из этого списка (прекращать подписку). Списки рассылки оказались незаменимым средством для организации

рабочих групп, осуществляющих какой-то проект, но могут использоваться и просто для общения с себе подобными. Серверное программное обеспечение для поддержки списков рассылки называется менеджером списков рассылки. При работе со списками рассылки используются два почтовых адреса: административный и обычный. Посылая короткие стандартные команды по административному адресу, можно подписаться на список рассылки, прекратить подписку, получить список всех подписчиков и т.д. Обычный адрес служит непосредственно для обмена сообщениями – письма на этот адрес получают все подписчики данного списка рассылки. Списки рассылки бывают модерлируемыми и немодерлируемыми. В модерлируемых списках рассылки каждое сообщение предварительно направляется на одобрение главному редактору, который называется модератором. В качестве модератора может выступать как человек, так и компьютерная программа. В последнем случае отбор осуществляется по определенным ключевым словам. В немодерлируемых списках рассылка происходит автоматически. Подписка осуществляется автоматически в открытых списках рассылки и направляется на одобрение модератору в закрытых. Назначение модератора – не допустить массовую рассылку сообщений рекламного характера.

- **Usenet news** – служба телеконференций. Служба телеконференций похожа на циркулярную рассылку электронной почты, в ходе которой одно сообщение отправляется не одному корреспонденту, а целой группе (такие группы называют телеконференциями или группами новостей). Сообщения, направляемые на сервер группы новостей, отправляются с него на все серверы, с которыми он связан. Распространяясь во все стороны, менее чем за сутки сообщения охватывают весь земной шар. На каждом из серверов поступившее сообщение хранится определенное время (обычно неделю), и все

желающие могут ознакомиться с ним в течении этого времени. Существуют тысячи групп, посвященных различным вопросам, начиная от туризма и до компьютерных протоколов (сегодня в мире насчитывают порядка 50 000 тематических групп новостей). Пользователь по своему желанию может подписаться на любые группы новостей. Основной прием использования групп новостей состоит в том, чтобы задать вопрос, обращаясь ко всему миру, и получить ответ или совет от тех, кто с этим вопросом уже разобрался. Многие квалифицированные специалисты мира (конструкторы, инженеры, ученые, врачи, юристы, программисты) регулярно просматривают сообщения телеконференций, проходящих в группах, касающихся их сферы деятельности. Такой просмотр называется мониторингом информации. Регулярный мониторинг позволяет специалистам точно знать, что нового происходит в мире по их специальности, какие проблемы беспокоят большие массы людей и на что надо обратить особое внимание в своей работе. В современных промышленных и проектно-конструкторских организациях принято, что специалисты высшего звена периодически отвечают через систему телеконференций на типовые вопросы пользователей своей продукции. В некоторых группах новостей также осуществляется модерация для отсева в первую очередь рекламной информации. В отличие от списков рассылки, группы новостей не существуют по отдельности, а объединены в общую иерархию. Можно участвовать в них, используя Microsoft Internet News (входит в состав Windows, начиная с Windows 95) или Netscape Messenger, который поддерживает не только почту, но и новости. Новостями можно пользоваться, используя браузер, используя сервер deja.com.

- **Web-форумы** выполняют те же функции, что и Web списки рассылки, но используют для этого только браузер. Они выглядят как обычные Web - страницы,

организованные таким образом, что каждый пользователь может дописывать в них свое сообщение. В настоящее время Web -форумы представляют собой наиболее удобное и простое средство для групповых дискуссий и объявлений. Вместе с тем они требуют более продолжительного подключения к Интернету по сравнению со списками рассылки.

- **FTP** – служба передачи файлов. Служба FTP имеет свои серверы в мировой сети, на которых хранятся архивы данных. Со стороны клиента для работы с серверами FTP может быть установлено специальное программное обеспечение, хотя в большинстве случаев браузеры WWW обладают встроенными возможностями для работы по протоколу FTP.
- **IRC** – предназначена для прямого общения нескольких людей в режиме реального времени. Пользователи подключаются к одному из каналов тематических групп и участвуют в разговоре, который ведется не голосом, а текстом. Каждый пользователь может создать свой собственный канал и пригласить в него участников беседы. В отличие от системы телеконференций, где общение доступно всему миру, в системе IRC общение проходит только в пределах одного канала.
- **ICQ** – предназначена в первую очередь для поиска сетевого IP-адреса пользователя, подключенного в данный момент к Интернет. ICQ является акронимом выражения I seek you -я ищу тебя. Первая версия программы была запущена в 1996г. Для пользования этой службой надо зарегистрироваться на ее центральном сервере (<http://www.icq.com>) и получить персональный идентификационный номер UIN (Universal Internet Number). Данный номер можно сообщить партнерам по контактам, и тогда служба ICQ приобретает характер *Интернет-пейджера*. При каждом подключении к Интернет программа ICQ, установленная на компьютере пользователя, определяет текущий IP-адрес и сообщает

его центральной службе, которая, в свою очередь, оповещает его партнеров по контактам. ICQ позволяет обмениваться сообщениями, посылать файлы и URL, играть по сети в различные игры или просто болтать в мини-чате с приятелями. Удобная и несложная система настроек позволяет самостоятельно выбрать требуемый уровень конфиденциальности – от простейшего до самого «засекреченного», когда вы видите всех, а вас не видит никто. На начало 2000 г. число пользователей программы превысило 60 млн человек. Информация об ICQ доступна на сервере <http://www.icq.ru/>.

- **WAIS** – диалоговая система с оконным интерфейсом для поиска данных по ключевым словам в контексте. Запросы посылаются в WAIS на упрощенном английском языке. В сети Интернет существует более 200 WAIS- библиотек. Большая часть материалов этих библиотек относится к области научных исследований и компьютерных наук.
- **Word Wide Web (WWW)** – самая популярная служба Интернет, хотя ее часто отождествляют с Интернет. WWW – это единое информационное пространство, состоящее из сотен миллионов взаимосвязанных электронных документов, хранящихся на *Web-серверах*. Отдельные документы, составляющие *пространство Web*, называют *Web-страницами*. Группы тематически объединенных Web-страниц называют *Web-узлами (Web-сайтами* или просто *сайтами*). Один физический Web-сервер может содержать достаточно много Web-узлов, каждому из которых отводится отдельный каталог на жестком диске сервера. Электронные Web-документы предназначены для просмотра на экране компьютера, причем заранее неизвестно на каком. Поэтому Web-документы не могут иметь «жесткого» форматирования. Оформление выполняется непосредственно во время их воспроизведения на компьютере клиента и происходит оно в соответствии с настройками программы, выполняющей просмотр. Программы для просмотра Web-

страниц называют *браузерами*. Браузер выполняет отображение на экране, руководствуясь командами, которые автор документа внедрил в его текст. Такие команды называются *тегами*.

Основными компонентами технологии WWW являются следующие:

- *язык гипертекстовой разметки документов HTML*. С любым фрагментом текста или с рисунком с помощью тегов можно связать иной Web-документ, то есть установить гиперссылку. В этом случае при щелчке левой кнопкой мыши на тексте или рисунке, являющемся гиперссылкой, отправляется запрос на доставку нового документа. Этот документ может иметь гиперссылки на другие документы;
- *универсальный способ адресации ресурсов в сети URL*. Гипертекстовая связь между сотнями миллионов документов, хранящихся на серверах Интернета, была бы невозможна, если бы каждый документ в этом гиперпространстве документов не обладал своим уникальным адресом. Адрес любого файла во всемирном масштабе определяется *унифицированным указателем ресурса – URL*.

Адрес URL состоит из трех частей.

1. Указание службы, которая осуществляет доступ к данному ресурсу (обычно обозначается именем прикладного протокола, соответствующего данной службе. Так, например, для службы WWW прикладным является протокол HTTP (HyperText Transfer Protocol – протокол передачи гипертекста). После имени протокола ставится двоеточие(:) и два знака «/» косая черта):

http://....

2. Указание доменного имени компьютера (сервера), на котором хранится данный ресурс:

http://www.abcd.com....

3. Указание полного пути доступа к файлу на данном компьютере. В качестве разделителя используется символ «/»:

<http://www.abcd.com/Files/New/abcdef.zip>

При записи URL-адреса важно точно соблюдать регистр символов. В отличие от правил работы в MS DOS и Windows, в Интернете строчные и прописные символы считаются разными.

- третьей компонентой является протокол обмена данными – HTTP. Данный протокол предназначен для обмена гипертекстовыми документами и учитывает специфику такого обмена.
- последней составляющей является спецификация Common Gateway Interface (CGI). Она была специально разработана для расширения возможностей WWW за счет подключения внешнего программного обеспечения. Программа, написанная в соответствии со спецификацией CGI, называется CGI - скриптом. Они могут быть написаны на любом языке программирования (C, C++, Pascal) или на командном языке (MS DOS, Perl).

Адресация в Интернете

Для указания адреса информационного ресурса в Интернете используются несколько систем имен. Наиболее широко распространена *региональная система имен*. В начальный период своего развития система Интернет была небольших размеров, и каждый пользователь Интернета имел свое собственное имя. На узле сети была создана регистратура, куда посылался запрос, и в ответ можно было получить файл – список имен и адресов. Этот файл называется *«host file»*. Он регулярно распространялся по всей сети. Имена были простыми и уникальными. По мере расширения Интернет остро встал вопрос регистрации и получения имени новым компьютером. Да и сам host file принял громадные размеры. Для устранения возникших трудностей была принята новая система имен. Ее назвали *доменной системой имен* – DNS, а способ адресации – способом адресации по доменному принципу. Доменная

система имен – это назначение имен путем передачи сетевым группам ответственности за их подмножество имен. Строится эта система по иерархическому принципу. Каждый уровень этой системы называется доменом. В 80-х годах в США были определены первые домены верхнего уровня: образовательных (edu), коммерческих (com), государственных (gov). Когда сеть Интернет стала принимать международный характер и перешагнула за пределы США, появились национальные домены – двухбуквенная кодировка государств. Например, Узбекистан имеет кодировку uz , Россия - ru , США – us. В настоящее время почти 300 стран имеют такого рода кодировку. Единый каталог Интернет находится у SRI International - государственной организации США.

Вслед за доменами, определяющими национальную (государственную) принадлежность, следуют домены, определяющие либо регионы, либо организации. Далее следуют следующие уровни иерархии.

Домены в именах отделяются друг от друга точками. Первым в имени стоит название рабочей машины – реального компьютера с IP-адресом. Это имя создано и поддерживается группой, которая в свою очередь является частью национальной сети. Например,.....

Поиск адреса по доменному имени осуществляется в следующем порядке. После ввода имени компьютер должен преобразовать его в адрес. Для этого он начинает запрашивать помощь у DNS-серверов. Это узлы, обладающие соответствующей базой. DNS-сервер начинает обработку имени с правого его конца и двигается по нему влево, постепенно сужая поиск.

Поиск информации в Интернете

Поиск информации в Интернете обеспечивается наличием поисковых серверов. Поисковыми серверами называют выделенные компьютеры, которые автоматически

просматривают все ресурсы Интернета и индексируют их содержание. Достаточно такому серверу передать фразу или набор ключевых слов, описывающих интересующую вас тему, и сервер возвратит вам список ресурсов, то есть список гиперссылок на страницы, на которых упоминаются сведения, соответствующие вашему запросу. Таких серверов в настоящее время довольно много, например, InfoSeek (www.infoseek.com), Rambler (www.rambler.ru), Яндекс (www.yandex.ru). Поисковые системы классифицируют по методам поиска.

Окно поисковой системы представлено на рис. 6.1.

Поиск информации в Интернете может производиться и с помощью каталогов. В каталогах Интернет хранятся тематически систематизированные коллекции ссылок на различные сетевые ресурсы, в первую очередь на документы WWW. Занесение ссылок в каталоги осуществляется не автоматически, а специально занимающимися этим людьми. Естественно, они хотят сделать свои каталоги наиболее полными. Пользователю предлагается самостоятельно перемещаться по иерархической структуре разделов и подразделов, на нижнем уровне которой располагается небольшое число ссылок, заслуживающих внимания. Поисковый каталог обеспечивает высокое качество поиска. В настоящее время широко используются глобальные каталоги сети, такие как Yahoo! (www.yahoo.com), atRus (www.atrus.ru).



Рис.6.1. Окно поисковой системы Rambler.ru

Поисковый индекс обеспечивает поиск по заданным ключевым словам. В результате поиска формируется набор гиперссылок на Web-страницы, содержащие указанные термины.

При проведении первичного поиска по конкретной теме целесообразно использовать поисковые каталоги. Для специалистов, хорошо знакомых с ресурсами Интернета по своей специальности, более полезны поисковые индексы. Они позволяют разыскивать малоизвестные и узкоспециализированные ресурсы.

Программа Internet Explorer 5.0 имеет специальные средства поиска без явного обращения к поисковым системам. Для этого необходимо ввести ключевое слово **go**, **find** или **?** и ключевую фразу или набор ключевых слов. Поиск будет производиться с помощью поисковой системы, заданной по умолчанию. Результаты поиска отображаются в виде списка ссылок.

Помимо вышеуказанных, существуют и специализированные поисковые серверы:

- для поиска E-mail, адресов и людей/компаний: Fourll Directory, Lookup, Nynex Interactive Yellow Pages for business, Phone Directory;
- поиск программного обеспечения: FTP Search, Jumbo;
- поиск в телеконференциях: DejaNews.

Просмотр информации в WWW.

Гипермедийные документы Web существуют в виде файлов HTML-формата, и для их просмотра нужна специальная программа, называемая Интернет обозревателем или браузером. Большинство обозревателей являются графическими и способны работать под управлением Windows, начиная с Windows 95 и последующих версий. Наиболее популярны браузеры Microsoft Internet Explorer и Netscape Navigator. Запуск Internet Explorer осуществляется выполнением команды **Пуск►Программы►Internet Explorer** или щелкнув по значку Internet Explorer на **Рабочем столе**. Окно Internet Explorer представлено на рис. 7.2. Окно имеет такие же элементы графического интерфейса, как и большинство приложений Windows: строка заголовка, главное меню, панели инструментов, полосы прокрутки. Важнейшим элементом окна обозревателя является поле **Адрес**, которое служит для указания адреса Web-страницы. Состав панелей Internet Explorer можно изменить, используя меню Вид. Меню Сервис позволяет вызвать программу Microsoft Outlook для операций с электронной почтой и новостями. Чтобы еще более упростить доступ к программам, файлам, папкам и избранным Web-страницам, можно разместить панель инструментов на панели задач Windows.

Поиск информации в Web.

Поиск информации в Web может осуществляться несколькими способами. Один из них – адреса URL – универсальных локаторов ресурсов, отображаемых в поле Адрес. После запуска Internet Explorer автоматически загружается «домашняя» (начальная) страница. Чаще всего

это страница фирмы-разработчика Web-обозревателя. Чтобы направиться куда-то в Web, необходимо внести в поле **Адрес** полный адрес, например,

Навигация на Web-странице.

Для навигации на Web-странице имеется несколько способов.

Связи – это подсвеченный текст или пиктограммы, на которые надо щелкнуть для просмотра дополнительной информации. При установке на связь-объект указателя мыши он приобретает вид кисти руки. Это означает, что такое слово является гипертекстовой ссылкой. Каждая связь указывает на другой документ Web или графическое изображение, видео- или аудиоклип или на другой файл.

Красные связи – показывают связи станций, которые вы уже посетили, поэтому можно пропустить ее.

Кнопка **Назад** – возвращает к предыдущему документу. Кнопка **Вперед** – выводит следующий документ.

Кнопка **Домой** – повторно загружает первую страницу, которая выводилась при входе в Internet Explorer.

Кнопка **Обновить** – повторно загружает текущий выводимый документ, если при передаче он был искажен.

Кнопка **Журнал** на стандартной панели инструментов – показывает список посещений и позволяет возвращаться в ранее посещенные места.

Explorer позволяет открывать несколько документов Web одновременно, при этом каждый документ будет отображаться в отдельном окне. Для этого необходимо выполнить команду **Файл ► Создать ► Окно**.

Internet Explorer позволяет записать нужные страницы в папку **Избранное**. Для этого необходимо перейти на добавляемую страницу, затем в меню **Избранное** выбрать пункт **Добавить в избранное**. При желании можно отредактировать название для данной страницы в папке.

Чтобы открыть одну из избранных страниц, в меню **Избранное** выбрать из списка нужную страницу. По мере

увеличения количества избранных страниц их можно упорядочить по папкам.

Адреса часто используемых Web-страниц Internet Explorer позволяет записать на панели ссылок, которая расположена рядом с адресной строкой. Добавить страницу на панель ссылок можно несколькими способами:

- перетащить значок выбранной страницы из адресной строки на панель ссылок;
- перетащить ссылку с Web-страницы на панель ссылок;
- перетащить ссылку в папку **Ссылки** в списке **Избранное**.

Чтобы вызвать страницу, достаточно выбрать ссылку на нее.

Если при просмотре информации на Web-странице текст на экране выглядит написанным непонятными символами, то, скорее всего, следует изменить кодировку гипертекста. Для включения функции автоматического выбора кодировки в меню **Вид** обозревателя следует указать пункт **Вид кодировки** и включить опцию **Автоматический выбор**.

Электронная почта

Электронная почта – самая эффективная из служб Интернет в смысле затрат. Обычно она достигает любой точки земного шара за несколько минут. Электронная почта не использует географическую адресацию, и ее адреса состоят из двух частей:

имя_абонента@адрес_компьютера_в_Интернет

Например, shavkat@tgtu.sarcor.uz. Прописные и строчные буквы в почтовом адресе не различаются.

Электронную почту чаще всего используют следующим образом. Соединившись с провайдером, запускают почтовую программу и получают накопившуюся почту. Подготовив ответы и новые письма, отправляют их. Если какое-то из писем не может быть сразу доставлено адресату, например, если в данный момент не работает сервер, на котором находится его почтовый ящик, то письмо будет поставлено в

очередь на отправку. Каждые 10-15 минут будут производиться новые попытки. Если через несколько часов сообщение все еще не отправлено, то отправитель получает первое (предварительное) уведомление, к которому приложена копия отправленного сообщения. Это уведомление носит информационный характер и не требует от отправителя никакой реакции. Попытки отправить сообщение будут продолжаться еще несколько дней. Если и они будут безуспешны, то отправитель получает второе (окончательное) уведомление с копией отправленного сообщения. Таким образом, любое отправленное сообщение не может затеряться и исчезнуть бесследно.

Почтовые программы для персональных компьютеров используют разные протоколы для приема и отправки почты. При отправке почты программа взаимодействует с сервером исходящей почты, или SMTP-сервером, по протоколу SMTP. Прием почты осуществляется по протоколу POP3. Используется также протокол IMAP, который позволяет выборочно копировать письма, пришедшие на адрес получателя, с почтового сервера на компьютер получателя.

Существует очень много почтовых программ, значительная часть из которых бесплатна. Наиболее распространены следующие программы:

- **Microsoft Internet Mail** – поставляется в составе операционной системы, начиная с Windows 95; более свежая версия может быть скопирована с сервера www.microsoft.com. Эта программа наиболее корректно работает с письмами на русском языке, однако не поддерживает IMAP-протокол.

- **Outlook Express**. Для работы с электронной почтой в Internet Explorer имеется компонент Outlook Express. Для его запуска надо выбрать команду **Сервис►Почта** и новости и далее команду в зависимости от вида работы.

Для создания нового сообщения выбрать команду **Создать сообщение**. В открывшемся окне, выбрав команду **Формат►Вид кодировки**, определить кодировку символов сообщения, ввести адрес получателя, указать тему сообщения

и ввести текст сообщения. Для вставки в текст рисунка выбрать команду **Вставка►Рисунок**, открыть окно **Рисунок**, выбрать файл, указать параметры его расположения в бланке сообщения и щелкнуть **ОК**.

Выделив фрагмент текста в сообщении и выбрав команду **Вставка►Гиперссылка**, можно определить адрес Web-документа в Интернет, который будет раскрываться при щелчке мышкой по этому тексту. Командой **Формат►Фон** можно определить фоновый рисунок и звук, цвет фона сообщения.

Для отправки с сообщением какого-нибудь файла щелкнуть кнопку **Вложить** на стандартной панели инструментов Outlook Express, в диалоговом окне **Вставка** выбрать папку и файл и нажать кнопку **Вложить**. После этого на бланке сообщения появится поле **Присоединить:**, в котором будет имя вложенного файла.

Для отправки подготовленного сообщения щелкнуть кнопку **Отправить**. Можно подготовить в папке **Исходящие** несколько сообщений, завершая их создание выбором команды **Файл►Отправить позже**. В этом случае все подготовленные сообщения будут отправлены при первом подключении к сети.

При подключении к сети Outlook Express загрузит сообщения с почтового сервера. Для получения сообщений выбрать команду **Сервис►Почта и новости►Чтение почты**. Для просмотра полученных сообщений щелкнуть значок папки **Входящие** на панели Outlook Express или в списке папок. Для просмотра сообщений в отдельном окне дважды щелкнуть по нему в списке сообщений. Для ответа на сообщение можно щелкнуть кнопку **Ответить** на стандартной панели инструментов.

Outlook Express поддерживает IMAP-протокол.

- Netscape Messenger – бесплатная программа фирмы Netscape, входит в состав Netscape Communicator. Обладает достоинствами Outlook Express.

- Eudora – почтовая программы фирмы Qualcomm. Полная версия программы продается за деньги. Не рекомендуется для переписки по-русски.

При первом запуске почтовой программы автоматически запускается мастер по настройке, который запрашивает минимальную необходимую информацию: адрес и название серверов входящей и исходящей почты. Может отдельно запрашиваться имя пользователя – эта та часть почтового адреса, которая находится слева от знака «@». При первой проверке почты и всех последующих, если не пометить галочкой «Запомнить пароль», программа приглашает ввести пароль. Если программа поддерживает как POP3, так и IMAP, при настройке следует выбрать POP3.

Можно воспользоваться почтой и не имея почтовой программы. Существует большое количество серверов, которые предлагают завести бесплатный почтовый ящик и позволяют работать с почтой, используя только браузер. Это службы mail.ru, www.tomcat.ru, hotmail.com, mail.yahoo.com и другие. Такое использование почты имеет определенные достоинства. Можно менять провайдеров, не меняя свой адрес электронной почты. Можно просматривать свою почту с любого компьютера, подключенного к Интернет, и не тратить время на конфигурирование почтовой программы. Недостатком такой почты является то, что нельзя минимизировать время подключения к Интернет в той мере, в какой позволяют почтовые программы. К тому же общедоступные почтовые сервера зачастую перегружены.

Контрольные вопросы

1. Что такое компьютерные сети, каков их состав, назначение?
2. Виды сетей и способы передачи информации в них.
3. Что такое Интернет? Какие возможности он предоставляет?
4. Какие возможности для поиска информации в Интернет вы знаете? Чем отличаются информационно-поисковые серверы от каталогов?
5. Перечислите сервисы Интернета и опишите их назначение.
6. Что такое WWW и каковы основные компоненты технологии WWW?
7. Какие способы навигации в Интернете предоставляет пользователю Internet Explorer?
8. Какие вы знаете серверы общения в Интернете, их особенности.
9. Что такое электронная почта, каковы ее отличительные особенности и преимущества?
10. Для чего создаются конференции в Интернет?

Лекция № 7.

Вопросы компьютерной безопасности, методы её реализации.

Содержание:

1. Понятие компьютерной безопасности.
2. Компьютерные вирусы.
3. Методы и средства антивирусной защиты.
4. Меры безопасности при работе в Интернет.
5. Криптографические системы защиты информации в сети.

Понятие безопасности в вычислительной технике является весьма широким. Оно подразумевает и надежность работы компьютера, и сохранность ценных данных, и защиту информации от внесения в нее изменений неуполномоченными лицами, и сохранение тайны переписки в электронной связи. В сфере вычислительной техники правоприменительная практика пока развита недостаточно, поэтому надежность работы компьютерных систем во многом опирается на меры самозащиты. Как говорится, «спасение утопающих – дело рук самих утопающих».

Компьютерные вирусы

Массовое применение персональных компьютеров оказалось тесно связанным с появлением программ-вирусов, препятствующих нормальной работе компьютера, разрушающих файловую структуру дисков и наносящих ущерб хранимой в компьютере информации. Компьютерный вирус – это программный код, встроенный в другую программу или в документ, или в определенные области носителя данных и предназначенный для выполнения несанкционированных действий на несущем компьютере. Вирус не может существовать в «полной изоляции»: он должен каким-то образом обеспечить передачу себе управления и, соответственно, он использует код других программ, информацию о файловой структуре или просто имена других программ.

Итак, компьютерный вирус – это специально написанная программа, способная самопроизвольно

присоединяться к другим программам, создавать свои копии и внедрять их в файлы, системные области компьютера и в вычислительные сети с целью нарушения работы программ, порчи файлов и каталогов, создания всевозможных помех в работе компьютера.

В настоящее время известно более 5000 программных вирусов, и классифицировать их можно по разным признакам. Например, может быть предложена следующая классификация:

- по среде обитания;
- по способу заражения среды обитания;
- по воздействию;
- по особенностям алгоритма.

В зависимости от среды обитания вирусы можно разделить на сетевые, файловые, загрузочные и файлово-загрузочные. **Сетевые вирусы** распространяются по различным компьютерным сетям. **Файловые вирусы** внедряются главным образом в исполняемые модули, т.е. в файлы, имеющие расширение .exe или .com. Файловые вирусы могут внедряться и в другие типы файлов, однако в таких типах файлов они теряют способность к размножению. **Загрузочные вирусы** внедряются в загрузочный сектор диска (Boot-сектор) или в сектор, содержащий программу загрузки системного диска (Master Boot Record). **Файлово-загрузочные** вирусы поражают как файлы, так и загрузочные сектора дисков.

По способу заражения вирусы делятся на резидентные и нерезидентные. Резидентный вирус при заражении компьютера оставляет в оперативной памяти свою резидентную часть, которая потом перехватывает обращение операционной системы к объектам заражения (файлам, загрузочным секторам дисков) и внедряется в них. Резидентные вирусы находятся в памяти и являются активными вплоть до выключения или перезагрузки компьютера. Нерезидентные вирусы не заражают (не инфицируют) память компьютера и являются активными

ограниченное время.

По степени воздействия вирусы можно разделить на следующие группы:

- неопасные, не мешающие работе компьютера, но уменьшающие объем оперативной памяти и памяти на дисках. Действие этих вирусов проявляется в каких-либо графических или звуковых эффектах;
- опасные вирусы, которые могут привести к различным нарушениям в работе компьютера;
- очень опасные, воздействие которых может привести к потере программ, уничтожению данных, стиранию информации в системных областях диска.

Из-за большого разнообразия вирусы трудно классифицировать по особенностям алгоритма. Простейшие вирусы – *паразитические*, они изменяют содержимое файлов и секторов диска и могут быть достаточно легко обнаружены и уничтожены. Можно отметить *вирусы-репликаторы*, называемые червями, которые распространяются по компьютерным сетям, вычисляя адреса сетевых компьютеров и записывают по этим адресам свои копии. Известны вирусы-невидимки, называемые *стелс-вирусами*, которые очень трудно обнаружить и обезвредить, так как они перехватывают обращения операционной системы к пораженным файлам и секторам дисков и подставляют вместо своего тела незараженные участки диска. Наиболее трудно обнаружить полиморфные вирусы (*вирусы-мутанты*), содержащие алгоритмы шифровки-расшифровки, благодаря которым копии одного и того же вируса не имеют ни одной повторяющейся цепочки байтов. Имеются также *квазивирусные* или «троянские» программы, которые не способны к самораспространению, однако представляют большую угрозу: маскируясь под полезную программу, разрушают загрузочный сектор и файловую систему дисков.

Компьютерные вирусы можно классифицировать и по типам:

- программные вирусы;

- загрузочные вирусы;
- макровирусы.

Программные вирусы – это блоки программного кода, целенаправленно внедренные внутрь других прикладных программ. При запуске программы, несущей вирус, происходит запуск имплантированного в нее вирусного кода. Работа этого кода вызывает скрытые от пользователя изменения в файловой системе жестких дисков и/или в содержании других программ. Вирусный код может воспроизводить себя в теле других программ – этот процесс называется размножением. По прошествии определенного времени, создав достаточное количество копий, программный вирус может перейти к разрушительным действиям – нарушению работы программ и операционной системы, удалению информации, хранящейся на диске. Этот процесс называется вирусной атакой.

Считается, что никакой вирус не в состоянии вывести из строя аппаратное обеспечение компьютера. Однако бывают случаи, когда программные повреждения приходится устранять заменой аппаратных средств. Например, в большинстве современных материнских плат базовая система ввода-вывода (BIOS) хранится в перезаписываемых постоянных запоминающих устройствах (так называемая флэш-память). Возможность перезаписи информации в микросхеме флэш-памяти используют некоторые программные вирусы для уничтожения данных BIOS. В этом случае для устранения последствий заражения требуется либо замена микросхемы с BIOS, либо ее перепрограммирование на программаторе.

Программные вирусы поступают на компьютер при запуске непроверенных программ, полученных на внешнем носителе или принятых из Интернета. При обычном копировании зараженных файлов заражение компьютера произойти не может. В связи с этим все данные, принятые из Интернета, должны проходить обязательную проверку на безопасность. Обычный прием распространения «тройанских

коней» - приложение к электронному письму с рекомендацией извлечь и запустить якобы полезную программу.

От программных вирусов загрузочные отличаются методом распространения. Они поражают не программные файлы, а определенные системные области магнитных носителей (гибких и жестких дисков). Обычно заражение происходит при попытке загрузки компьютера с магнитного носителя, системная область которого содержит загрузочный вирус. При попытке загрузить компьютер с гибкого диска происходит сначала проникновение вируса в оперативную память, а затем в загрузочный сектор жесткого диска. Далее этот компьютер становится сам источником распространения загрузочного вируса.

Макровирусы поражают документы, выполненные в некоторых прикладных программах, имеющих средства для исполнения так называемых макрокоманд. К таким документам относятся документы текстового процессора Word. Заражение происходит при открытии файла документа в окне программы, если в ней не отключена возможность исполнения макрокоманд.

Рассмотрим схему функционирования простого загрузочного вируса, заражающего дискеты. В загрузочных вирусах выделяют две части – голову и хвост. Хвост может быть и пустым. Как только в зараженный компьютер вставлена дискета, не защищенная от записи, вирус приступает к заражению. Инфицирование вирус производит следующим образом:

- выделяет некоторую область диска и помечает ее как недоступную операционной системе (в простейшем случае занятые вирусом сектора помечаются как сбойные - bad);
- копирует в выделенную область диска свой хвост и оригинальный (здоровый) загрузочный сектор;
- замещает программу начальной загрузки в загрузочном секторе (настоящем) своей головой;

- организует цепочку передачи управления согласно схеме.

Таким образом, голова вируса теперь первой получает управление, вирус устанавливается в память и передает управление оригинальному загрузочному сектору. Вместо нормальной цепочки загрузки **Программа начальной загрузки (ПЗУ)→Программа начальной загрузки (диск)→Система** в ней появляется новое звено и загрузка осуществляется по следующей цепочке **Программа начальной загрузки (ПЗУ)→Вирус→Программа начальной загрузки (диск)→Система**.

Файловые вирусы действуют по следующей схеме. При запуске исполняемых файлов система считывает из записи в каталоге первый кластер файла и далее все остальные кластеры. Вирус записывается в некоторые свободные секторы дискеты, которые он помечает как сбойные. Кроме того, он сохраняет информацию о первых кластерах исполняемых файлов в резервных битах, а на место этой информации записывает ссылки на себя. Таким образом, при запуске любого файла вирус получает управление (операционная система запускает его сама), резидентно устанавливается в память и передает управление вызванному файлу.

Причины появления и распространения компьютерных вирусов заключаются как в психологии человеческой личности и ее теневых сторонах (зависти, мести, тщеславии непризнанных творцов), так и в отсутствии аппаратных средств защиты и противодействия со стороны операционной системы компьютера.

Методы и средства антивирусной защиты

Основными признаками заражения компьютера являются следующие:

- прекращение работы или неправильная работа ранее успешно функционировавших программ;

- медленная работа компьютера;
- невозможность загрузки операционной системы;
- исчезновение файлов и каталогов или искажение их содержимого;
- изменение даты и времени модификации файла;
- изменение размеров файлов;
- неожиданное значительное увеличение количества файлов на диске;
- существенное уменьшение размера свободной оперативной памяти;
- вывод на экран непредусмотренных сообщений или изображений;
- частые зависания и сбои в работе компьютера.

Существует три рубежа защиты от компьютерных вирусов:

- предотвращение поступления вирусов;
- предотвращение вирусной атаки, если вирус все-таки попал в компьютер;
- предотвращение разрушительных последствий, если атака все-таки произошла.

Соответственно существуют три метода реализации защиты:

- программные методы защиты;
- аппаратные методы защиты;
- организационные методы защиты.

Основным средством защиты данных является их резервное копирование. Резервные копии важных или конфиденциальных данных сохраняют на внешних носителях, которые хранят в сейфах. К программным методам защиты относят использование антивирусных программ, которые можно разделить на следующие виды:

1. **программы-детекторы.** Они осуществляют поиск вирусов в оперативной памяти и в файлах и при

обнаружении их выдают соответствующее сообщение. Недостатком является то, что они могут находить только те вирусы, которые известны разработчикам таких программ;

2. программы-доктора, фаги, программы-вакцины. Они осуществляют поиск зараженных файлов и их лечение, т.е. удаляют из файла тело программы-вируса. Поиск вирусов и их уничтожение осуществляется вначале в оперативной памяти, а затем переходят к «лечению» файлов. Наиболее известные полифаги: Aidstest, Scan, Norton AntiVirus, Doctor Web, Casperski. Учитывая появление новых вирусов, требуется регулярное обновление версий программ детекторов и программ-докторов;

3. программы-ревизоры. Они относятся к самым надежным средствам защиты от вирусов. Ревизоры запоминают исходное состояние программ, каталогов и системных областей диска тогда, когда компьютер не заражен, а затем периодически сравнивают текущее состояние с исходным. Обнаруженные изменения выводятся на экран. Может осуществляться также очистка проверяемой программы от изменений, внесенных вирусом. К числу программ-ревизоров относится программа Adinf;

4. программы-фильтры, или «сторожа». Они представляют собой небольшие резидентные программы, предназначенные для обнаружения подозрительных действий при работе компьютера, характерных для вирусов. Это могут быть попытки коррекции файлов с расширением .exe, .com, изменение атрибутов файла, запись в загрузочные сектора диска, загрузка резидентной программы. При попытке какой-либо программы совершить указанные действия «сторож» посылает пользователю сообщение и предлагает запретить или разрешить указанные действия. Программы фильтры весьма полезны, так как позволяют обнаружить вирус на

ранней стадии его существования, в то же время они не печат файлы и диски. Примером программы-фильтра является программа Vsafe, входящая в состав пакета утилит MS DOS;

5. вакцины, или иммунизаторы. Это резидентные программы, предотвращающие заражение файлов. Вакцина модифицирует программу или диск таким образом, что вирус воспринимает их зараженными и не внедряется в них. Вакцины имеют ограниченное применение.

К аппаратным средствам защиты можно отнести простое отключение перемычки на материнской плате, которое не позволит осуществить стирание перепрограммируемой микросхемы ПЗУ (флэш-BIOS), независимо от того, кто будет пытаться это сделать: компьютерный вирус, злоумышленник или неаккуратный пользователь.

Основными мерами по профилактике заражения компьютера вирусами являются следующие:

1. Оснастить свой компьютер современными антивирусными программами и периодически обновлять их версии.

2. Перед считыванием с дискет информации, записанной на других компьютерах, всегда проверять дискеты на наличие вирусов.

3. При переносе на свой компьютер файлов в архивированном виде проверять их сразу после разархивации на жестком диске.

4. Периодически проверять на наличие вирусов жесткие диски компьютера, запуская антивирусные программы для проверки памяти, файлов и системных областей дисков с защищенной от записи дискеты, предварительно загрузив операционную систему с защищенной от записи системной дискеты.

5. Всегда защищать свои дискеты от записи при работе на других компьютерах, если на эти дискеты не будет производиться запись информации.

6. Обязательно делать архивные копии на дискетах ценной информации.

7. Не оставлять в приемнике дисководов дискеты при включении или перезагрузке операционной системы, чтобы исключить заражение компьютера загрузочными вирусами.

8. Использовать антивирусные программы для входного контроля всех исполняемых файлов, получаемых из компьютерных сетей.

Меры безопасности при работе в Интернет

При работе в Интернет необходимо, в первую очередь, соблюдать все меры предосторожности, какие приняты и при работе с локальным компьютером. Специфическими мерами являются следующие.

При загрузке Web-страниц на диск могут попасть активные Java - апплеты, объекты ActiveX и программные файлы, нарушающие работу компьютера. Internet Explorer обеспечивает повышенную безопасность и возможность использования 128-битного ключа шифрования. Он позволяет разделить просматриваемые Web-узлы сети на четыре зоны, каждая из которых, в свою очередь, имеет несколько уровней безопасности. Самый строгий режим следует предусмотреть для незнакомых Web-узлов. Чем выше уровень защиты, тем больше будет ограничений для передачи информации с Web-узла на ваш компьютер.

Подключенный к Интернет компьютер можно защитить от несанкционированного доступа с помощью брандмауэра, который входит в комплект поставки Windows XP. Брандмауэр позволяет использовать Windows XP в роли Интернет-шлюза для других систем локальной сети.

Одним из методов защиты информации в Интернет является шифрование данных. В настоящее время в Интернете используются несимметричные криптографические системы,

основанные на использовании двух ключей – открытого (публичного) и закрытого (личного).

Контрольные вопросы

1. Что такое компьютерный вирус? Какими свойствами обладают компьютерные вирусы?
2. По каким признакам классифицируют компьютерные вирусы? Перечислите типы вирусов.
3. Какие вирусы называются резидентными и в чем особенность таких вирусов?
4. Каковы отличия вирусов-репликаторов, мутантов и «троянских» программ?
5. Опишите схему функционирования загрузочного вируса.
6. Опишите схему функционирования файлового вируса.
7. Что такое полиморфный вирус? В чем заключается его опасность?
8. Каковы пути проникновения вирусов в компьютер и признаки заражения компьютера?
9. Перечислите основные меры по защите от компьютерных вирусов.

Лекция № 8,9.

Языки программирования в Интернете. HTML, PHP, JAVASCRIPT.

Содержание:

1. Понятие скрипт-языка.
2. Язык HTML. Создание web-страниц.
3. Создание страницы HTML.
4. Структура документа HTML.
5. Основные теги языка

С активным развитием глобальной сети было создано немало реализаций популярных языков программирования, адаптированных специально для Интернета. Все они отличаются характерными особенностями: языки являются интерпретируемыми, интерпретаторы для них распространяются бесплатно, а сами программы – в исходных текстах. Такие языки называют скрипт – языками.

Для пользователей Интернета предоставлены самые разнообразные стандартные программы, работающие с узкоспециализированными задачами. Все эти специализированные средства хороши для решения стандартных задач, на которых они рассчитаны. Скажем, TELEPORT PRO хорош, если нам надо выкачать сайт целиком, а WATZNEW! - для отслеживания обновлений всяческих лент новостей, но более опытные пользователи для реализации своих нестандартных творческих замыслов используют методы программирования, то есть приступают к программированию.

Языков программирования, пригодных для работы с Интернетом существуют великое множество. В последнее время с интернет-технологиями, как правило, ассоциируется два из них: PERL и JAVA.

Язык PERL был создан в 1986 году как инструмент

для администрирования и конфигурирования системных ресурсов в сети, состоящий из UNIX-компьютеров. PERL - это самостоятельный и мощный язык программирования, ориентированный на обработку текста. При этом он совсем несложен для освоения и может оказать неоценимую помощь Web-мастеру в ходе разработки и обслуживания Web-сайтов. Популярность языка PERL продолжает расти по ряду причин:

PERL является межплатформенным языком, максимально идентично поддерживаемым в разных операционных системах, отличаясь только в нескольких неизбежных деталях (например, число байтов, для представления длинного целого); PERL обладает определенными графическими возможностями за счет взаимодействия с модулем Tk.pm; PERL замечательно стыкуется с базами данных и серверами, работающим по технологии WINDOWS OLE AUTOMATION, с другими процессорами и т.д.

PERL - очень мощный язык, но слишком высоки системные издержки во время вызова программы на каждый запрос страницы, особенно WINDOWS. В последнее годы в качестве альтернативы быстро набирает популярность язык программирования Web-приложений PHP(Personal Home Page), работающее на очень многих серверах в Интернете. Как и все процедурные языки, PHP можно разделить на собственно язык и библиотеку функций. Существует большое количество инструментальных средств для PHP, интерфейсы ко всем популярным СУБД, почтовым протоколам, графическим файлам и т.д.

Технологии JAVA стремительно набирают популярность. Две самые необходимые вещи, без которых программирование на JAVA является невозможным, - это JAVA SOFTWARE DEVELOPE KIT(SDK) и документация

к нему. Стандартная библиотека содержит несколько тысяч классов и десятки тысяч методов, и запомнить даже самые часто употребляемые из них можно лишь после пары лет работы программистом. Но тем не менее, документация JAVA - одна из самых удобных в использовании, и работать с ней одно удовольствие.

Сеть Интернета представляет собой гигантский набор информации, основанной на гипертекстовых документах. Любой гипертекстовый документ представляет собой обычный текстовый файл в формате ANSI ASCII, содержащий собственно текст и специальные теги для его разметки, а также ссылки на другие подобные документы, графические изображения и любые иные файлы. Когда браузер – программа просмотра гипертекста – загружает подобный файл, все теги обрабатываются браузером в соответствии с правилами специального языка – HTML (Hyper Text Markup Language – Язык Разметки Гипертекста).

Неудачный, так и не реализованный стандарт языка HTML версии 3.0 сменил стандарт 3.2. В его разработке участвовали такие компании, как Netscape Communications Corporation, Microsoft, IBM, Novell, SoftQuad, Sun Microsystems и Spyglass.

На сегодняшний день выпущен еще один стандарт HTML версии 4.0, в который добавлены многие полезные расширения.

Следует отметить, что HTML, вопреки распространенному заблуждению, ни в коей мере не является языком программирования. HTML – это типичный язык разметки, т.е. с его помощью можно оформлять документы, создавать ссылки, но ни как не писать программы. Даже специальные эффекты, которые можно увидеть на страницах Web, создаются не с помощью HTML, а с использованием дополнительных

средств – например, встроенных в документ программ на языке JavaScript.

Основы языка разметки гипертекста (HTML).

Создание Web-сайтов реализуется с помощью языка разметки гипертекстовых документов HTML (Hyper Text Markup Language). Технология HTML состоит в том, что в обычный текстовый документ вставляют управляющие символы (тэги) и в результате получают Web-страницу. Браузер при загрузке Web-страницы представляет ее на экране в том виде, который задается тэгами.

Некоторые тэги имеют атрибуты, определяющие свойства тэга. Атрибут — это имя свойства, которое может принимать определенные значения.

Для создания Web-страниц служат простейшие текстовые редакторы, которые не включают в создаваемый документ управляющие символы форматирования текста. В качестве такого редактора в Windows можно использовать стандартное приложение Блокнот.

HTML-код страницы помещается внутрь контейнера. Без этих тэгов браузер не в состоянии определить формат документа и правильно его интерпретировать. Web-страница разделяется на две логические части: заголовок и содержание. Заголовок Web-страницы заключается в контейнер и содержит справочную информацию о странице, которая не отображается браузером, а также название документа.

Название Web-страницы содержится в контейнере и выводится в строке заголовка браузера.

Набор символов

Формально, набор символов, используемых в HTML документе должен:

- включать ISO Latin 1, известную также как ISO 8859-1 кодировку, так как она принадлежит к набору стандартов ISO 8859

- быть совместимым с ISO 10646 и Unicode.

В практической работе Вы должны **использовать**

только ISO Latin 1 набор символов. Сейчас и в ближайшем будущем Вы можете твердо рассчитывать на его обширную поддержку приложениями. Поддержка ISO Latin 1 *должна существовать* во всех браузерах, однако иногда с этим существуют проблемы. Также Вы можете придерживаться ASCII набора символов, которые являются подмножеством ISO Latin 1, в особенности, если у Вас нет необходимости в написании символов с диакритическим знаком или символов, не входящих в английский алфавит (a - z).

Если Ваша *клавиатура* или текстовый редактор не позволяет Вам *вводить* (т.е. сразу печатать) некоторые ISO Latin 1 символы, например, а или ñ, Вы можете использовать символьные escape последовательности.

Некоторые предупреждения для тех, кто создает документы на микрокомпьютерах:

- Наборы символов DOS и Macintosh не совместимы с ISO Latin 1. Это касается использования любых символов вне набора ASCII, так как для них необходимо преобразование. Некоторые программы могут делать эти преобразования автоматически, однако в конверсионных таблицах могут быть ошибки.

- Набор символов Windows наиболее согласован с ISO Latin 1, однако есть некоторые кодовые позиции, которые зарезервированы в качестве управляющих символов в ISO Latin 1 и, тем не менее используются для изображения видимых символов в наборе символов Windows. Наиболее известные из них - два различных тире "en тире" и "em тире", которые не надо смешивать с дефисом (-) или подчеркиванием (_), принадлежащими к ISO Latin 1 (и даже к ASCII). Если Вы используете такие символы, пользователи Windows систем вероятно увидят их как положено, однако на всех других системах символы скорее всего будет выглядеть, как помарки. (Обычно, такие символы даже не выводятся совсем).

ТЕГИ HTML

Тег HTML состоит из следующих друг за другом в определенном порядке элементов:

- левой угловой скобки < (такого же, как "меньше чем" символа)
 - необязательного слэша /, который означает, что тег является *конечным тегом, закрывающим некоторую структуру*. Таким образом, в этом контексте Вы можете читать символ /, как *конец...*
 - имени тега, например TITLE или PRE
 - необязательных, если даже тег может иметь их, атрибутов. Тег может быть без атрибутов или сопровождаться одним или несколькими атрибутами, например: ALIGN=CENTER
 - правой угловой скобки > (такой же, как символа "больше чем").
- Примеры:
- ```
<H1>
<H1 ALIGN=LEFT>
```

## ЭЛЕМЕНТЫ HTML

Большинство, но не все теги HTML спарены так, что за открывающим тегом следует соответствующий закрывающий тег, а между ними содержится текст или другие теги, например:

```
<H1>Foreword</H1>
```

В таких случаях два тега и часть документа, отделенная ими, образуют блок, называемый **HTML элементом**. Некоторые теги, например <HR>, являются элементами HTML сами по себе, и для них соответствующий конечный тег неверен. Далее мы будем называть теги по их именам, опуская обязательные угловые скобки.

## АТРИБУТЫ

Для каждого тега определяется множество возможных **атрибутов**. Большинство тегов допускает один или несколько атрибутов, однако атрибутов может и совсем не быть. **Спецификация атрибута** состоит из расположенных в следующем порядке:

- имени атрибута, например WIDTH
- знака равенства (=)

- значения атрибута, которое задается строкой символов, например, "80".

Всегда полезно заключить **значение атрибута** в кавычки, используя либо одинарные ('80'), либо двойные кавычки ("80"). Строка в кавычках не должна содержать такие же кавычки внутри себя. Так, если дата заключена в двойные кавычки, используйте одинарные кавычки для последующего заключения в кавычки, и наоборот. Предпочтительно использование двойных кавычек, так как для глаза человека бывает трудно отличить одинарные кавычки от символов, подобных символам акцентирования.

Вы можете также опустить кавычки для значений атрибутов, которые состоят только из следующих символов (обратитесь к технической концепции имени):

- символов английского алфавита (A - Z, a - z)
- цифр (0 - 9)
- промежутков времени
- дефисов (-)

Таким образом, WIDTH=80 и ALIGN=CENTER - разрешенное сокращение для WIDTH="80" и ALIGN="CENTER". Ссылка на URL, например, HREF=foo.htm, допустима, однако, **когда URL используется с атрибутами, он должен быть заковычен**, например HREF="http://www.hut.fi/". Существуют некоторые браузеры, которые допускают отсутствие кавычек или наличие элементов с открывающими кавычками без закрывающих. Однако, такую практику лучше не применять.

В пределах значения атрибута теги HTML не воспринимаются. А escape последовательности распознаются и интерпретируются, как символы.

Если значение атрибута такое же, как его имя, может быть использован **минимальный синтаксис** атрибута. То есть <UL COMPACT="COMPACT"> можно сократить до <UL COMPACT>. Некоторые агенты пользователей (программы просмотра пользователя, браузеры) даже требуют минимизации для некоторых атрибутов (COMPACT, ISMAP, CHECKED, NOWRAP, NOSHADE, NOHREF). Так что лучше использовать минимизированный синтаксис там, где это возможно.

Значения атрибутов должны быть отделены пробелами или незаполненными строками.

### Case чувствительность (чувствительность к выбору)

Что касается имен тегов, атрибутов и большинства значений атрибутов, HTML является **case нечувствительным языком**. Вы можете, например, написать TITLE, или Title, или title, или даже tItLE, если Вам нравится. (В этом документе используется написание на верхнем регистре. Это должно помочь читателю отличить HTML код от нормального текста.) Но существуют и **case чувствительные** конструкции языка, а именно:

- **escape-последовательности** (более официально называемые *символьными объектами*), которые начинаются знаком & (например, &lt;)

- **URL**, так как он может содержать наименования файлов, которые являются case чувствительными во многих операционных системах (например, в Unix). Для рассмотрения URL, смотрите <http://www.w3.org/pub/WWW/Addressing/Addressing.html>. Что касается синтаксиса URL, смотрите <ftp://ftp.funet.fi/pub/doc/rfc/rfc1738.txt> (полные URL записи) и <ftp://ftp.funet.fi/pub/doc/rfc/rfc1808.txt> (относительные URL записи).

**Разделение на строки и использование пробелов и символов табуляции.** Когда документ выводится на экран, пробелы и пустые линии не сохраняются, за исключением текста, заключенного в теги PRE (предварительно отформатированный текст). То есть любая последовательность пробелов, символов табуляции и пустых линий эквивалентна единственному пробелу в файле HTML. С другой стороны, пробел в файле HTML может быть представлен с использованием любого количества пробелов или новыми (пустыми) строками.

Термин *newlin* (новая линия) используется, чтобы ТЕОРЕТИЧЕСКИ обозначить конец строки. SGML определяет, что строка (запись) должна начинаться стартовым символом (newline - перевод строки, LF, ASCII код 10) и заканчиваться

символом конца записи (возврат каретки, CR, ASCII 13). На практике, HTML документы представляются и передаются с использованием newline представления, согласованного с данными компьютерными системами. Поэтому, браузеры HTML допускают поддержку любого из трех общепринятых представлений разделения строк, обозначаемых последовательностью CR LF, только CR, или только LF, и на основе такого допущения и выделяют ошибки представления начальных и конечных символов записи.

Таким образом, не имеет значения, как Вы разделите текст на строки, так как перевод строки эквивалентен пробелу. Заметим, однако, что Вы *не должны в HTML разделять слово на две строки*. Если Вы, например, разделили слово international на две строки, как это приведено ниже:

inter-

national

это будет интерпретироваться, как

inter- national

Таким образом, для логичного представления Вашего документа, Вы должны использовать такие теги HTML, как P или BR, чтобы при необходимости обеспечить перевод строки.

Браузеры обычно не разделяют слова на две строки, за исключением тех случаев, когда слово содержит дефис. Ссылочные спецификации HTML 3.2 не очень точны в отношении этого вопроса и при обсуждении раздела таблицы, приводится следующее:

Для некоторых агентов пользователя может быть необходимо или желательно перевести строку в пределах слова. В таких случаях желательно убедиться визуально, что это произошло.

Опасайтесь того, чтобы **длина линии вышла за границы Вашего контроля**. Это зависит от браузера, компьютера и установок, сделанных людьми, просматривающими Ваш документ. Вообще говоря, нет средств, чтобы *предотвратить* перевод строки между словами, но Вы можете *попытаться* предупредить перевод строки, используя непрерывные пробелы.

Что касается использования перевода строки в сочетании с тегами HTML, то есть специальные правила:

- Перевод строки, следующий сразу за начальным тегом, игнорируется. Например, строки

- `<P>`

- `Text`

эквивалентны строке

`<P>Text`

- Точно также, игнорируется перевод строки предшествующий конечному тегу. Например, строки

- `Text`

- `</P>`

эквивалентны строке

`Text</P>`

*Символ горизонтальной табуляции (HT)* можно использовать в документе HTML. Но в пределах элемента PRE символ табуляции имеет особую интерпретацию, эквивалентен пробелу и не несет в себе информацию о табуляции какого-либо вида. (Чтобы представить табулированные данные, используйте элемент table.) Практически лучше избегать включения символов табуляции в код HTML и использовать вместо этого соответствующее количество пробелов, если нужно отформатировать код источника HTML документа в табулированном виде.

### Классификация элементов

Способы, которыми теги HTML могут сочетаться, определяются терминами элементов и их классификацией. Гораздо удобнее определить, например, что элемент H1 может содержать только текстовые элементы, чем давать длинный список подходящих элементов, особенно, если учесть, что один и тот же список должен появляться во многих контекстах, и этот список может изменяться при добавлении новых текстовых элементов к будущим версиям HTML.

Элементы HTML можно классифицировать на три основные категории:

- **заголовочные элементы**, т.е. элементы, используемые в элементе HEAD ("шапка", заголовок) и

содержащие информацию о документе в целом: TITLE, ISINDEX, BASE, META, LINK, SCRIPT, STYLE

- **блоковые элементы**, включающие элементы, которые специфицируют *структуру документа*, например, разделение на части и параграфы (абзацы): H1, H2, H3, H4, H5, H6, ADDRESS, и элементы P, UL, OL, DL, PRE, DIV, CENTER, BLOCKQUOTE, FORM, ISINDEX, HR, TABLE

- **текстовые элементы**. *Текстовыми элементами могут быть:*

- простой текст, возможно содержащий escape последовательности (например, &amp;);
- выражения разметки: EM, STRONG, DFN, CODE, SAMP, KBD, VAR, CITE
- разметка шрифта: TT, I, B, U, STRIKE, BIG, SMALL, SUB, SUP
- специальные элементы: A, IMG, APPLET, FONT, BASEFONT, BR, SCRIPT, MAP
- элементы формы: INPUT, SELECT, TEXTAREA

Любой текстовый элемент, включая простой текст, может при необходимости появиться везде - путем задания абзаца (параграфа) (P элемент).

Правило "большого пальца", которое может помочь в запоминании того, какой элемент - блоковый, а какой - текстовый: блоковые элементы завершают абзац, текстовые - нет.

*Заметьте:* часто блоковые элементы могут содержать как текстовые, так и другие блоковые элементы, например, блоки могут быть вложенными. Текстовые элементы также могут быть вложенными. Но *текстовые элементы не могут включать блоковые элементы*. Например, выражение

```
<CITE><H3>Origin of Species</H3></CITE>
```

неверно (так как CITE - текстовый элемент, а H3 - блоковый элемент) и к тому же нелогично (Вы не можете придать заголовку, как *структуре*, значение цитаты, так ведь?). Тогда как запись

```
<H3><CITE>Origin of Species</CITE></H3>
```

была бы правильной, хотя отдельные браузеры воспринимают это с трудом. Еще один пример: не заголовок

(headings) вставляется в атрибут NAME тега A, а наоборот. Так же было бы неверно иметь завершение абзаца (тег P) в пределах, например, элемента STRONG. Хотя некоторые браузеры понимают это, семантика этого сомнительна, и Вы должны были бы использовать отдельные начальные и конечные теги STRONG в пределах каждого абзаца (если Вы действительно желаете выделить такие большие порции текста!).

### Разрешенная вложенность элементов

Этот раздел описывает, каким образом в HTML 3.2. элементы могут быть вложены. Здесь не приводятся правила упорядочения или обеспечения повторной вложенности элементов. Здесь просто даются ответы на вопросы в форме: *может ли элемент X появиться в пределах элемента Y?*

Та же самая информация представлена в описании отдельных тегов в частях *Допустимый контекст* и *Содержимое*. Здесь это представлено в сжатой форме и носит иллюстративный характер.

Заметки на память:

(Пояснения к списку)

- Текст со словами на верхнем регистре описывает элементы HTML.
- Слово, написанное на нижнем регистре - термин, описывающий сочетание HTML элементов.
- Элементы могут содержать внутри себя список других элементов, которые находятся в пределах этого элемента. Если нет такого списка, никакая вложенность элементов не разрешена. Исключения составляют элементы **block** и **text**. Для них допустимо такое содержание, как это описано в настоящем разделе.
- #PCDATA означает "символьные данные, используемые при грамматическом разборе" (содержит только escape последовательности, например, &uml;); теги HTML - не разрешены).
- body.content (body-содержание) означает: элементы, которые перечислены под элементом BODY.

HTML

TR

HEAD

TH, TD

TITLE, SCRIPT, STYLE	body.content
#PCDATA	ADDRESS
ISINDEX, BASE,	text
META, LINK	P
BODY	text
H1, H2, H3, H4, H5, H6	<b>text</b>
text	#PCDATA
<b>block</b>	TT, I, B, U, STRIKE,
P	BIG, SMALL, SUB, SUP
text	text
UL, OL, DIR, MENU	EM, STRONG, DFN,
LI	CODE, SAMP, KBD, VAR,
text	CITE
block	text
<i>(в пределах DIR или</i>	A
<i>MENU, элемент LI не</i>	text
<i>содержит блок)</i>	IMG
DL	APPLET
DT	text
text	PARAM
DD	FONT
text	text
block	BASEFONT, BR
PRE	SCRIPT
text <i>(без IMG, BIG,</i>	#PCDATA
<i>SMALL, SUB, SUP, FONT)</i>	MAP
DIV, CENTER,	AREA
BLOCKQUOTE	INPUT
body.content	SELECT
FORM	OPTION
body.content без FORM	#PCDATA
ISINDEX	TEXTAREA
HR	#PCDATA
TABLE	
CAPTION	
text	

Чтобы упростить описания элементов, автор использует

термин *текстовый контейнер*, чтобы обозначить любой элемент, который *может* непосредственно содержать текстовый элемент. Следующие элементы являются текстовыми контейнерами:

A, ADDRESS, APPLET, B, BIG, BLOCKQUOTE, BODY, CAPTION, CENTER, CITE, CODE, DFN, DIV, EM, FONT, FORM, H1, H2, H3, H4, H5, H6, HTML, I, KBD, LI, P, PRE (с ограничениями), SAMP, SMALL, STRIKE, STRONG, SUB, SUP, TD, TH, TT, U, VAR.

Нижеследующие элементы не являются текстовыми контейнерами, однако могут содержать текстовые элементы не непосредственно, а через элементы, которые являются текстовыми контейнерами:

DD, DIR, DL, DT, MENU, OL, TABLE, TR, UL.

Нижеследующие элементы совсем не включают текстовые элементы:

AREA, BASE, BASEFONT, BR, HEAD, HR, IMG, INPUT, ISINDEX, LINK, MAP, META, OPTION, PARAM, SCRIPT, SELECT, STYLE, TEXTAREA, TITLE.

Также автор вводит термин *блоковый контейнер*, чтобы обозначить любой элемент, который *может* непосредственно содержать блоковый элемент. Блоковыми контейнерами являются:

BLOCKQUOTE, BODY, CENTER, DD, DIV FORM HTML, LI (в пределах UL или OL), TD, TH.

### **Заметки о разном:**

об escape последовательностях (символьных объектах), именах, цветовых атрибутах, ширине, пикселях, вертикальном выравнивании (VALIGN), непрерывных пробелах (&nbsp;), комментарии

Этот подраздел рассматривает технические проблемы, связанные с некоторыми тегами HTML. *Рекомендуем пропустить его в первом чтении, чтобы вернуться потом при*

необходимости. Описания тегов содержат ссылки на дополнительную литературу.

### **Escape последовательности (символьные объекты)**

Escape последовательности или, точнее, символьные объекты, представляют собой метод представления специальных символов. Например, escape последовательность `&lt;` обозначает символ "меньше чем" (`<`).

Некоторые символы, такие как `<`, используются в HTML в специальном значении, поэтому должен существовать способ их выражения, как символьных данных внутри самого документа или в URL. Принято использовать следующие нотации (соответствия):

обычное имя символа (имена) символа	HTML запись символа (escape последовательность)
символ "меньше чем", левая угловая скобка	<code>&amp;lt;</code>
символ "больше чем", правая угловая скобка	<code>&amp;gt;</code>
амперсанд	<code>&amp;amp;</code>
двойные кавычки	<code>&amp;quot;</code>

Двойные кавычки (`"`) могут использоваться и в пределах обычного текста.

Заметим, что *точка с запятой* является частью escape последовательности, но в принципе она может быть опущена. Однако одни браузеры обработают такой код HTML, а другие могут и не понять.

В escape последовательности важен вариант написания символов, то есть различаются заглавные и строчные буквы. Например, амперсанд (`&`) не может быть представлен как `&AMP` (escape последовательность - неопределена). Escape последовательность `&auml;` и `&Auml;` - *a* умляют (*a* диерезис, написание *a* с двумя точками над ней) - обозначает два разных



нотацию <SUP><SMALL>TM</SMALL></SUP>. Нотация &shy; задает "мягкий дефис", который имеет неопределенную семантику в HTML. Нотация &nbsp; установленная для "условного прерывающегося пробела", отсутствует в ISO Latin 1 и, возможно, такой символ скоро вообще не будет использоваться.)

### **Имена**

При изучении HTML слово *имя* используется, как технический термин. (Вероятно, более соответствующим термином мог бы быть *идентификатор*, как бы в концепции грубого сходства с идентификатором в языках программирования). Имя представляет собой последовательность символов, содержащую только

- буквы английского алфавита (A - Z, a - z)
- цифры (0 - 9)
- промежутки времени
- дефисы (-)

и начинающуюся с буквы.

Концепция имени имеет место в описании атрибутов HTTP-EQUIV и NAME элемента [META](#) и в описании атрибута NAME элемента [PARAM](#).

Строка символов, которая используется для NAME иногда может содержать и другие символы, однако тогда она должна быть взята в [кавычки](#).

### **Цвета**

В элементах [FONT](#) и [BODY](#) некоторые конструкции HTML могут быть использованы, чтобы специфицировать цвета: например, цвет фона, цвет текста по умолчанию или цвет текста [связи](#).

Так как все мониторы и программы могут отображать цвет, действительное отображение может быть черно-белым или с различными оттенками серого.

Для спецификации цвета, как значения атрибута, существуют два варианта:

- Символьная нотация RED. Определено шестнадцать таких имен (см. ниже), которые можно записывать в любом регистре с кавычками или без.

- Цифровое обозначение в шестнадцатиричной записи, например, "#FF0000", которая контролирует, каким образом цвет формируется из основных цветов - красного, зеленого и голубого - в так называемое sRGB цветовое пространство, при этом обозначение должно быть взято в кавычки.

Конечно, символические нотации много легче и более понятны. С другой стороны численные обозначения дают больше возможностей.

Цвета были первоначально отобраны, как 16 цветный стандарт, поддерживаемый Windows VGA палитрой.

См. также:

- Шестнадцатиричные цветовые коды для BGCOLOR
- Установка фонового и текстового цветов Colors
- 216 цветов в Netscape

#### **Атрибут WIDTH (ширина)**

Значение атрибута WIDTH в тегах, например, HR или TABLE можно специфицировать двумя альтернативными путями:

- как *процентную часть* промежутка между текущими левой и правой границами; в этом случае значение атрибута должно быть взято в кавычки и за числом процентов должен сразу следовать знак процента, например, WIDTH="80%"
- в пикселях, при этом задается целое число (кавычки не требуются), например WIDTH=212.

Первый вариант относительной спецификации более предпочтителен, т.к. автор не может знать размера пикселя на экране читающей программы пользователя.

**Пиксели.** Используемые в атрибутах значения пикселя определяют пиксели *экрана*. Физический размер пикселя определяется монитором пользователя.

При выводе документа на прибор с очень высоким разрешением, например, лазерный принтер, браузеры должны умножать значения в пикселях на соответствующий коэффициент. То есть, если программа просмотра имеет дисплей с 75 пикселями на дюйм, а вывод должен осуществляться на лазерный принтер с 600 точками на дюйм, значение атрибута HTML в пикселях умножается на 8.

**Вертикальное выравнивание (VALIGN).** В некоторых тегах, связанных с [таблицами](#), может быть использован атрибут VALIGN для управления вертикальным выравниванием, т.е. позиционированием в вертикальном направлении ячеек таблицы.

Чтобы позиционировать содержимое ячеек в верхней части, середине или нижней части ячейки таблицы соответственно, задаются следующие значения атрибута VALIGN: TOP (верх), MIDDLE (середина), BOTTOM (низ).

По умолчанию VALIGN=MIDDLE.

**Непрерывный пробел (&nbsp;sp;) (Non-breaking spaces).** Последовательность &nbsp;sp; является [escape нотацией](#) для символа, который в других контекстах обычно называется непрерывным пробелом, или, для краткости, NBSP. Согласно [ISO 8859](#), этот символ должен быть представлен, как обыкновенный пробел (пропуск, пустое место), однако такой, который не заменялся бы переводом строки (как это часто бывает с обыкновенным пробелом в текстовой обработке). То есть для того, чтобы два слова были на одной строке с некоторым расстоянием между ними, между этими словами должны появиться непрерывные пробелы. (Действительная ширина промежутка между словами может меняться, и нет необходимости привязываться к определенному количеству пробелов в файле HTML.)

Вопрос о том, надо предупреждать перевод строки, когда осуществляется вывод HTML документа, не имеет однозначного ответа. [Спецификации HTML 2.0](#) говорят:

Использование знаков непрерывного пробела и указателя мягкого дефиса не рекомендуется из-за того, что их поддержка не получила широкого распространения.

Более того, хотя [HTML 3.2 Ссылочные спецификации](#) не вносят ясности в этот вопрос, в разделе описания атрибута NOWRAP элементов [TH](#) и [TD](#) указывается, что знак &nbsp;sp; должен действовать как непрерывный пробел, по крайней мере в пределах ячеек таблицы.

Если Вы используете непрерывные пробелы, используйте их *вместо* обычных пробелов, а не в дополнение к ним. Например, если Вы хотите предупредить перевод строки

между version и 3, наберите version 3 (но не version 3).

С другой стороны, в [таблице](#) в HTML 3.2 &nbsp; может иметь совершенно другое значение, которое может быть описано, как *непустое пространство*: когда таблица представлена с бордюром, ячейки без содержания (просто пробелы не учитываются) изображаются без бордюра, однако &nbsp; в разметке указывается!

### Комментарии

Файл HTML может содержать комментарии, дающие пояснения для человека, читающего HTML код. Комментарии не влияют каким-либо образом на представление документа, т.е. они игнорируются браузером.

Вы можете начать комментарии с четырехсимвольной последовательности `<!--` (знак "меньше чем", восклицательный знак, два дефиса) и завершить его трехсимвольной последовательностью `-->` (два дефиса, знак "больше чем"). Например: `<!-- Написано Южкой Корпела -->`.

Однако, обычно предпочитают включать информацию о документе в элементы HTML, например, [МЕТА](#), и также делать частью самого документа информацию о его цели, авторе, дате создания и последней модификации.

Таким образом, комментарии желательно делать только в редких случаях, например, для объяснения кода HTML или устаревших понятий. Помните, что комментарии является частью файла HTML и передаются вместе с документом. Поэтому, во избежании расширения файла, если Вы хотите много сказать, изложите все это в отдельном документе, а в текст вставьте только URL документа.

Редакторы и конвертеры HTML часто сами вставляют несколько комментирующих строк в начало HTML файла. Такие строки могут быть полезны, и их не стоит убирать.

### Контрольные вопросы:

1. Для чего используются языки программирования для Интернет?
2. Для чего используется HTML?

3. Структура программы HTML?
4. Каковы требования к оформлению тегов?
5. Как выделить теги из общего текста описания страницы?
6. Поясните термины открывающий тег и закрывающий тег?
7. Назовите основные теги языка HTML и их назначение?

### Лекция № 10.

#### Современные тенденции развития информационных технологий

Президент республики Узбекистан И.Каримов в своих выступлениях всегда подчеркивает подготовка кадров по информационным технологиям, также внедрение Интернет технологий во все сферы является актуальной задачей в настоящее время.

В этом направлении ставятся следующие задачи: создание новых учебных пособий, современных педагогических и информационных технологий, переподготовка кадров с высшим образованием, подготовка педагогов для обучения студентов лицея, колледжей.

В нашей республике в настоящее время проводятся ряд мероприятий по развитию компьютеризации, эффективному и качественному применению высоких компьютерных технологий, по расширению рынка компьютерных технологий и т.д.

При переходе на новые экономические механизмы развитие всех технологий производится посредством развития возможностей информационных технологий. Во главе иностранных фирм действующих в нашей республике стоит американская компания **COMPUTERLAND**. В задачи фирм входят не только продажа компьютеров и техники, также разработка научно-практического проекта вместе с нашими специалистами. На сегодняшний день используются компьютеры **IBM, COMPAQ, HEWLETT PACCARD, EPSON, DAEWOO, SONY** фирм.

В развитии расширения и развития деятельности

иностранных фирм «Ўзэкспомарказ» играет большую роль. Каждый год проводятся научно-практические семинары, народные выставки с помощью этого центра. Практическое значение этих мероприятий заключается в том, что они помогают получать совершенствованные знания в этой области.

В настоящее время одним из тенденций развития информационных технологий является развитие мобильных средств коммуникаций и сотовой связи. Мобильными системами связи называют такие сети, которые обладают различными комбинациями мобильности: **Терминальная мобильность** - возможность МС получать услуги связи при движении и способность сети идентифицировать, определять местоположение и сопровождать терминал. **Персональная мобильность** - возможность пользователя получать услуги связи (прием и посылку вызова) с любой МС на базе персонального идентификатора и способность сети обеспечить эти услуги, в соответствии с потребностями пользователя. Персональная мобильность подразумевает способность сети определять МС пользователя с целью операции, сопровождения и выполнения вызова. В настоящее время в России получили применение следующие мобильные системы: -транкинговые систем - сотовые системы -системы персонального радиовызова (пейджеры) -системы бесшнуровой телефонии -глобальные спутниковые системы.

**Сотовые системы связи** В настоящее время ведется интенсивное внедрение сотовых сетей связи общего пользования. Такие сети предназначены для обеспечения подвижных и стационарных объектов телефонной связью и передачей данных. В сотовых сетях подвижными объектами являются либо наземные транспортные средства, либо непосредственно человек, находящийся в движении и имеющий портативную абонентскую станцию (подвижный абонент). Возможность передачи данных подвижному абоненту резко расширяет его возможности, поскольку кроме телефонных сообщений он может принимать телексные и факсимильные сообщения, различного рода графическую информацию (планы местности, графики движения и т. п.), медицинскую информацию и многое другое. Особое значение сотовые

системы связи приобретают в связи с активным внедрением во все сферы человеческой деятельности персональных компьютеров, разнообразных баз данных, сетей ЭВМ. Доступ к ним через сотовую сеть позволит подвижному абоненту оперативно и надежно получить необходимую информацию. Соответственно возрастет и роль систем связи, повысятся требования к качеству передачи информации, пропускной способности, надежности работы. Увеличение объема информации потребует сокращения времени доставки и получения абонентом необходимой информации. Именно поэтому уже сейчас наблюдается устойчивый рост мобильных средств радиосвязи (автомобильных и портативных радиотелефонов), которые дают возможность сотруднику той или иной службы вне рабочего места оперативно решать производственные вопросы. Радиотелефон перестал быть символом престижа и стал рабочим инструментом, который позволяет более эффективно использовать рабочее время, оперативно управлять производством и постоянно контролировать ход технологических процессов, что обеспечивает дополнительные доходы при использовании радиотелефона в производстве. Внедрение сотовых сетей во многие отрасли народного хозяйства позволит резко повысить производительность труда на подвижных объектах, добиться экономии материально-трудовых ресурсов, обеспечить автоматизированный контроль технологических процессов, создать надежную систему управления транспортными средствами или мобильными роботами, распределенными на большой территории и входящими в состав гибких автоматизированных систем управления. Свое название сотовые сети получили в соответствии с сотовым принципом организации связи, согласно которому зона обслуживания (территория города или региона) делится на большое число малых рабочих зон или сот в виде шестиугольников. В центре каждой рабочей зоны расположена базовая станция, осуществляющая связь по радиоканалам со многими абонентскими станциями, установленными на подвижных объектах, находящихся в ее рабочей зоне. Базовые станции соединены проводными телефонными линиями связи с

центральной станцией данного региона, которая обеспечивает соединение подвижных абонентов с любыми абонентами телефонной сети общего пользования с помощью коммутационных устройств. При перемещении подвижного абонента из одной зоны в другую производится автоматическое переключение канала радиосвязи на новую базовую станцию, тем самым осуществляется эстафетная передача подвижного абонента от передающей к последующей (соседней) базовой станции. Управление и контроль за работой базовых и абонентских станций осуществляется центральной станцией, в памяти ЭВМ которой, сосредоточены как статические, так и динамические данные о подвижных объектах и состоянии сети в целом. В отличие от транкинговых систем в сотовых сетях подвижной связи радиосвязь базовой станции с абонентской станцией осуществляется в пределах малой рабочей зоны, что позволяет многократно использовать одни и те же частоты в зоне обслуживания. Число абонентов в сотовой сети определяется пропускной способностью и числом базовых станций, равным числу рабочих зон, которое возрастает по квадратическому закону с уменьшением радиуса рабочей зоны при постоянном радиусе зоны обслуживания. Если десять лет назад радиус рабочей зоны в сотовой телефонии был приблизительно равен 5-15 км, то в настоящее время в среднем он равен 200 м. Так уменьшение радиуса рабочей зоны с 30 до 0,5 км позволит увеличить в 3600 раз число подвижных абонентов, оснащенных радиосвязью и имеющих возможность выхода на телефонную сеть общего пользования. Следовательно, эффективность использования спектра радиочастот в сотовых сетях во много раз выше, чем в транкинговых системах, что позволит в перспективе обеспечить управление большим числом наземных подвижных объектов. С уменьшением радиуса рабочей зоны появляется возможность уменьшить мощность передатчиков и чувствительность приемников, что значительно улучшит электромагнитную совместимость абонентов в сотовых сетях и электромагнитную совместимость между сотовыми сетями и другими системами, использующими определенные спектры радиочастот, а также позволит снизить стоимость и габаритные размеры абонентской

станции, обеспечить доступ к базам данных и ЭВМ. Отмеченные преимущества позволяют уже в настоящее время повысить оперативность управления и контроля в работе подведомственных предприятий и организаций, улучшить качество технологических процессов в системах с большим числом транспортных средств. Стремительный рост объемов передаваемой информации требует значительного сокращения времени доставки и обработки абонентом необходимой информации. Это одна из причин быстрого роста мобильных средств связи на базе сотовых сетей. Внедрение сотовых сетей означает появление принципиально нового вида связи - массовой радиотелесвязи, т. е. нового вида услуг. Уже сейчас абонентский терминал сотовой сети - сотовый радиотелефон - признается многими зарубежными экспертами первичным терминалом, которым абонент пользуется как в стационарном состоянии (дома, на службе), так и в движении. Широкое внедрение портативных Сотовых Радио Телефонов в перспективе позволит обеспечить каждого человека персональным телефоном со своим индивидуальным номером. Создание систем массовой радиотелесвязи с большим числом подвижных абонентов, большой пропускной способностью и высоким качеством приема сообщений возможно только при использовании сотового принципа построения системы связи. Этим и объясняется повышенный интерес к сотовым сетям. Использование сотовой связи широким кругом потребителей в отраслях транспорта, связи, энергетики, строительства, сферы обслуживания, ремонта и др. приносит существенный экономический эффект.

**Электронный бизнес:** повышение эффективности бизнеса, основанное на использовании информационных технологий, для того чтобы обеспечить взаимодействие деловых партнеров и создать интегрированную цепочку добавленной стоимости. Понятие "электронный бизнес" шире понятия "электронная коммерция", касающегося только коммерческой деятельности, поскольку охватывает всю систему взаимоотношений с партнерами и заказчиками. **Электронная коммерция:** маркетинг, подача предложений, продажа, сдача в

аренду, предоставление лицензий, поставка товаров, услуг или информации с использованием компьютерных сетей или Интернета. Понятие "электронная коммерция" шире, чем "коммерция в Интернете", поскольку в него входят все виды электронной коммерческой деятельности. **Интернет-коммерция**, торговля в Интернете: коммерческая деятельность в Интернете, когда процесс покупки/продажи товаров или услуг (весь цикл коммерческой/финансовой транзакции или ее часть) осуществляется электронным образом с применением Интернет-технологий. Существует два класса систем для электронной коммерции: - "Бизнес-Бизнес" (Business-to-Business - B2B) и "Бизнес-Потребитель" (Business-to-Customer - B2C). К системам B2C относятся: web-витрина - оформленный web-дизайновскими средствами прайс-лист торговой компании, не содержащий бизнес-логики торгового процесса; Интернет-магазин, содержит кроме web-витрины всю необходимую бизнес логику для управления процессом Интернет-торговли (бэк-офис), а торговая Интернет-система (ТИС) представляет собой Интернет-магазин, бэк-офис которого полностью (в режиме реального времени) интегрирован в торговый бизнес-процесс компании. Интернет-торговля - только часть электронной коммерции, но очень бурно развивающаяся часть. Торговые операции через Интернет могут осуществлять многие организации - и производители товаров/услуг, и дистрибьюторы, и розничные торговые компании. Из известных трех типов систем Интернет-торговли: (web-витрины, Интернет-магазины и ТИС), в России практически нет ТИС, очень мало Интернет-магазинов, зато огромное количество web-витрин. Чем привлекательны и выгодны эти три типа систем для покупателей и продавцов? С точки зрения покупателя все три решения выглядят одинаково. Связано это с тем, что покупатель имеет дело с внешним оформлением любой системы, а это всегда web-каталог, система навигации и система оформления заказов. Практически предпочтения Покупателя зависят только от удобства использования web-каталога и системы навигации. Но как только покупатель начинает оформлять заказ, он убеждается в преимуществах Интернет-магазинов и ТИС. Преимущества эти проявляются в том, что покупателю могут предложить более

гибкую систему скидок, сразу выписать счет с учетом стоимости доставки и страховки. Кроме того, он сможет увидеть реальное состояние склада и получить информацию о прохождении своего заказа. Покупатель в первую очередь будет приобретать товары на сайтах тех компаний, которые предоставят лучшие цены и хороший сервис. Именно на этих конкурентных преимуществах строят свои планы по привлечению постоянной клиентуры Интернет-торговцы. С точки зрения продавцов эти три решения различаются весьма значительно. Web-витрина обходится торговым компаниям недорого, но: -web-витрина позволяет организовать только торговлю на заказ, наладить торговлю с реального склада практически невозможно; --ее использование не уменьшает затраты продавцов на содержание штата и операционные расходы; -web-витрина представляет собой очень неповоротливое решение с точки зрения управления и недостаточно гибкое с точки зрения организации маркетинговых акций; -имидж компании, открывшей и поддерживающей простую web-витрину всегда хуже, чем у компании, организовавшей Интернет-торговлю при помощи полнофункционального Интернет-магазина или ТИС. Но самое главное, организация Интернет-торговли при помощи web-витрины оказывается для торговой компании малоэффективным и даже часто нерентабельным делом. Интернет-магазин существенно более выгоден торговой компании (особенно среднего бизнеса), которая хочет реально управлять всем процессом Интернет-торговли и различными маркетинговыми акциями, торговать и на заказ, и со склада, уменьшить число менеджеров по продажам и т.д. На создание Интернет-магазина потребуется больше разовых затрат по сравнению с витриной, но они будут намного более эффективными, поскольку использование Интернет-магазинов существенно рентабельнее по обороту, чем использование web-витрин. При этом существует реальная альтернатива самостоятельному созданию громоздкого Интернет-магазина - аренда решения у специализированной компании. В этом случае большие разовые (и часто непроизводительные) затраты равномерно распределяются во времени.

**Виртуальные торговые площадки** Существует

распространенное мнение, что развитие Интернет покончит со множеством посредников. На деле происходит обратное, и, вместо сокращения количества прежних, Глобальная Сеть способствует появлению нового класса посредников. Возрастающие объемы B2B-коммерции приводят к возникновению e-marketplaces или виртуальных торговых площадок. Онлайн-торговая площадка - это место, где заключаются сделки между продавцом и покупателем, и осуществляется проведение финансово-торговых транзакций. Возможности Интернет позволяют совершать покупки/продажи в режиме реального времени, и, благодаря доступности Интернет, в торговой деятельности площадки могут участвовать компании из разных точек земного шара. Развитие торговых Интернет-площадок в перспективе (и, судя по всему, очень недалекой) позволит обеспечить более эффективный и свободный поток информации, товаров, платежей и других B2B услуг. По прогнозам аналитиков объем доходов от онлайн-торговых площадок в 2004 году достигнет 1300 млрд. долл. Принято выделять три типа виртуальных торговых площадок: - создаваемые покупателями (buyer-driven), - продавцами (supplier-driven или seller-driven) - третьей стороной (third-party-driven). Обычно возникновение тех или иных видов торговых площадок зависит от степени влияния покупателей и продавцов в данной области промышленности. Площадки типа buyer-driven. Одна или несколько крупных компаний создают свою торговую площадку для привлечения множества компаний-поставщиков. Эта концепция торговых площадок возникла в связи с потребностями крупных компаний в оптимизации процесса закупок, расширения торговых контактов и сети поставок. Площадки типа supplier-driven. Наряду с крупными покупателями крупные продавцы также играют активную роль в формировании торговых площадок. Это происходит по разным причинам. В частности, такие площадки играют роль электронного каталога/дистрибутора компаний производителей продукции данной крупной торговой организации. Торговые площадки типа third-party-driven, управляемые третьей стороной, призваны свести вместе покупателей и продавцов. Обычно такие площадки создаются теми, кто хорошо ориентируется в данном

секторе бизнеса и происходящих в нем бизнес-процессах. Начиная с нейтрального посредничества, многие подобные Интернет-площадки все более сближаются с ведущими компаниями на рынке, в некоторых случаях получая инвестиции в акционерный капитал. Однако это создает потенциальные проблемы с привлечением на рынок других компаний и контролем крупных промышленных инвесторов.

## Литература:

1. М.Арипов. Internet ва электрон алоҳа асослари (Ўзбек тилида). Тошкент, Университет, 2000, 132 б.
2. Стахнов Алексей. Linux. - 2-е изд.. - СПб.: БХВ-Петербург, 2005. - 944 с.: ил.. - (В подлиннике). - (Библиотека программиста №2; 2005/6).
3. Коппиен, Джеймс О. , Программирование на C++. Пер. с англ.. - СПб.: Питер, 2005. - 480 с.: ил.. - (Классика computer science). - (Библиотека программиста №2; 2005/4).
4. Шахов, М. , Photoshop CS и цифровая фотография/ М. Шахов, Т. Данилова, Ю. Гурский. - СПб.: Питер, 2005. - 336 с.: ил. + 8 цв. ил.. - (Популярный самоучитель). - (Библиотека программиста №6; 2005/5).
5. Бондаренко, С., Word 2003/ С. Бондаренко, М. Бондаренко. - СПб.: Питер, 2005. - 380, (4) с.: ил.. - (Популярный самоучитель). - (Библиотека программиста №4; 2005/4).
6. Иванов Виталий, Интернет для начинающих: Самоучитель. - СПб.: Питер, 2005. - 240 с.: ил.. - (Библиотека программиста №4; 2005/6). - (Самоучитель).
7. Отоцкий Юрий . Самоучитель Office XP. - СПб.: Питер, 2005. - 576 с.: ил., прил.. - (Самоучитель). - (Библиотека программиста №4; 2005/11).

## Содержание:

1. Лекция № 1.  
Автоматизация обработки данных.
2. Лекция № 2,3.  
Средства автоматизации научно-исследовательских работ. Работы с системой MathCad, MathLab.
3. Лекция №4.  
Системы автоматизированного проектирования
4. Лекция № 5.  
Компьютерные сети. Техническое обеспечение сетей.  
Локальная и глобальная сеть
5. Лекция № 6.  
Глобальная сеть Интернет. Использование сервисов Интернет в научной деятельности.
6. Лекция № 7.  
Вопросы компьютерной безопасности, методы её реализации.
7. Лекция № 8,9.  
Языки программирования в Интернете.  
HTML,PHP, JAVASCRIPT.
8. Лекция № 10.  
Современные тенденции развития информационных технологий
9. Литература

Сагатов Миразиз Ворисович

Каримова Дилбар Каримовна

Ирмухамедова Рано Мирзахитовна

Рашилов Шавкат Мугавеевич

Ганиева Тохира Иркиновна

Тошбекова Акида Абдухалиловна