Б.А. Гладких

ИНФОРМАТИКА

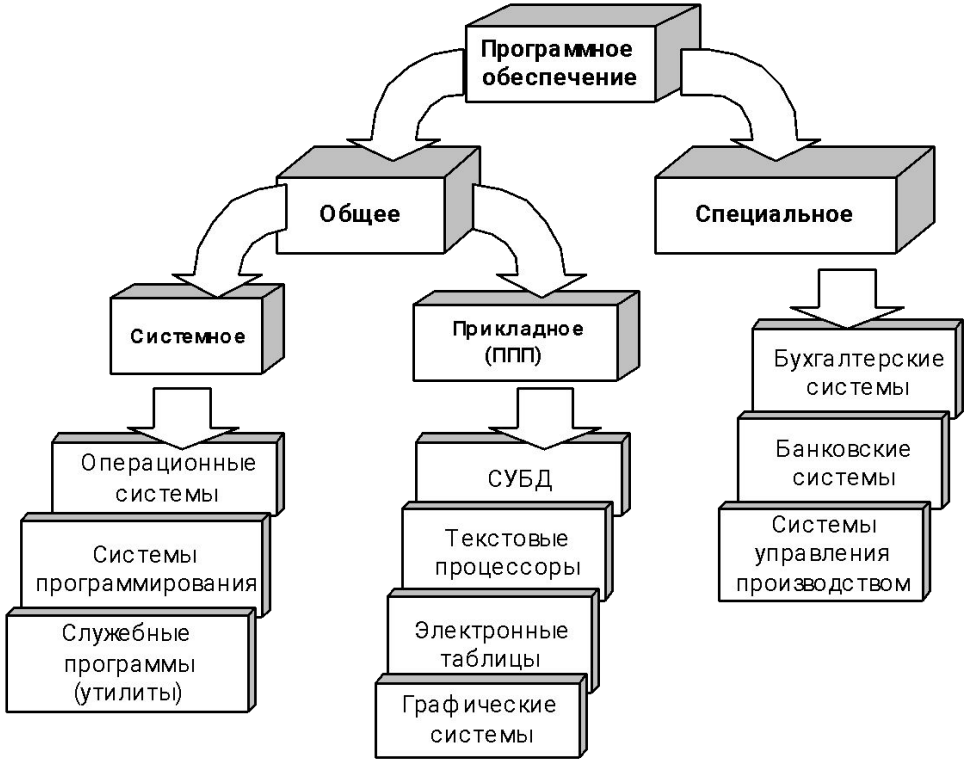
Историческое введение в специальность

Альбом иллюстраций

Глава 3.

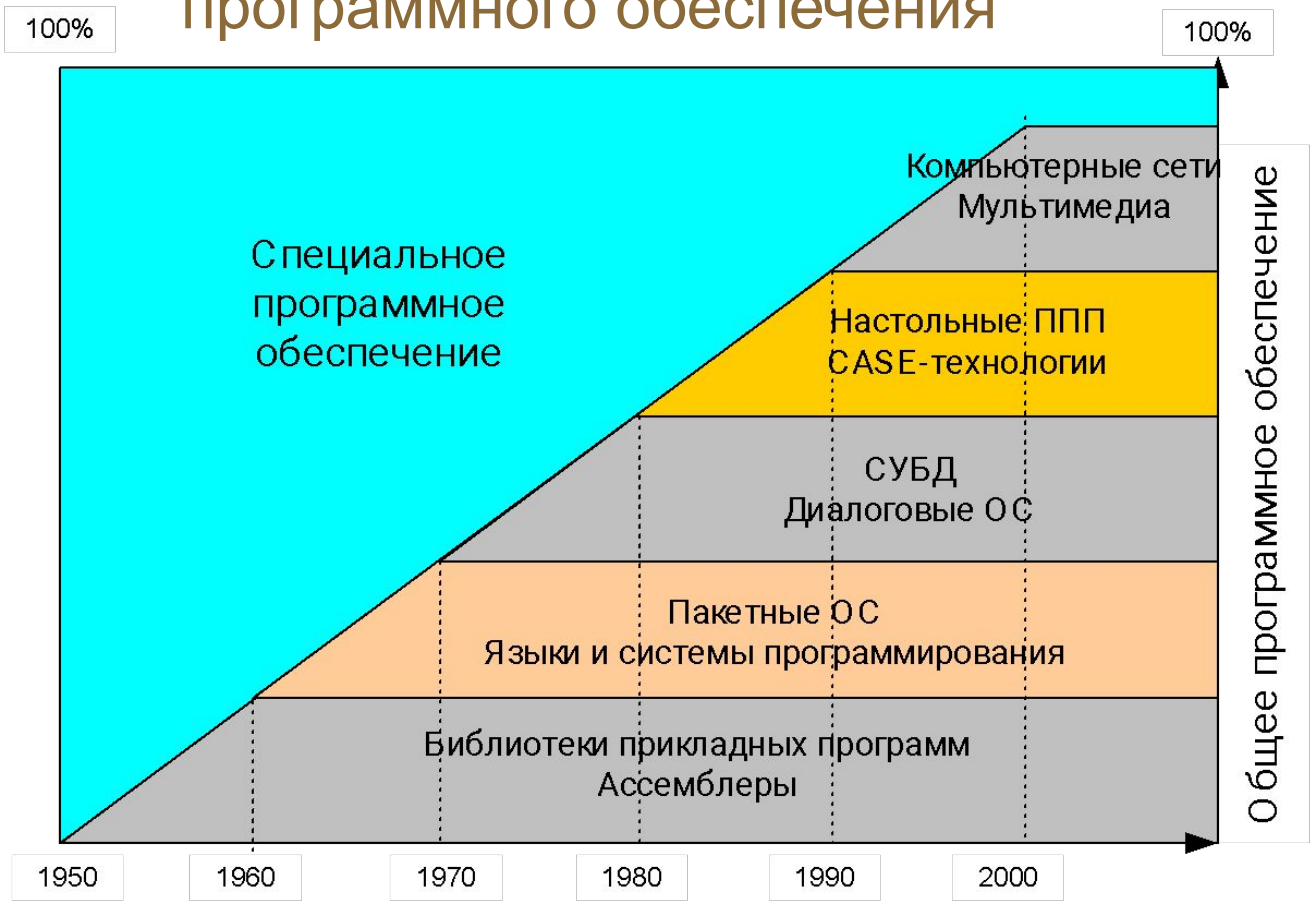
Программное обеспечение ЭВМ

3.1. Классификация и эволюция программного обеспечения



Классификация программного обеспечения

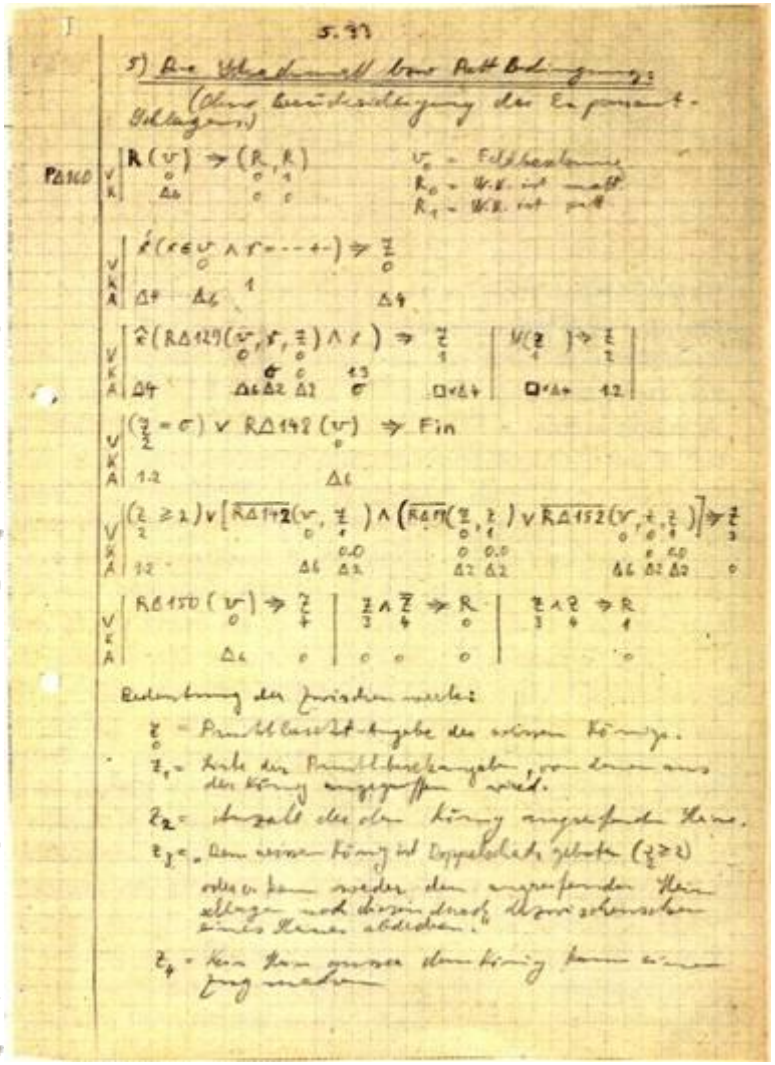
3.1. Классификация и эволюция программного обеспечения



Эволюция программного обеспечения. Подобно тому, как в океане из плавающей мути откладываются геологические пласты, из специального программного обеспечения с течением времени образуются слои общего ПО

3.2. Языки и системы программирования

Предыстория



Первая попытка создать высокоуровневый язык программирования принадлежит гениальному Конраду Цузе (конец 1940-х годов), разработавшему Plancalcul (планировщик вычислений).

«Plancalcul родился исключительно как результат теоретической работы, без всякой связи с тем, появится или нет в обозримом будущем машины, подходящие к программам на Plancalcul».

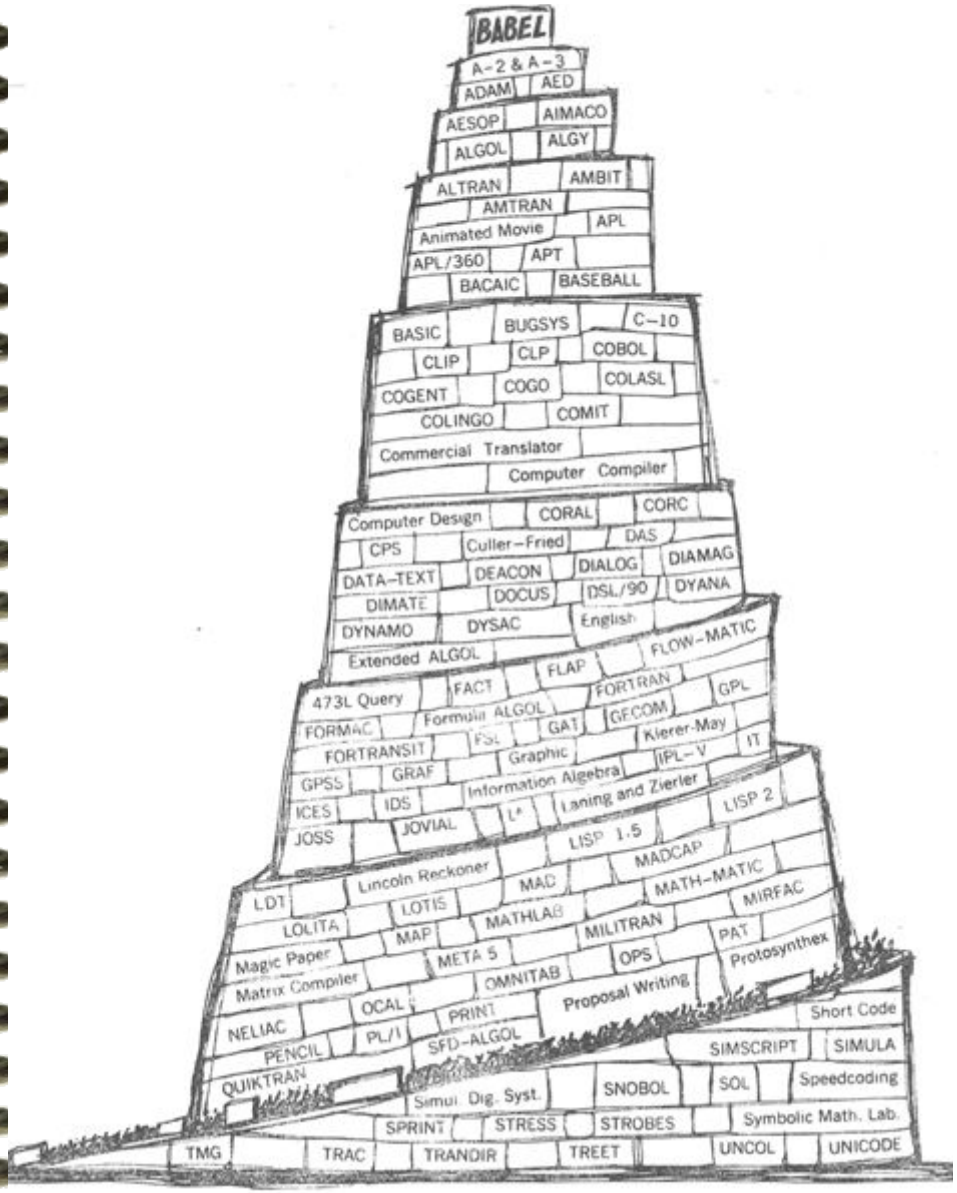
Фрагмент рукописи Цузе с шахматной программой на языке Plancalcul

3.2. Языки и системы программирования

Предыстория

Михаил Романович Шура-Бура и А.П. Ершов – создатели первых отечественных систем автоматизации программирования для ЭВМ «БЭСМ» и «Стрела»
(1954-1956 годы)

3.2. Языки и системы программирования

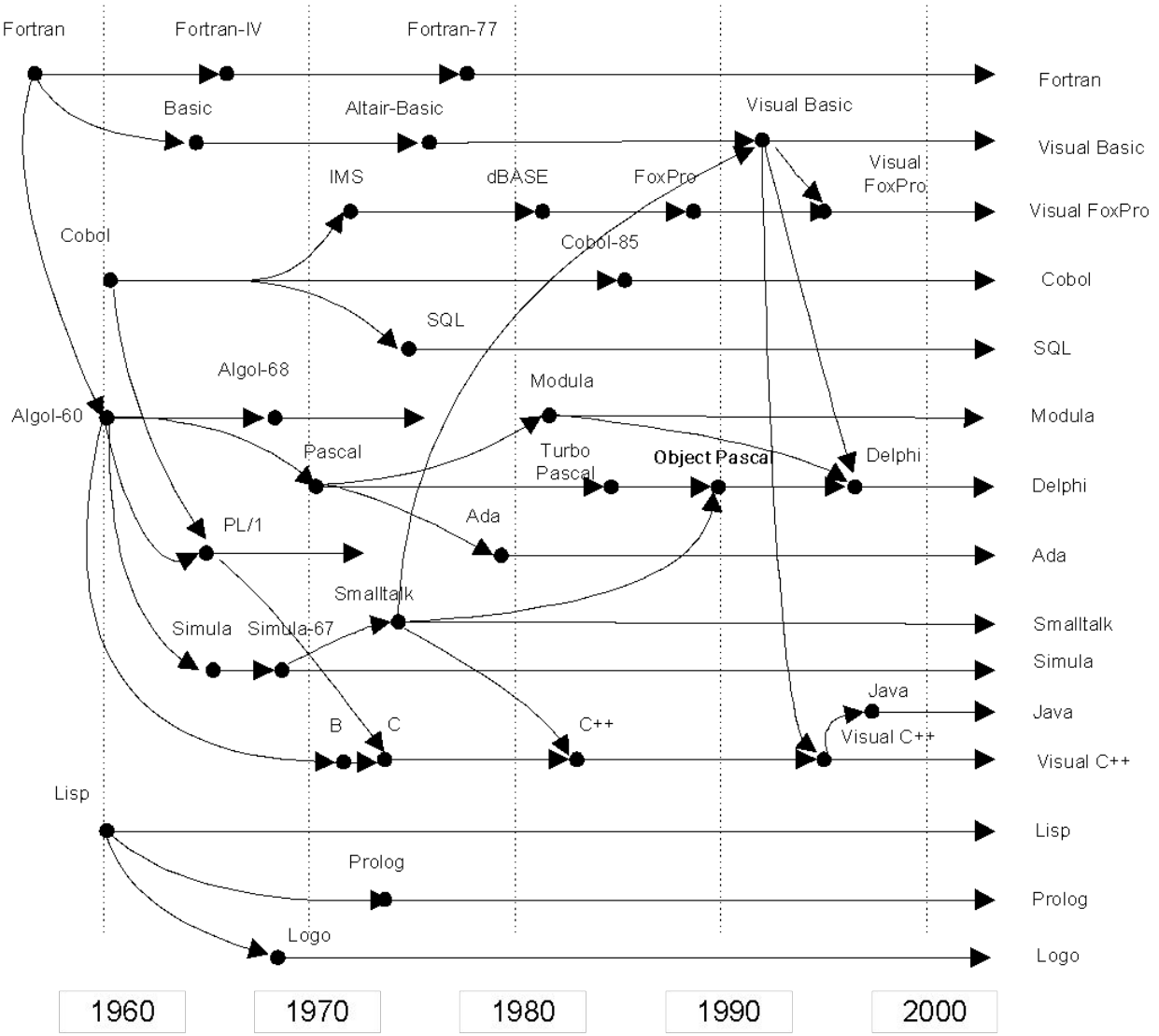


Наиболее активный период разработки языков и систем программирования приходится на 1960-е годы.

За это десятилетие в мире родилось более тысячи разнообразных языков, как универсальных, так и специализированных, но выжили и доросли до XXI века дожили немногие, в том числе бессмертные Fotran, Basic, Algol, Cobol, Simula, Lisp и их потомки.

На рисунке: «вавилонская башня» языков программирования, созданных в 1960-е годы

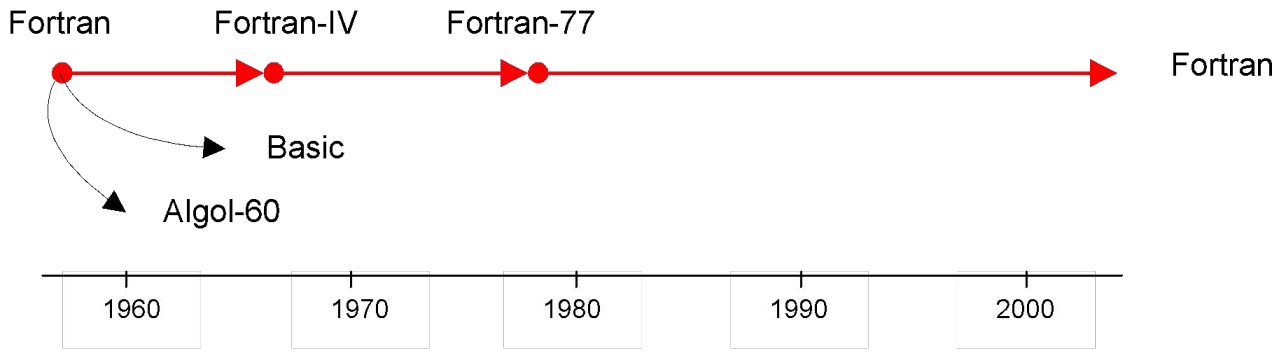
3.2. Языки и системы программирования



Родословная основных высокоуровневых языков программирования

3.2. Языки и системы программирования

Бессмертный Fortran



Fortran = FORmula TRANslator

Первый высокоуровневый язык программирования Fortran был разработан в фирме IBM под руководством Джона Бэкуса (Backus, John; p. 1924).

Работа над языком началась в 1954 г., первая реализация для IBM 704 в выполнена в 1957 г.

3.2. Языки и системы программирования

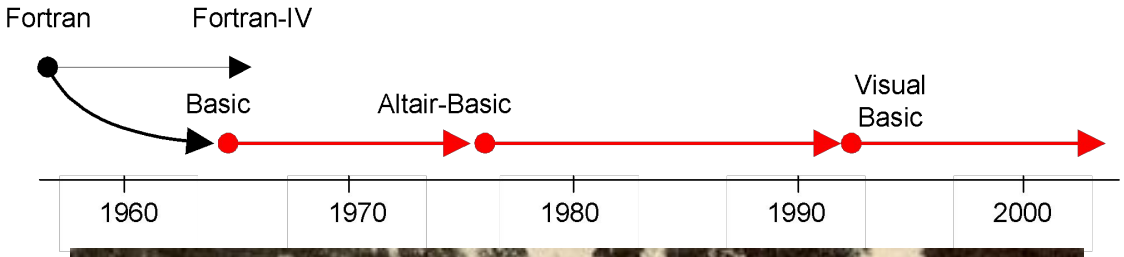
Бессмертный Fortran

```
C       MAIN PROGRAM
    101 FORMAT(208)
    102 FORMAT(//'N=',15, 5X, 'R=', 15
           1//6X, 'M', 5X, PROB)
    103     FORMAT(18, F14.10)
    201 READ(1,101) N, IR
        WRITE(3,102) N, IR
        IF(N) 202, 202, 203
    202 STOP
    203 IF(IR) 202, 202, 204
    204 M=O
        P=COMBF(N,M)*COMBF(IR-1,N-M-1)
        1/COMBF(N+IR-1,IR) ...
```

Фрагмент программы на языке Fortran

3.2. Языки и системы программирования

Basic – язык для начинающих



BASIC = Beginners All-purpuse Symbolic Instruction Code

Язык Basic был разработан в 1964 г. в Дармутском колледже в г. Хановере (Darmouth College, Hanover), штат Нью-Хемпшир

3.2. Языки и системы программирования

Basic – язык для начинающих

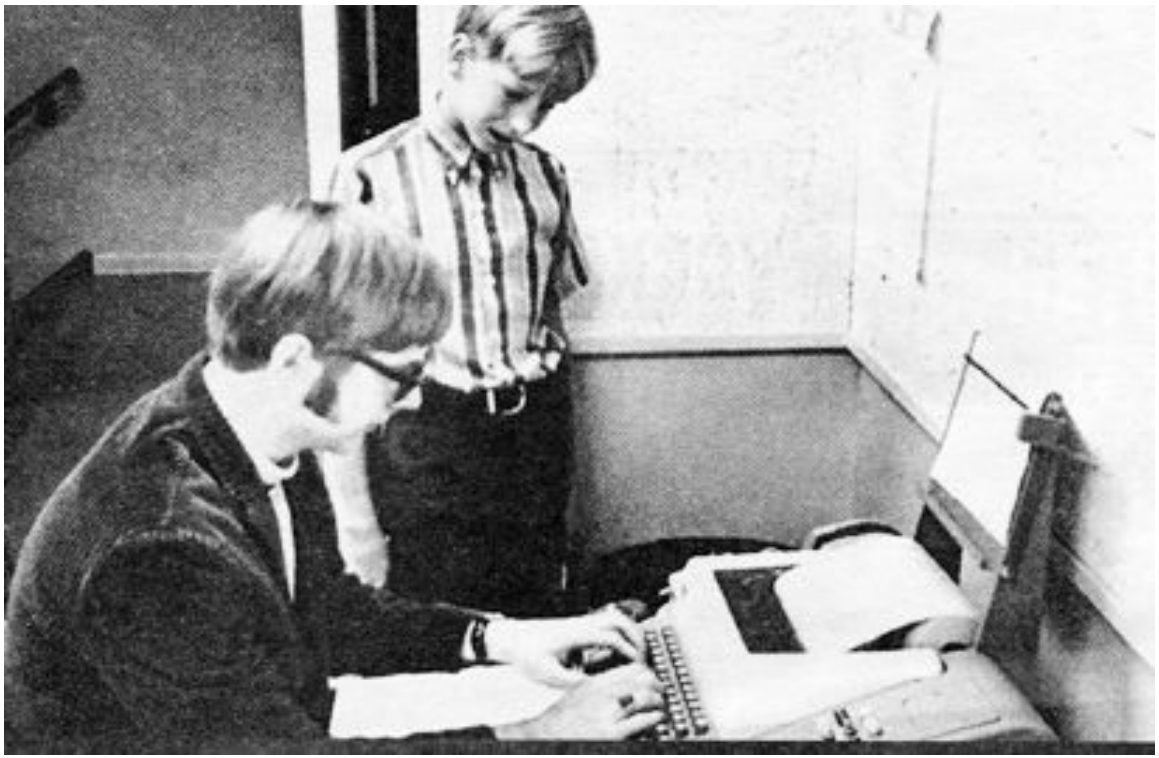
Авторы языка Basic.
Стоит Джон Кемени
(Kemeny, John G.; 1926-1993),
сидит Томас Курц
(Kurtz, Thomas E.; p. 1928)

```
10 dim A(5)
20 for i=1 to 5
30 input A(i)
40 next i
50 if i=5 then goto 140
60 if A(i)<=A(i+1) then goto 90
70 i=i+1
80 goto 130
90 z=A(i)
100 A(i)=A(i+1)
110 A(i+1)=z
120 i=1
130 goto 50
140 for i=1 to 5
150 print A(i)
160 next i
```

Простейшая программа на языке Basic

3.2. Языки и системы программирования

Basic – язык для начинающих



Будущие создатели Microsoft Пол Аллен (Allen, Paul; p. 1954) и Билл Гейтс (Gates, William; p. 1955) познакомились с Бэйсиком, работая в компьютерном классе школы в Сиэтле (снимок 1968 г.)

3.2. Языки и системы программирования

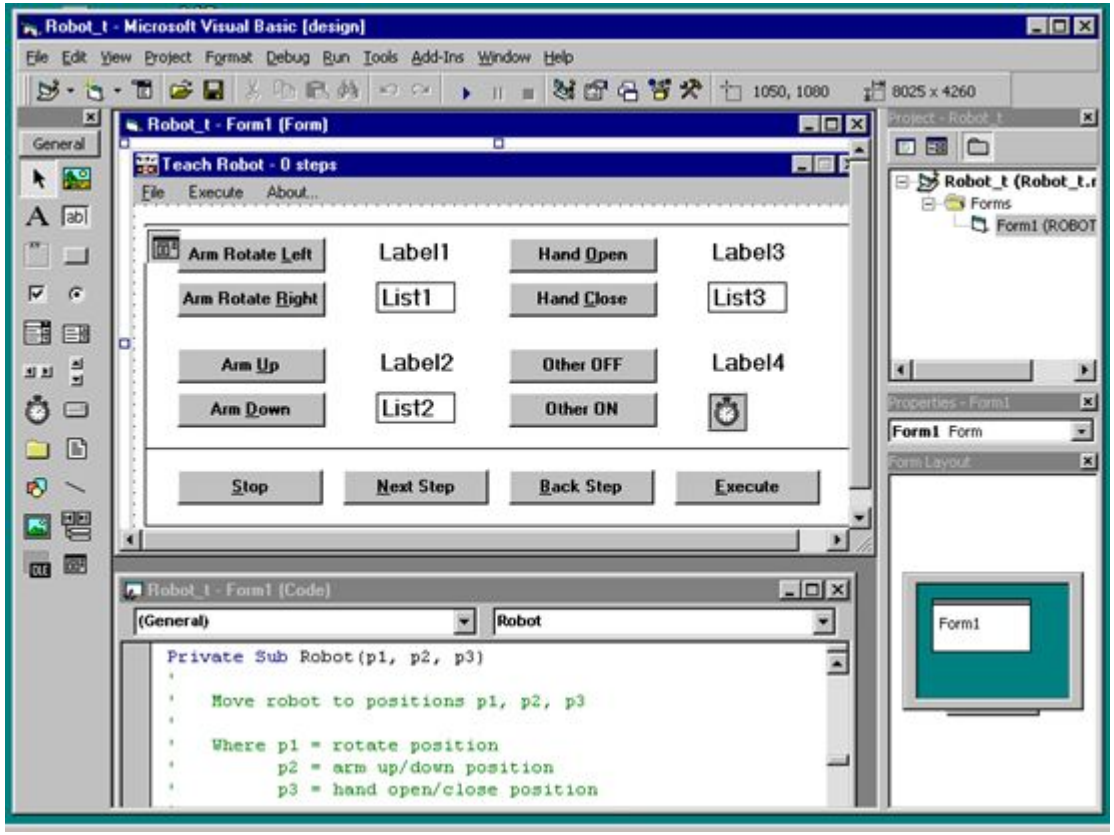
Basic – язык для начинающих

Начав с Бэйсика, компания Microsoft превратилась в крупнейшую софтверную империю, а Билл Гейтс – стал самым богатым человеком на планете

Штаб - квартира корпорации Microsoft в Редмонде (пригород Сиэтла)

3.2. Языки и системы программирования

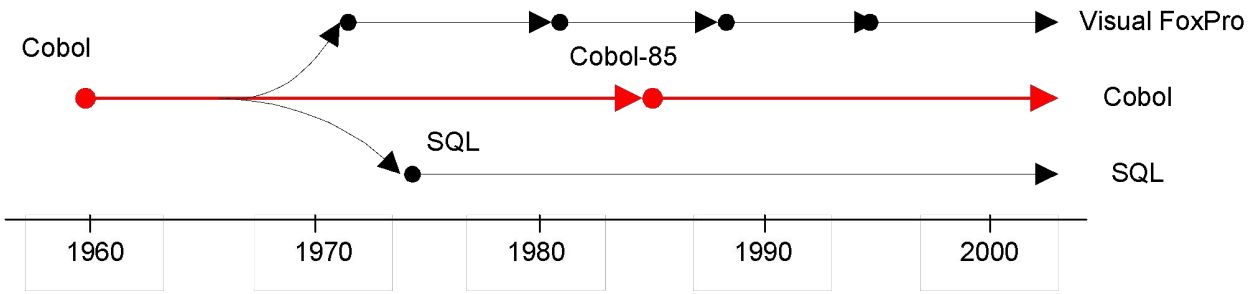
Basic – язык для начинающих



На протяжении нескольких десятилетий Visual Basic оставался фирменный языком компании Microsoft. В начале 1990-х годов он стал объектным и приобрел средства визуального проектирования

3.2. Языки и системы программирования

Cobol – язык для бухгалтеров



COBOL = COmmon Business-Oriented Language

На фото: разработчики языка Cobol у шуточного обелиска, присланного в их адрес в качестве намека на безнадежно медленную работу, способную похоронить саму идею. Справа внизу – Грейс Хоппер

3.2. Языки и системы программирования

Cobol – язык для бухгалтеров

Основные свойства языка Cobol:

- независимость программ от оборудования;
- независимость программ от данных;
- сложные структуры данных;
- синтаксис, приближенный к естественному английскому языку.

3.2. Языки и системы программирования

Cobol – язык для бухгалтеров

```
1010 IDENTIFICATION DIVISION.
1020 PROGRAM-ID “EXAMPLE”.
1030 ENVIROMENT DIVISION.
1040 INPUT-OUTPUT SECTION.
1050 FILE-CONTROL.
1060   SELECT CD ASSIGN TO “SYS010” UNIT-RECORD 2540R.
1070   SELECT TT ASSIGN TO “SYS009” UTILITY 2400.
1080 DATA DIVISION.
1090 FILE SECTION.
1100 FD     CD    DATA RECORD IS C
1110            LABEL RECORDS ARE OMITTED.
1120 01     C.
1130   02   C1  PICTURE 9(4).
1140   02   C2  PICTURE 9.
1150   02   C3  PICTURE X(70).
...
```

Программа на Коболе (начало)

3.2. Языки и системы программирования

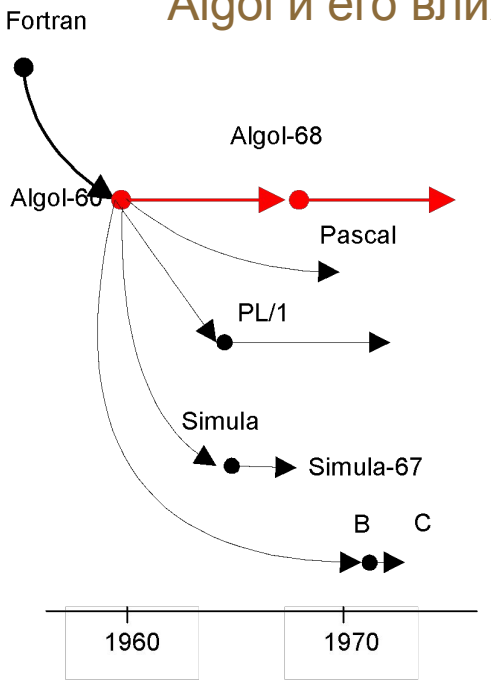
Cobol – язык для бухгалтеров

```
1290 PROCEDURE DIVISION.
1300 P1.   OPEN INPUT CD, OUTPUT TT.
1310 P2.   READ CD, AT END GO TO P3.
1320       MOVE C1 TO D1.
1330       MONE C2 TO D2.
1340       MOVE C3 TO D3.
1350       ADD C1, C2, GIVING D4.
1360       WRITE T FROM D.
1370       GO TO P2.
1380 P3.   CLOSE SD, TT.
1390       STOP RUN.
```

Программа на Коболе (окончание)

3.2. Языки и системы программирования

Algol и его влияние на языки программирования



ALGOL = ALGOritmic Language

В 1958 году в Цюрихе (Швейцария) состоялась международная конференция, предложившая проект нового универсального международного языка программирования Algol-58. В 1960 году на парижской конференции была принята окончательная версия под названием Algol-60.

На снимке: участники парижской конференции голосуют за Алгол-60.

3.2. Языки и системы программирования

Algol и его влияние на языки программирования

Основные свойства языка Algol-60:

- машинная независимость;
- формальный синтаксис;
- описание переменных и блочная структура;
- рекурсия

Нормальная форма Бэкуса-Наура (БНФ)

<цифра>::= 1|2|3|4|5|6|7|8|9|0

<целое без знака>::= <цифра>| <цифра> <целое без знака>

3.2. Языки и системы программирования

Algol и его влияние на языки программирования

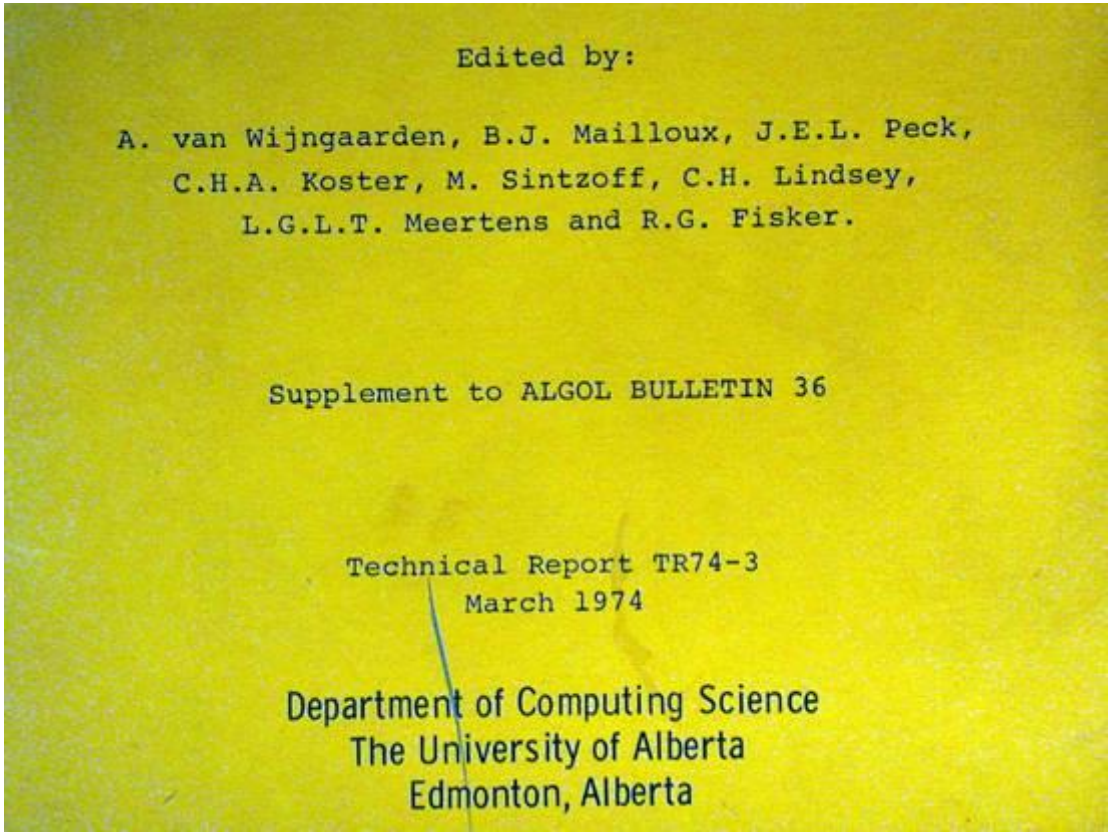
```
begin
    integer i, n;
    real s;
    real array x[1:n];
    s:=0;
    for i:=1 step 1 to n do
      s:=s+x[i];
    s:=s/n
end
```

Простейшая программа на Алголе-60, вычисляющая среднее арифметическое n чисел.

Синтаксис Алгола-60 сформировал стандарт для всех последующих языков программирования

3.2. Языки и системы программирования

Algol и его влияние на языки программирования



Edited by:

A. van Wijngaarden, B.J. Mailloux, J.E.L. Peck,
C.H.A. Koster, M. Sintzoff, C.H. Lindsey,
L.G.L.T. Meertens and R.G. Fisker.

Supplement to ALGOL BULLETIN 36

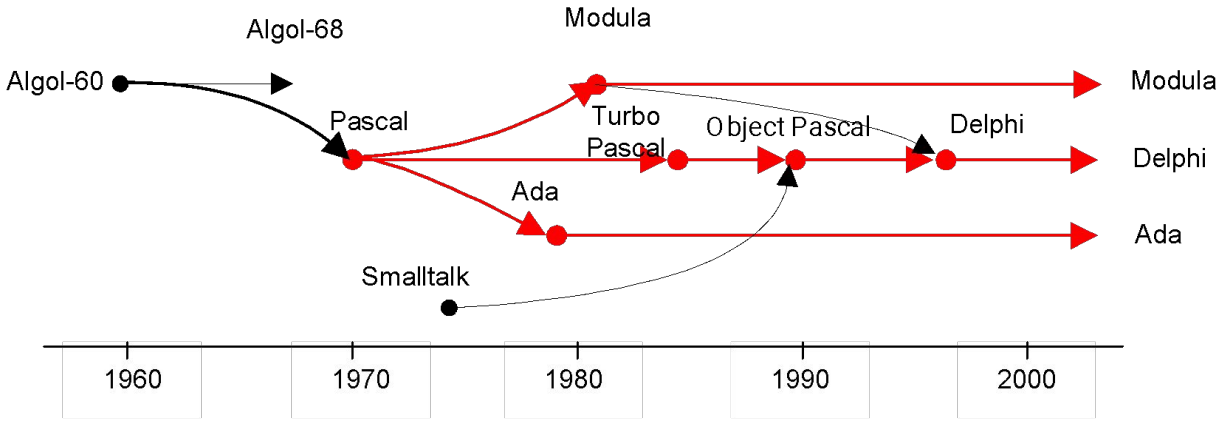
Technical Report TR74-3
March 1974

Department of Computing Science
The University of Alberta
Edmonton, Alberta

В результате многолетней переработки Алгола-60 комитетом IFIP появился язык Алгол-68 (пересмотренное сообщение под ред. А. ван Вейнгаардена (A. van Wijngaarden) и др. опубликовано в 1975 г.)

3.2. Языки и системы программирования

Pascal и его потомки



Член комитета по Алголу-68 Никлаус Вирт (Wirth, Niklaus; p. 1934) был против принятия переусложненного стандарта.

В знак доказательства своей правоты он разработал в 1971 г. простой и ясный алголоподобный язык, предназначенный прежде всего для обучения студентов в Федеральном техническом университете в Швейцарии. В честь изобретателя первой вычислительной машины Вирт назвал язык Паскалем.

3.2. Языки и системы программирования

Pascal и его потомки

```
var
        i, n: integer;
        s: float;
        x: array[1..n] of real;
begin
        s:=0;
        for i:=1 to n do
          s:=s+x[i];
        s:=s/n
end.
```

Программа на Паскале, вычисляющая среднее арифметическое n чисел

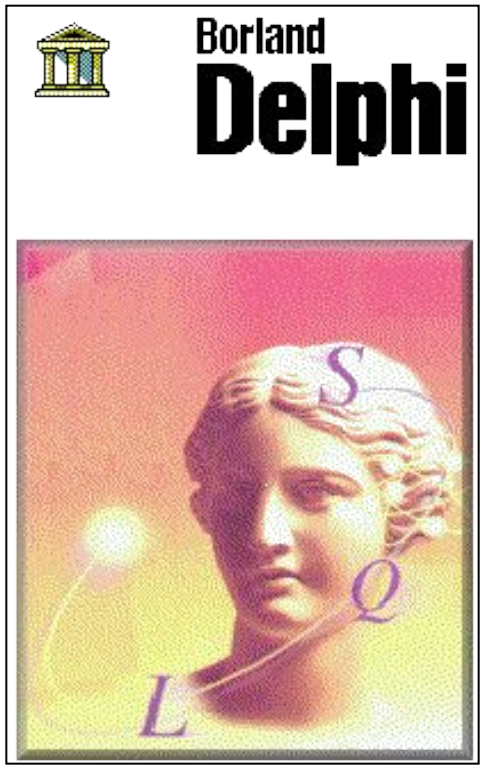
3.2. Языки и системы программирования

Pascal и его потомки

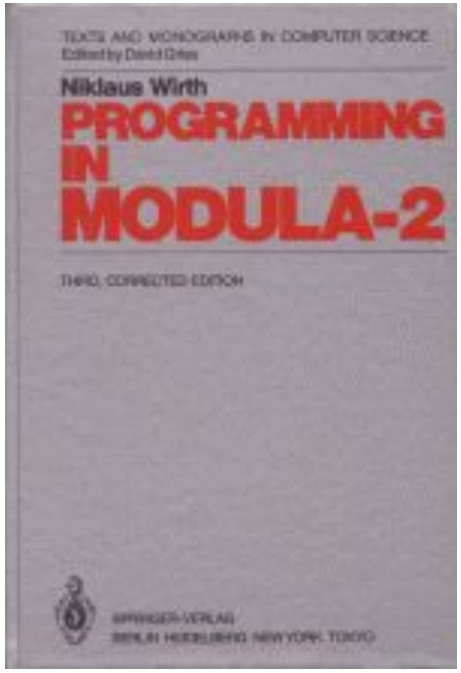
Новую жизнь языку Pascal дал Филипп Кан (Kahn, Philippe; p. 1938) – создатель компилятора Turbo Pascal для IBM PC и основатель компании Borland (1984 г.)

3.2. Языки и системы программирования

Pascal и его потомки



Среда разработки Delphi фирмы Borland объединила передовые достижения технологии программирования: объектное расширение языка Pascal, визуально-событийное проектирование, модульное структурирование и раздельная компиляция.



В отличие от учебного Паскаля, язык программирования Modula-2, предложенные Никлаусом Виртом, изначально предназначался для профессионального применения

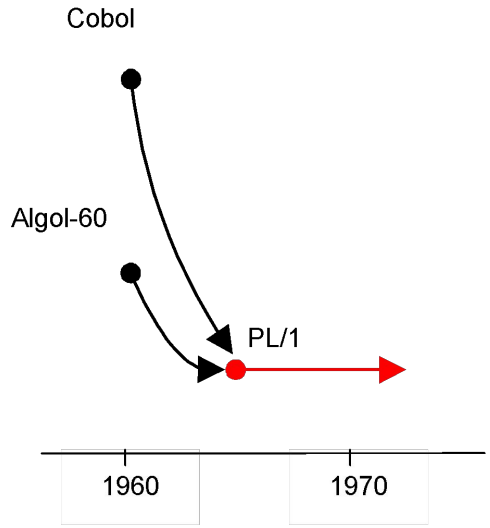
3.2. Языки и системы программирования

Pascal и его потомки

В 1975 году Министерство обороны США приняло решение разработать стандартный язык для программирования сложных и ответственных военных приложений. Был объявлен широкий международный конкурс, в котором приняли участие 15 групп разработчиков. В результате нескольких туров в мае 1979 года выявился победитель — французская фирма C.I.I., руководитель проекта Жан Ихбиа (Ichbiah, Jean).

Снимок сделан на II конференции по истории языков программирования, 1993 г.

3.2. Языки и системы программирования



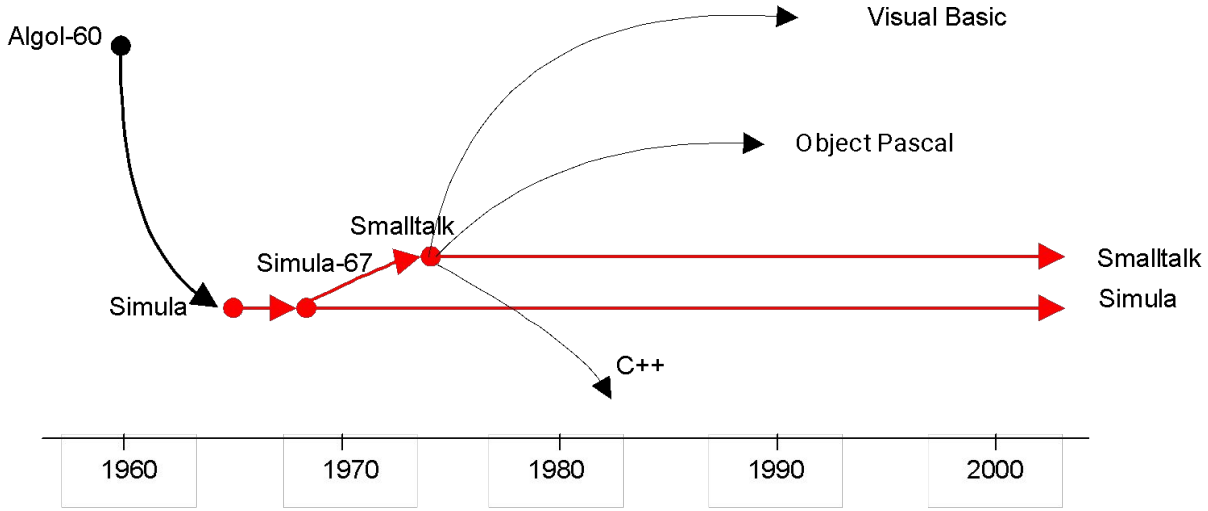
PL/1 = Programming Language One

Язык PL/1 был частью амбициозного проекта IBM S/360, он создавался в спешке и представлял собой механическую смесь идей из многих языков. Критики сравнивали его с елкой со множеством украшений.

```
EXAMPLE: PROCEDURE OPTIONS (MAIN);
ON ENDFILE (SYSIN) GO TO ENDING;
P1:     GET LIST (A, B, C);
        D = B*B — 4*A*C;
        E = —B/(A+A);
        IF D<0 THEN DO;
          X1, X2 = E;
          Y1 = SQRT(—D)/(A+A);
        END;
        ELSE DO;
          R = SQRT(D)/(A+A);
        …
        Y1 = 0;
        END;
        Y2 = —Y1;
        PUT LIST (X1, Y1, X2, Y2);
        GO TO P1;
ENDING:;
END EXAMPLE;
```

3.2. Языки и системы программирования

Simula и Smalltalk – революция в программировании



Simula = SIMULAlation

За разрабртку языка Simula Кристен Нигорд (Nygaard, Kristen; 1926-2002), на снимке слева, и Оле-Йохан Дал (Dahl, Ole-Johan; 1931-2002) были удостоены высшей награды компьютерного сообщества – медали Тьюринга

3.2. Языки и системы программирования

Simula и Smalltalk – революция в программировании

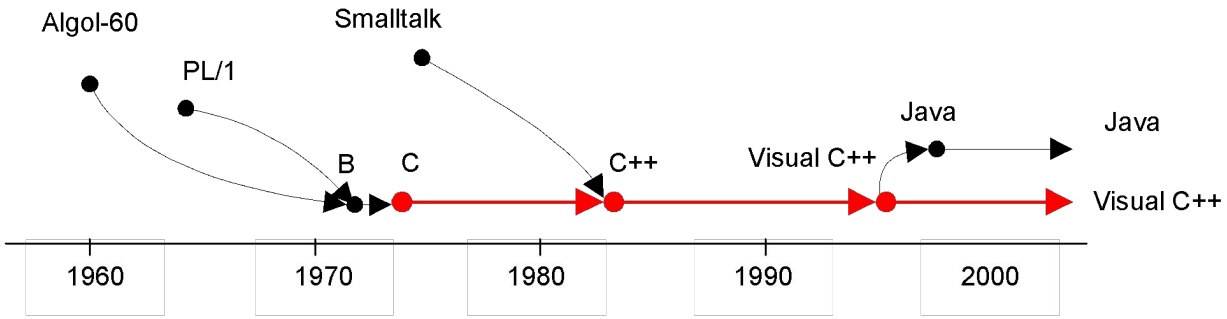
```
|a|
a := Array new: 5.
1 to: 5 do: [:i | a at: i put:
  (Prompter prompt: 'Введите элемент массива')
asNumber].
a := a asSortedCollection.
a do: [:i | Transcript putAll: i printString].
```

Простейшая программа на Smalltalk, вычисляющая среднее арифметическое пяти чисел

Алан Кей

3.2. Языки и системы программирования

C – язык для профессионалов



Язык Си (C) был создан Деннисом Ричи (Ritchie, Dennis M.; p. 1941) в 1973 году в Bell Labs в ходе разработки операционной системы UNIX. Он развивал язык Би (B), который основывался на созданном в Кембриджском университете языке BCPL (от Basic Combined Programming Language), который в свою очередь был потомком Алгола-60

3.2. Языки и системы программирования

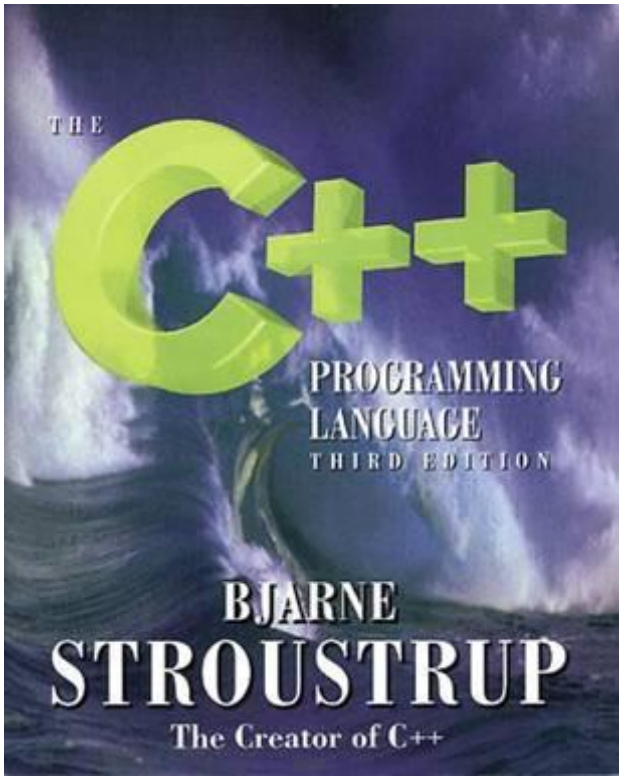
C – язык для профессионалов

```
float A[5];
  for(int i=0;i<5;i++)scanf("%f",&A[i]);
  i=0;
  while(i<4){
    if(A[i]<=A[i+1])i++;
    else{
      z=A[i];
      A[i]=A[i+1];
      A[i+1]:=z;
      i=0;
    }
  };
  for(i=0;i<5;i++)printf("%f\n",A[i]);
```

Текст на языке C отличается лаконичностью

3.2. Языки и системы программирования

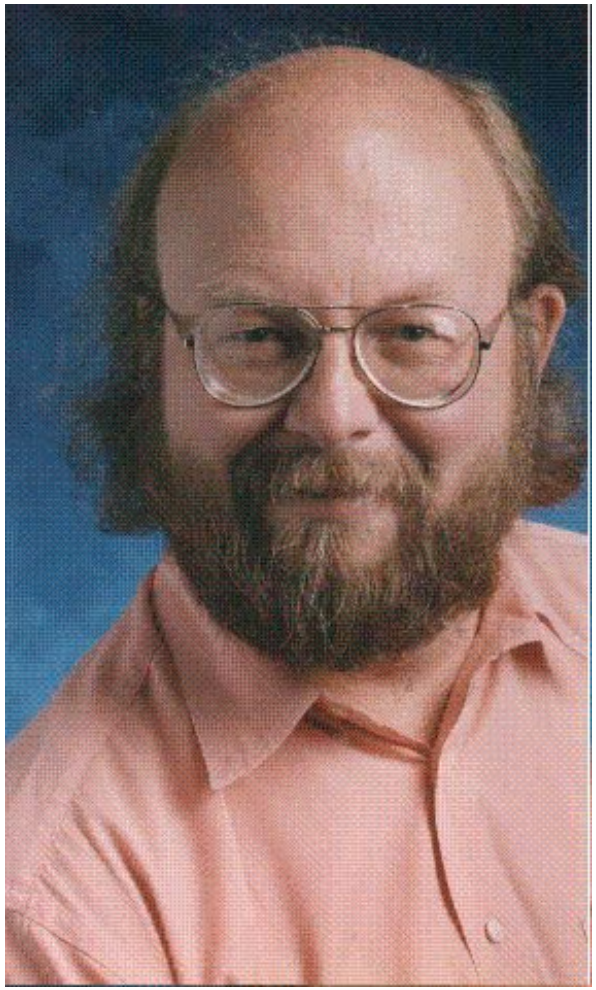
C – язык для профессионалов

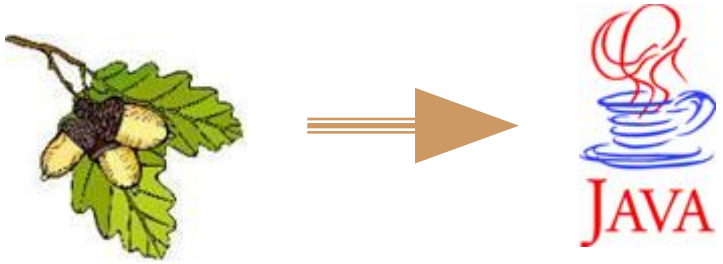


Бьярн Страуструп (Stroustrup, Bjarne; p. 1950) ввел в язык C объекты и превратил его в C++

3.2. Языки и системы программирования

Java – дитя интернета





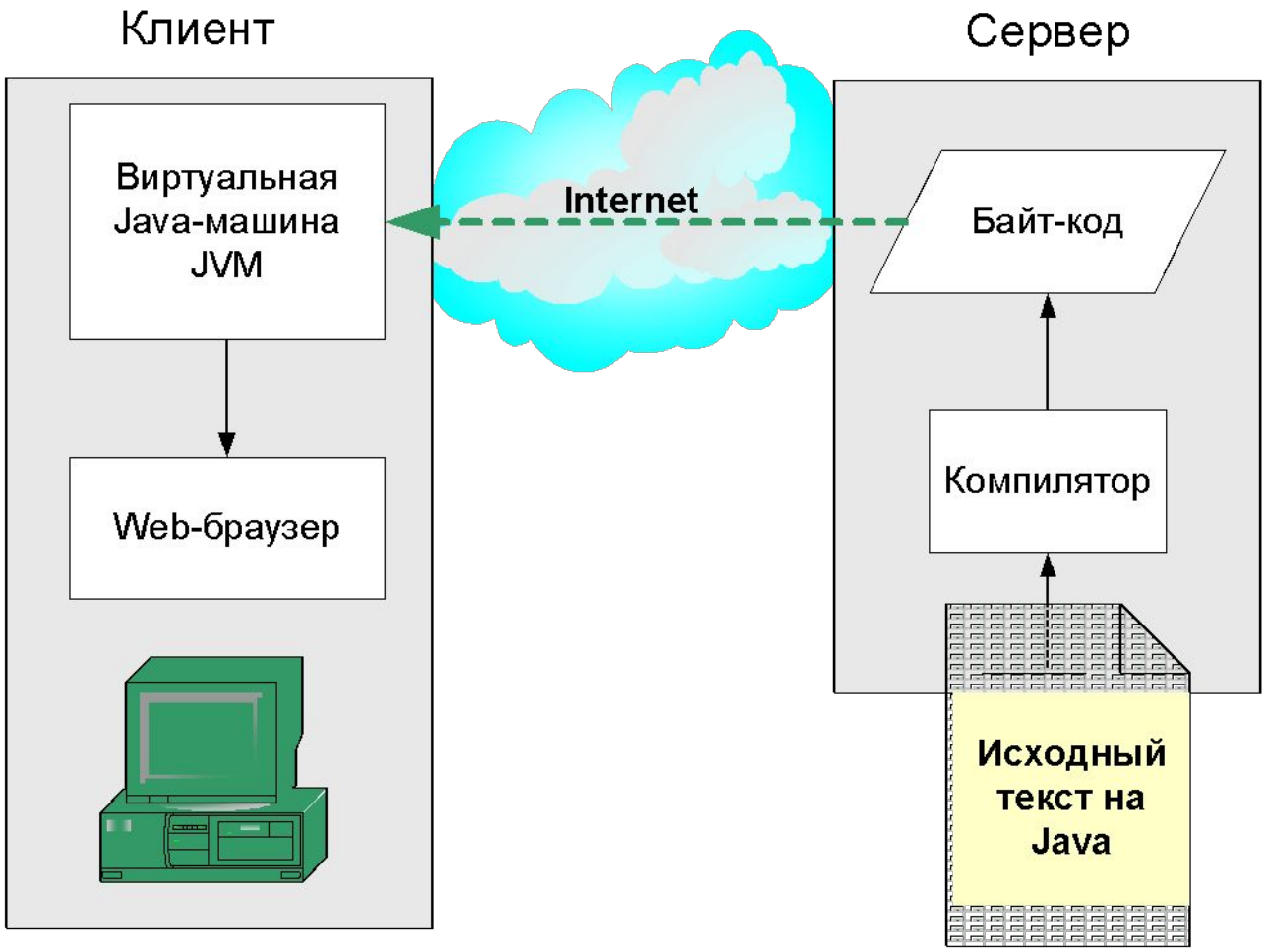
В 1995 г. фирма Sun Microsystems представила язык Java для программирования в интернете.

Он возник в ходе реализации проекта Oak («Дуб»), целью которого было создание системы программирования бытовых микропроцессорных устройств.

Джеймс Гослинг (Gosling, James) – автор Java.

3.2. Языки и системы программирования

Java – дитя интернета



Java - технология

3.2. Языки и системы программирования

Java – дитя интернета

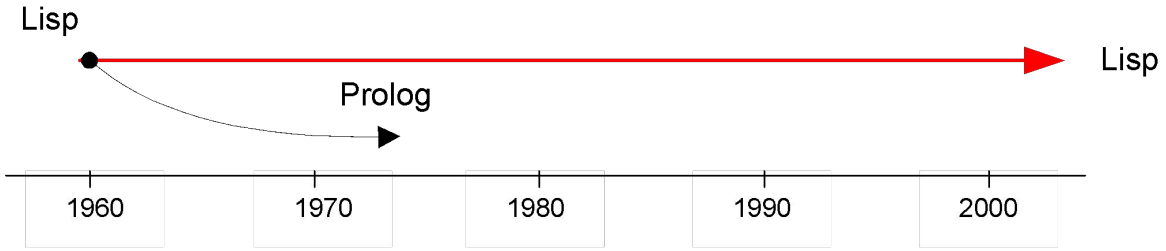
```
class test
{
  int i, n;
  float s;
  float x[n];
  public static void main( String args[]
)
  {
    n = 10;
    s = 0;
    for( i=1; i<=n; i++)
    {
      s = s + x[i-1];
      s = s / n;
    }
  }
```

Язык Java основан на C++

В качестве альтернативы Java корпорация Microsoft предложила язык C# (Си-шарп)

3.2. Языки и системы программирования

Долгожитель Lisp – инструмент функционального программирования



Дж. Маккарти и А.П. Ершов
Снимок 1975 г.

Lisp = LISt Processing

Язык Lisp создан в 1960 году Джоном Маккарти (McCarthy, John; p. 1927) в Массачусетском технологическом институте на теоретическом фундаменте лямбда-исчисления, предложенного еще в 1930 году известным американским логиком Алонзо Черчем.

3.2. Языки и системы программирования

Долгожитель Lisp – инструмент функционального программирования

```
(setq L `(8 5 13 11 10))
(defun sum (L)
  (cond ((null L) '0)
    (t (add (car L) (sum (cdr L))))
  )
)
(div (sum L) '5)
```

Примитивы:

cond — условная функция, проверяющая с помощью функции null пустоту списка;

add — суммирование аргументов;

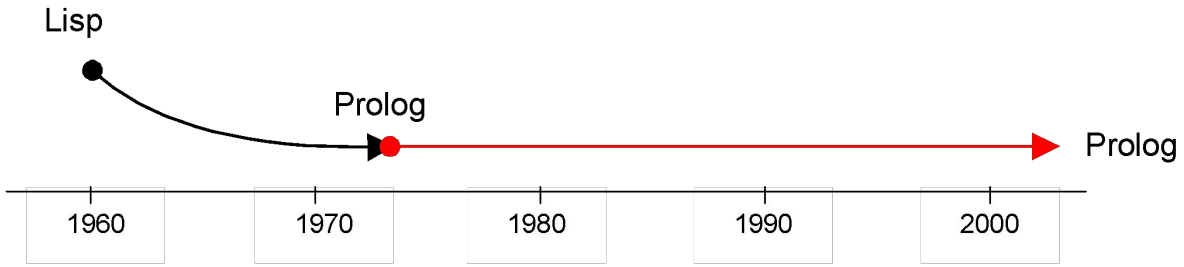
car — извлечение первого элемента из списка;

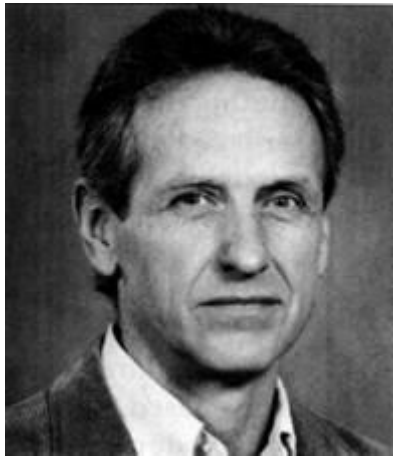
cdr — извлечение остатка списка (без первого элемента).

Программа на Lisp имеет специфический вид из-за обилия скобок. За это студенты прозвали его «Lots of Infuriating & Silly Parenthesis» - «Множество раздражающих и глупых скобок»

3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения





Prolog = PROgramming for LOGic

Теоретические основы языка были разработаны Робертом Ковальским (Kowalski, Robert) в Эдинбургском университете (Шотландия) в конце 1960-х годов

Первая практическая реализация языка осуществлена Аленом Кольмари (Colmerauer, Alain) в Марсельском университете (Франция) в 1972 г.

3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения

Факты:

муж (петя), муж (ваня),

муж (коля), жен (таня), жен (маша),

мать (ваня, таня), отец (ваня, петя),

отец (маша, ваня), отец (коля, ваня).

Правила вывода:

родитель (X, Y) :— отец (X, Y)

родитель (X, Y) :— мать (X, Y)

дед (X, Y) :— родитель (X, Z), отец (Z, Y)

брат (X, Y) :— муж (Y), родитель (X, Z),
родитель (Y, Z), X<>Y

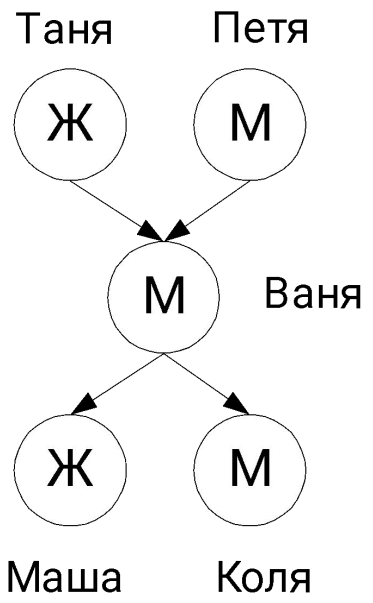
Примеры диалога:

GOAL> дед (коля, X) *Кто дед Коли?*

X = Петя

GOAL> брат (маша, X) *Кто брат Маши?*

X = Коля



Описание предметной области семейных отношений на языке Prolog

3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения

Проект ЭВМ V поколения – японский вызов мировой компьютерной индустрии, брошенный в начале 1980-х годов

3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения

Концептуальные отличия ЭВМ V поколения:

• новая технология производства микросхем, знаменующая переход от кремния к арсениду галлия, и дающая возможность на порядок повысить быстродействие основных логических элементов;

• новая архитектура (не фон-неймановская);

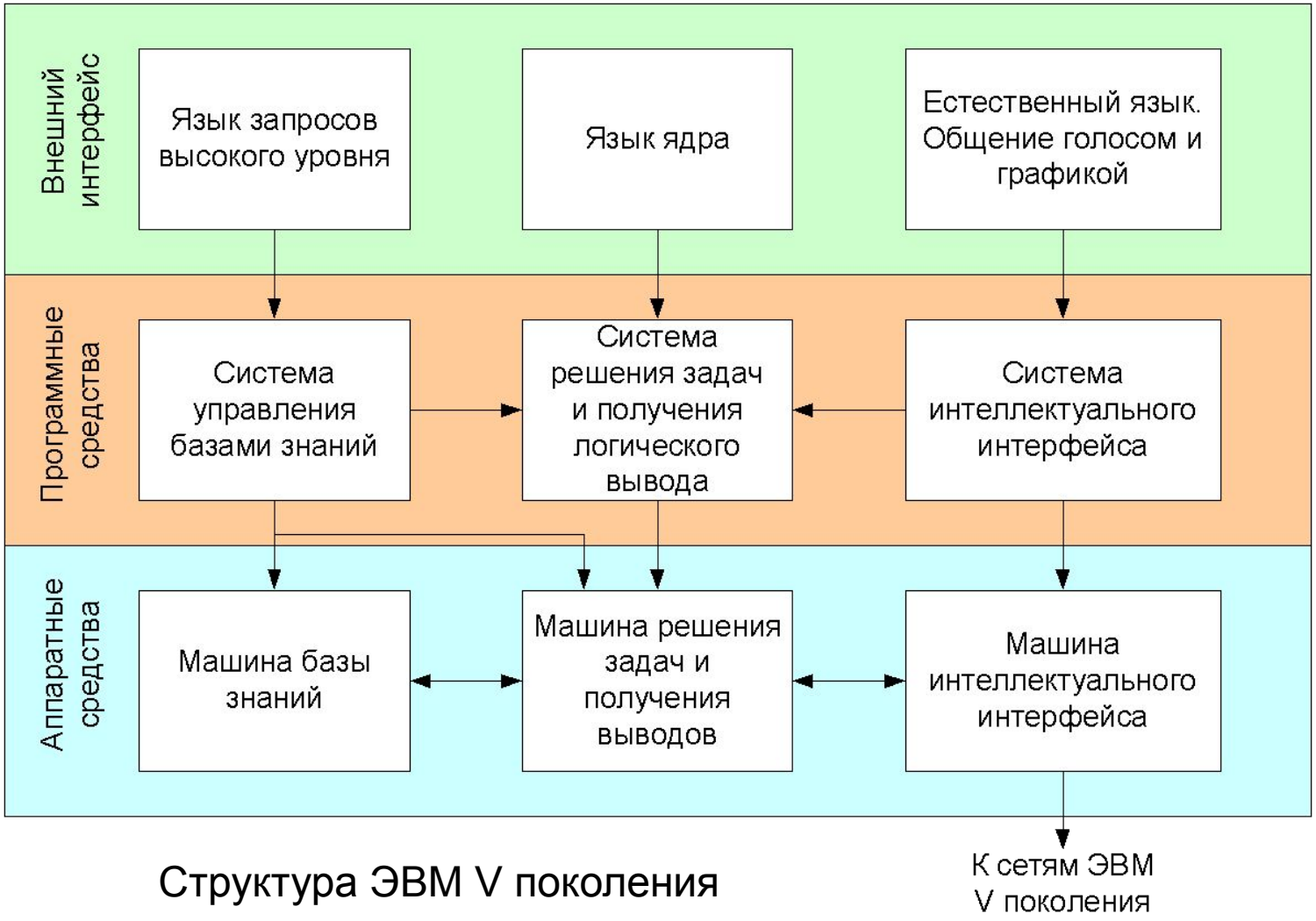
• новые способы ввода-вывода информации — распознавание и синтез речи и образов;

• отказ от традиционных алгоритмических языков программирования (Фортран, Алгол и т. п.) в пользу декларативных;

• ориентация на задачи искусственного интеллекта с автоматическим поиском решения на основе логического вывода.

3.2. Языки и системы программирования

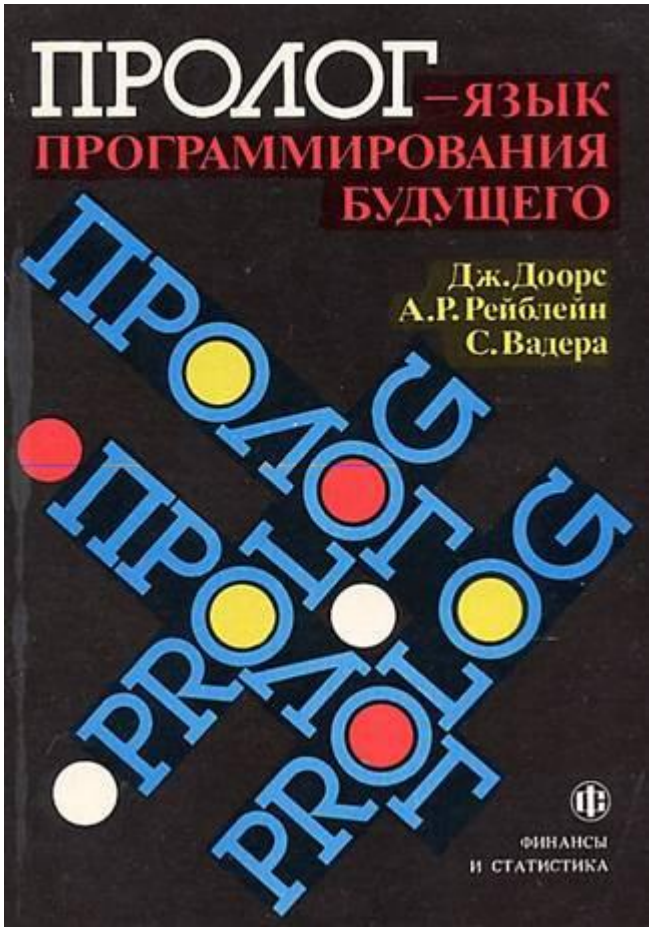
Prolog – несостоявшаяся мечта ЭВМ V поколения



Структура ЭВМ V поколения

3.2. Языки и системы программирования

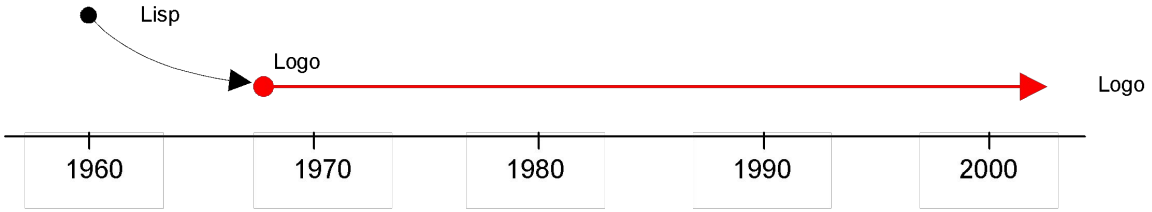
Prolog – несостоявшаяся мечта ЭВМ V поколения



В качестве основного языка ЭВМ V поколения предполагалось использовать Prolog

3.2. Языки и системы программирования

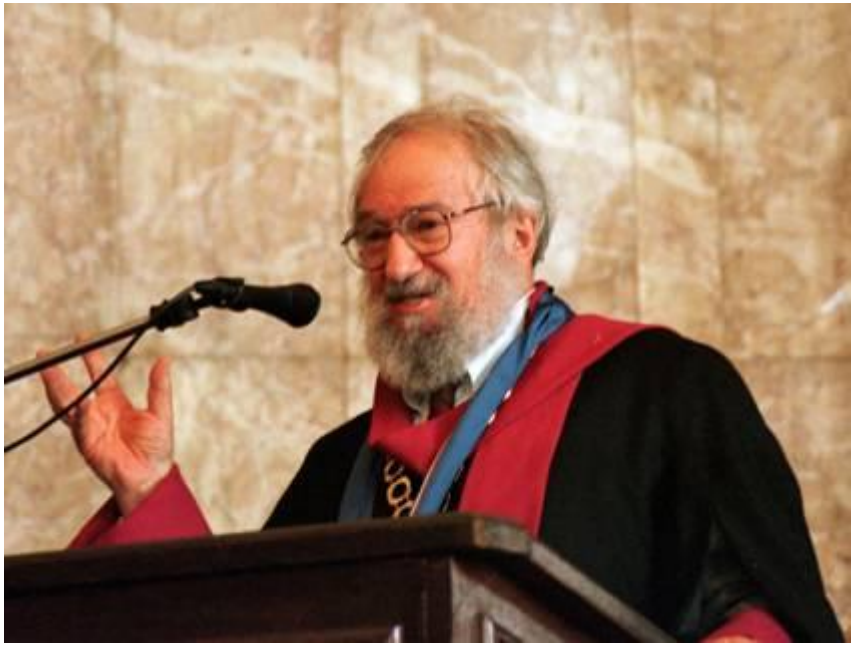
Logo – язык для самых маленьких



Язык Logo, изобретен в 1967 г. в MIT выдающимся математиком и педагогом Сеймуром Пейпертом (Papert, Seymour; p. 1928).

Пейперт в 1958-1963 годах работал в Женеве у знаменитого психолога Жана Пиаже (Piaget, Jean), где занимался детьми и природой их мышления.

Идейной основой Logo является язык Lisp



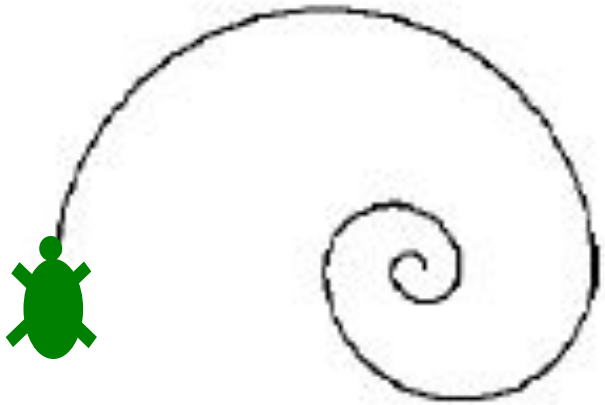
На фото: Сеймур Пейперт получает степень почетного доктора Софийского университета (1999 г.)

3.2. Языки и системы программирования

Logo – язык для самых маленьких

```
это дуга :шаг :число_шагов
  повтори :число_шагов
    [вперед :шаг направо 10]
Конец

это спираль :шаг
  если :шаг < 1 [стоп]
  дуга :шаг 18
  спираль :шаг / 2
конец
```

Цикл



Процедура с параметром

Рекурсия

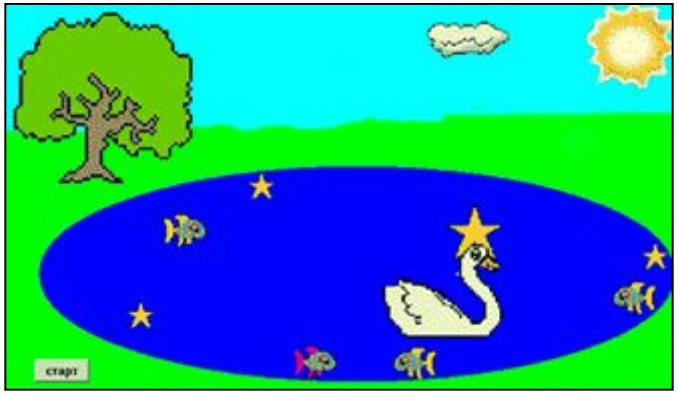
Программа на Logo управляет черепашкой, оставляющей видимый след. С помощью зрительных образов интерпретируются все базовые структуры программирования

3.2. Языки и системы программирования

Logo – язык для самых маленьких

Концепция Logo учитывает детскую психологию и рассчитана на обучение школьников, начиная с младших классов

На снимке: группа юных программистов на факультете информатики Томского государственного университета. Занятие ведет доцент Т.Н. Поддубная (2002 г.)



Новейшие реализации Logo используют принципы объектно-ориентированного программирования.

В программе Юли Гладких, 9 лет, черепашка в форме лебедя плавает по озеру.

3.2. Языки и системы программирования

Уроки истории

Основные парадигмы программирования:

- процедурное программирование (Fortran, Basic, Cobol, Algol, Pascal, Ada, C, Logo, FoxPro);
- объектно-ориентированное программирование (Simula, Smalltalk, Object Pascal, C++, Java, C#);
- визуально-событийное программирование (Visual Basic, Delphi, Visual C++, Visual Java, Visual FoxPro);
- функциональное программирование (Lisp);
- логическое программирование (Prolog).

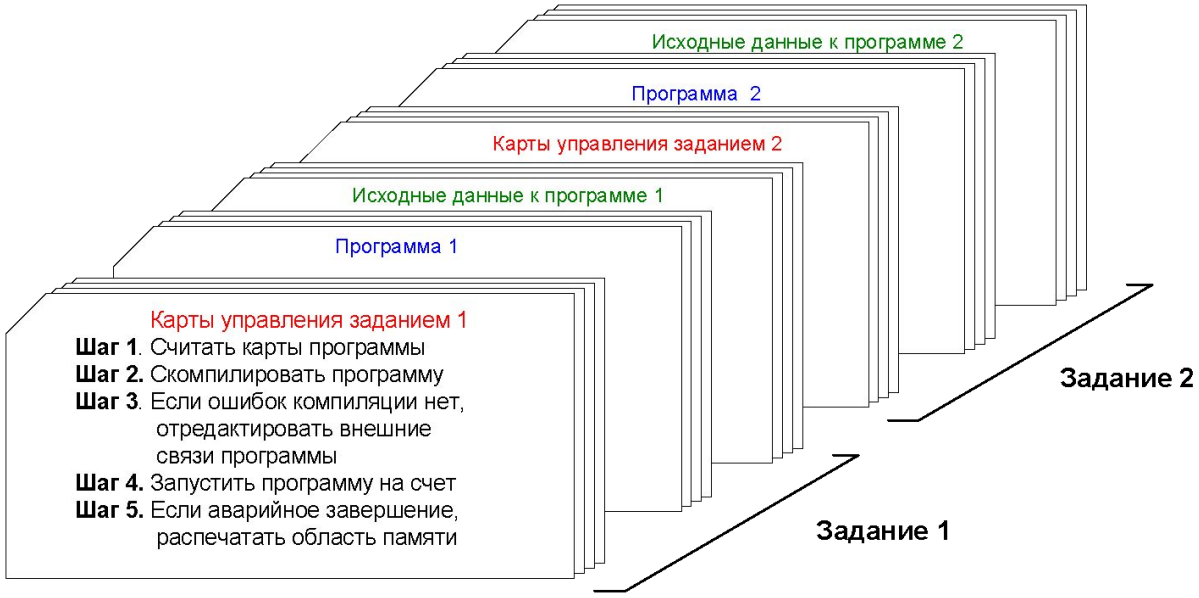
3.3. Операционные системы

50-е годы: человек-оператор

В ЭВМ первого поколения (1950-е годы) управление вычислениями велось с пульта управления. Каждая программа вручную вводилась с перфокарт и запускалась оператором. Это приводило к простоям дорогостоящего оборудования и ошибкам

3.3. Операционные системы

60-е годы: от автооператора до пакетных ОС с мультипрограммированием



Пакетные операционные системы, созданные в 1960-е годы, предназначены для автоматизации работы оператора. Перфокарты, соответствующие отдельным заданиям, собираются в пакет заданий. Каждое задание снабжается управляющими картами, задающими правила поведения автооператора на языке управления заданиями (Job Control Language - JCL)

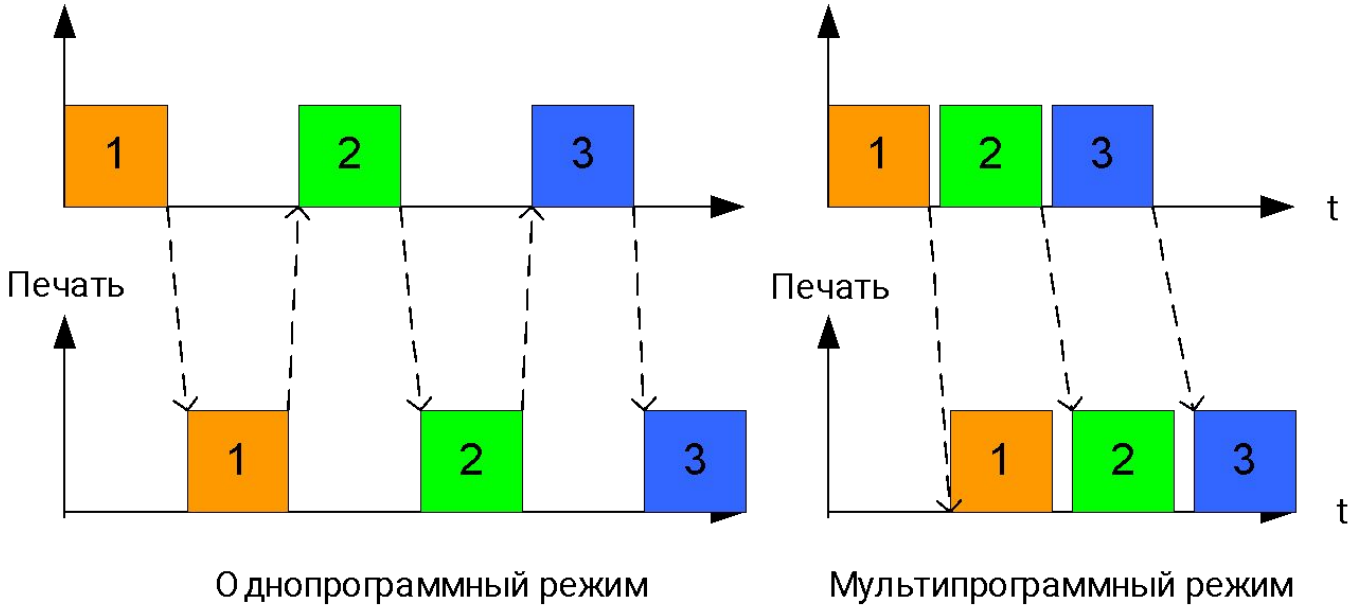
3.3. Операционные системы

60-е годы: от автооператора до пакетных ОС с мультипрограммированием

Пакет перфокарт загружается в читающее устройство. Операционная система по очереди считывает задания и сама выполняет необходимые действия по запуску программ

3.3. Операционные системы

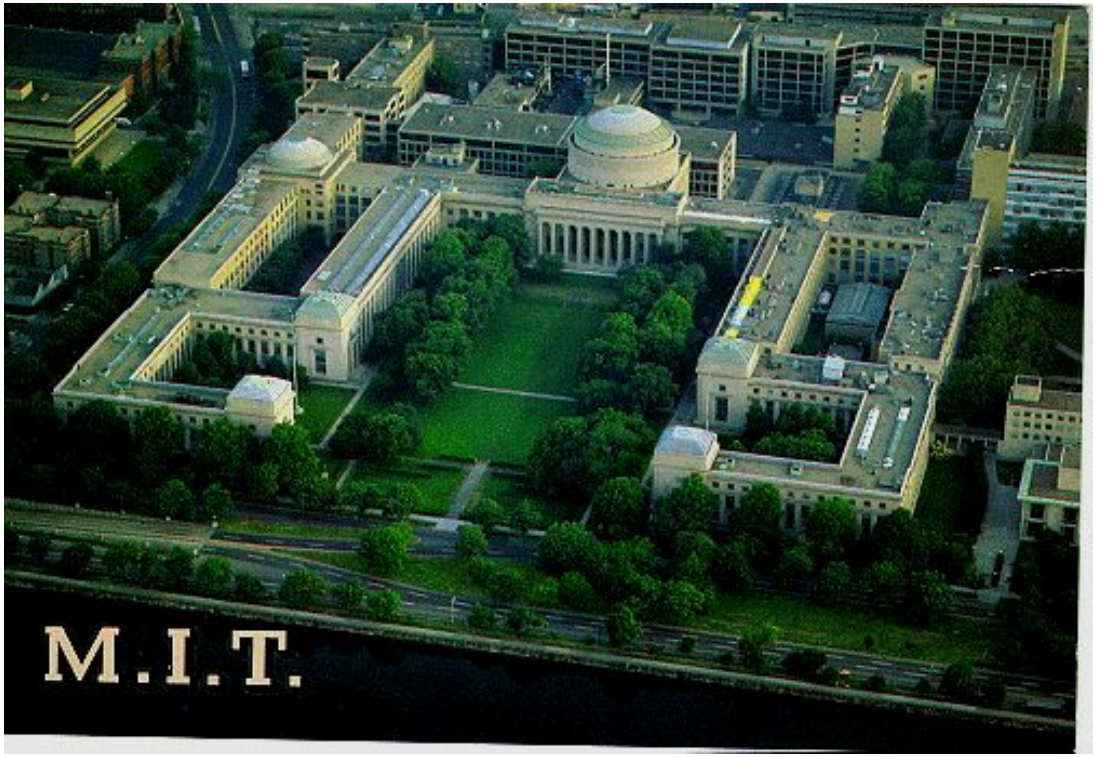
60-е годы: от автооператора до пакетных ОС с мультипрограммированием



Применение *мультипрограммного* режима работы пакетной операционной системы позволяет сократить простои отдельных устройств и увеличить производительность компьютера по сравнению с однопрограммным

3.3. Операционные системы

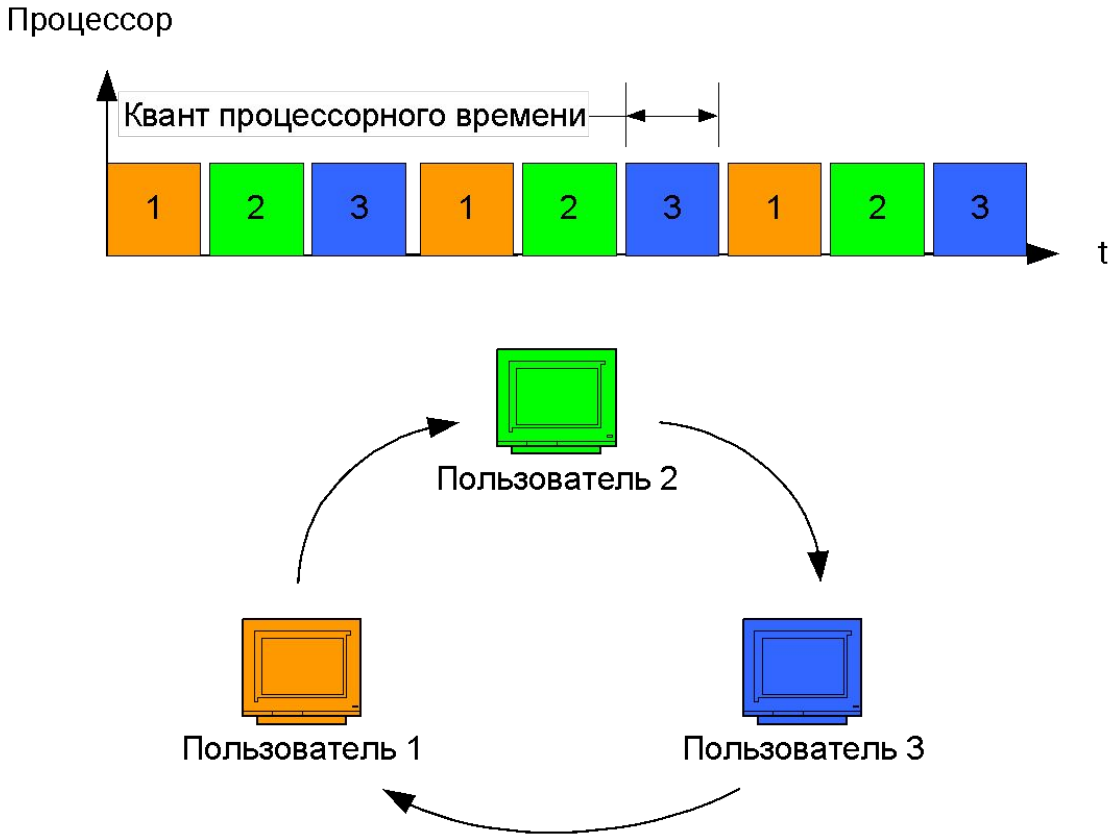
70-е годы: диалоговые ОС с разделением времени



Родина диалоговых операционных систем с разделением времени –Массачусетский технологический институт – МТИ (Massachusetts Institute of Technology – MIT), прославившийся многими достижениями в области вычислительной техники и информатики

3.3. Операционные системы

70-е годы: диалоговые ОС с разделением времени

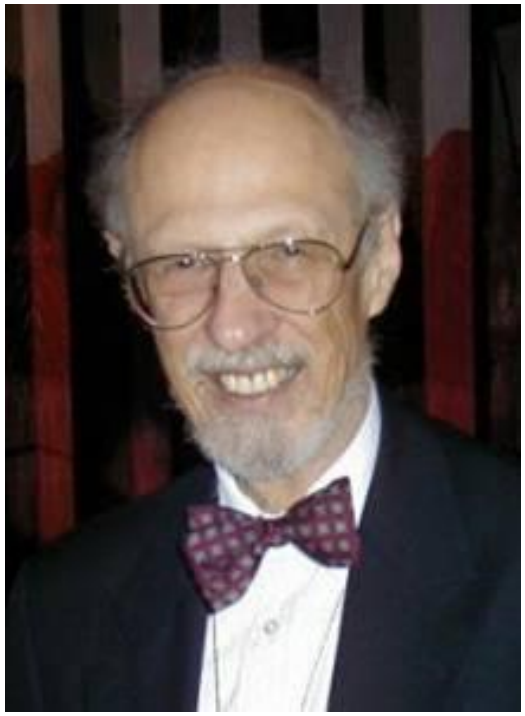


Принцип разделения времени был предложен Джоном Маккарти в 1959 году

3.3. Операционные системы

70-е годы: диалоговые ОС с разделением времени

Первая экспериментальная операционная система с разделением времени CTSS (Compartible Time Sharing System).была создана в МТИ в 1961 г. под руководством адъюнкт-профессора Фернандо Корбато (Corbato, Fernando; p. 1926))



Система функционировала на модифицированной ЭВМ IBM-7090

3.3. Операционные системы

70-е годы: диалоговые ОС с разделением времени

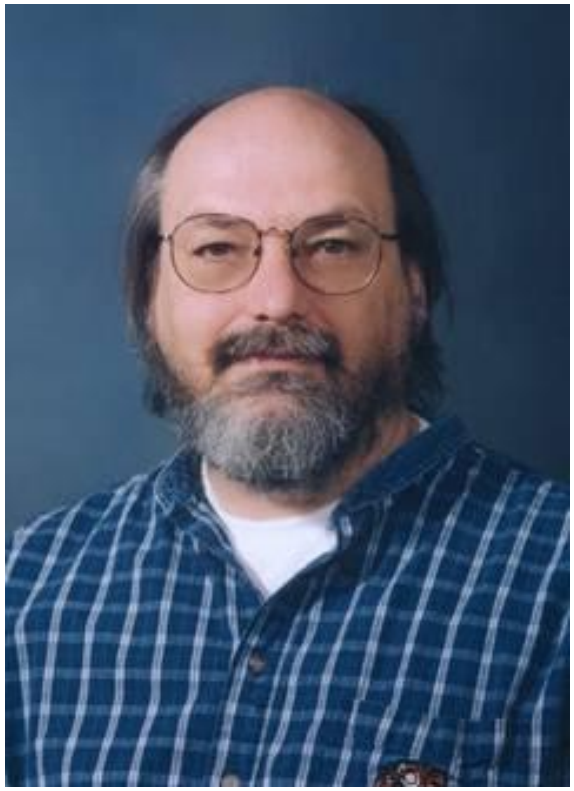
На основе опыта CTSS в МТИ в 1962-1970 годах велась разработка большой учебной системы коллективного доступа MAC (Multiplexed Access Computer), в рамках которой была создана ОС РВ Multics

Для проекта MAC фирма General Electric поставила ЭВМ GE-645

Система Multics стала прародителем всех будущих ОС РВ, она была полигоном, на котором выросло поколение системных программистов

3.3. Операционные системы

70-е годы: диалоговые ОС с разделением времени



Сотрудник Bell Labs Кеннет Томпсон (Thompson, Kenneth; p. 1943), участвовавший в проекте Multics, по собственной инициативе разработал ОС Unix для PDP-7 (1969 г.)

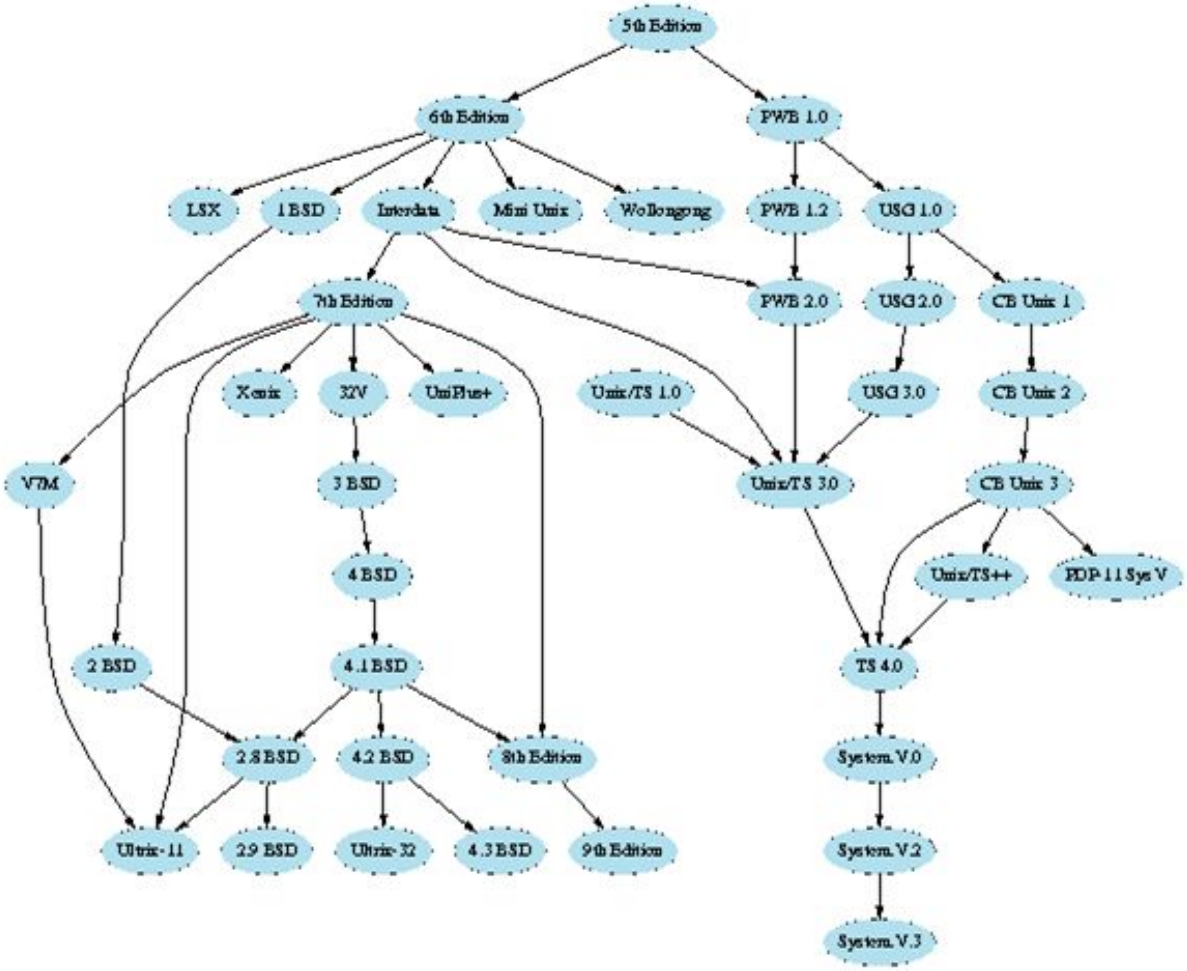
3.3. Операционные системы

70-е годы: диалоговые ОС с разделением времени

Вместе с Деннисом Ричи (на фото слева), создавшим язык С, они перенесли Unix на ЭВМ PDP-11 с двумя телетайпами в качестве терминалов (1971-1973 годы)

3.3. Операционные системы

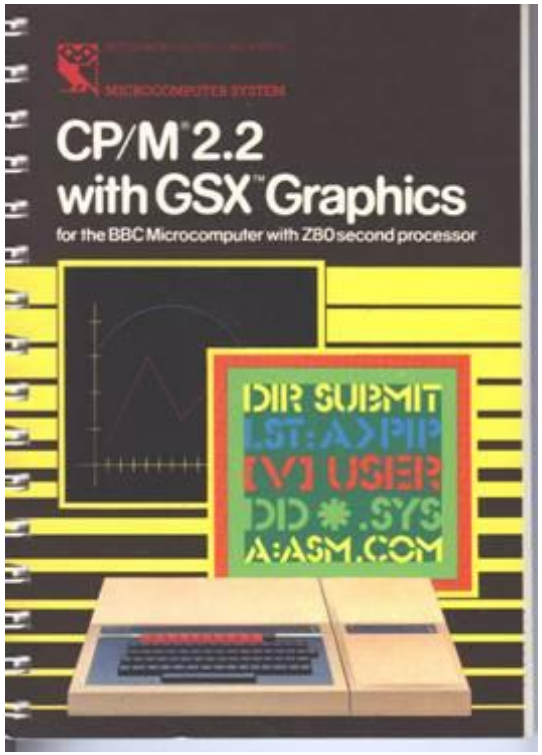
70-е годы: диалоговые ОС с разделением времени



В 1970-80-е годы появилось множество некоммерческих и коммерческих версий системы Unix

3.3. Операционные системы

80-90-е годы: настольные ОС



Стандартной операционной системой для первого поколения 8-битовых персональных компьютеров была CP/M, написанная в 1976 году Гэри Килдолом (Kildall, Gary; 1942-1994), основателем компании Digital Research.

3.3. Операционные системы

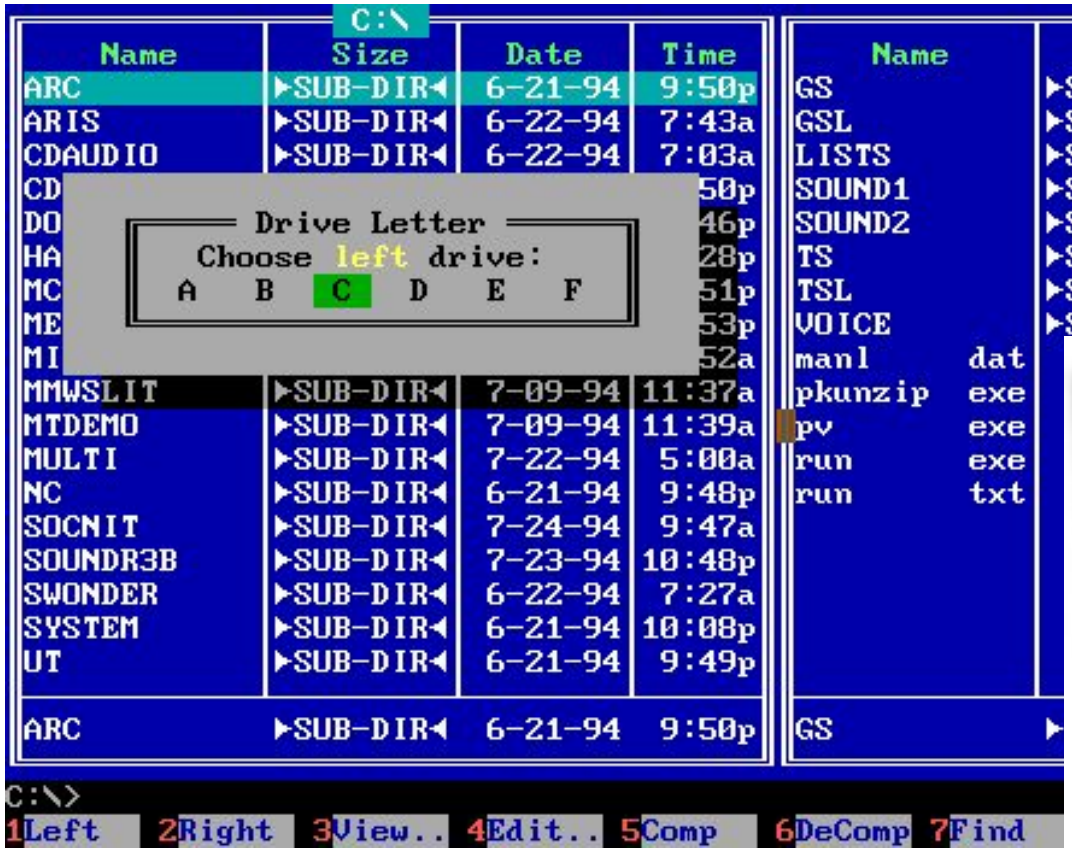
80-90-е годы: настольные ОС

```
C:\WINDOWS>dir /?
Displays a list of files and subdirectories in a directory.

DIR [drive:][path][filename] [/P] [/W] [/A[[:]attributes]]
  [/O[[:]sortorder]] [/S] [/B] [/L] [/V] [/4]

  [drive:][path][filename]
              Specifies drive, directory, and/or files to list.
              (Could be enhanced file specification or multiple
  /P          Pauses after each screenful of information.
  /W          Uses wide list format.
  /A          Displays files with specified attributes.
  attributes   D  Directories                R  Read-only files
               H  Hidden files               A  Files ready for
               S  System files               -  Prefix meaning
  /O          List by files in sorted order.
  sortorder    N  By name (alphabetic)       S  By size (smalle
               E  By extension (alphabetic)  D  By date & time (earliest first)
               G  Group directories first    -  Prefix to reverse order
               A  By Last Access Date (earliest first)
  /S          Displays files in specified directory and all subdirectories.
  /B          Uses bare format (no heading information or summary).
  /L          Uses lowercase.
  /V          Verbose mode.
  /4          Displays year with 4 digits (ignored if /V also given).

Switches may be preset in the DIRCMD environment variable.  Override
preset switches by prefixing any switch with - (hyphen)--for example, /-W.

C:\WINDOWS>
```

Операционная систем MS DOS, выпущенная в 1981 г. фирмой Microsoft для IBM PC, базировалась на ОС Q-DOS (Quick and Dirty — быстрая и грязная), купленной у Тима Паттерсона (Tim Patterson) из фирмы Seattle Computer Products

3.3. Операционные системы

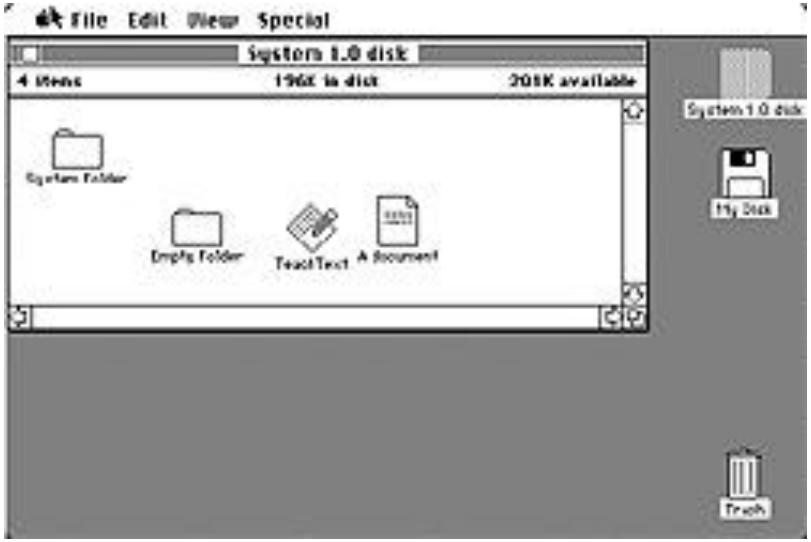
80-90-е годы: настольные ОС



Для облегчения работ с DOS широко использовались командно-файловые оболочки, самой популярной из которых была Norton Commander, созданная основателем фирмы Semantec Питером Нортоном (Norton, Peter),

3.3. Операционные системы

80-90-е годы: настольные ОС



Операционная система Mac OS 1.0, выпущенная в 1984 г. фирмой Apple для ПК Macintosh, имела графический пользовательский интерфейс (GUI), основанный на многолетних исследованиях Xerox PARC .

Система управлялась мышью, она сформировала стандарт на все последующие поколения графических ОС. Внешний вид самой Mac OS за 15 лет почти не изменился.

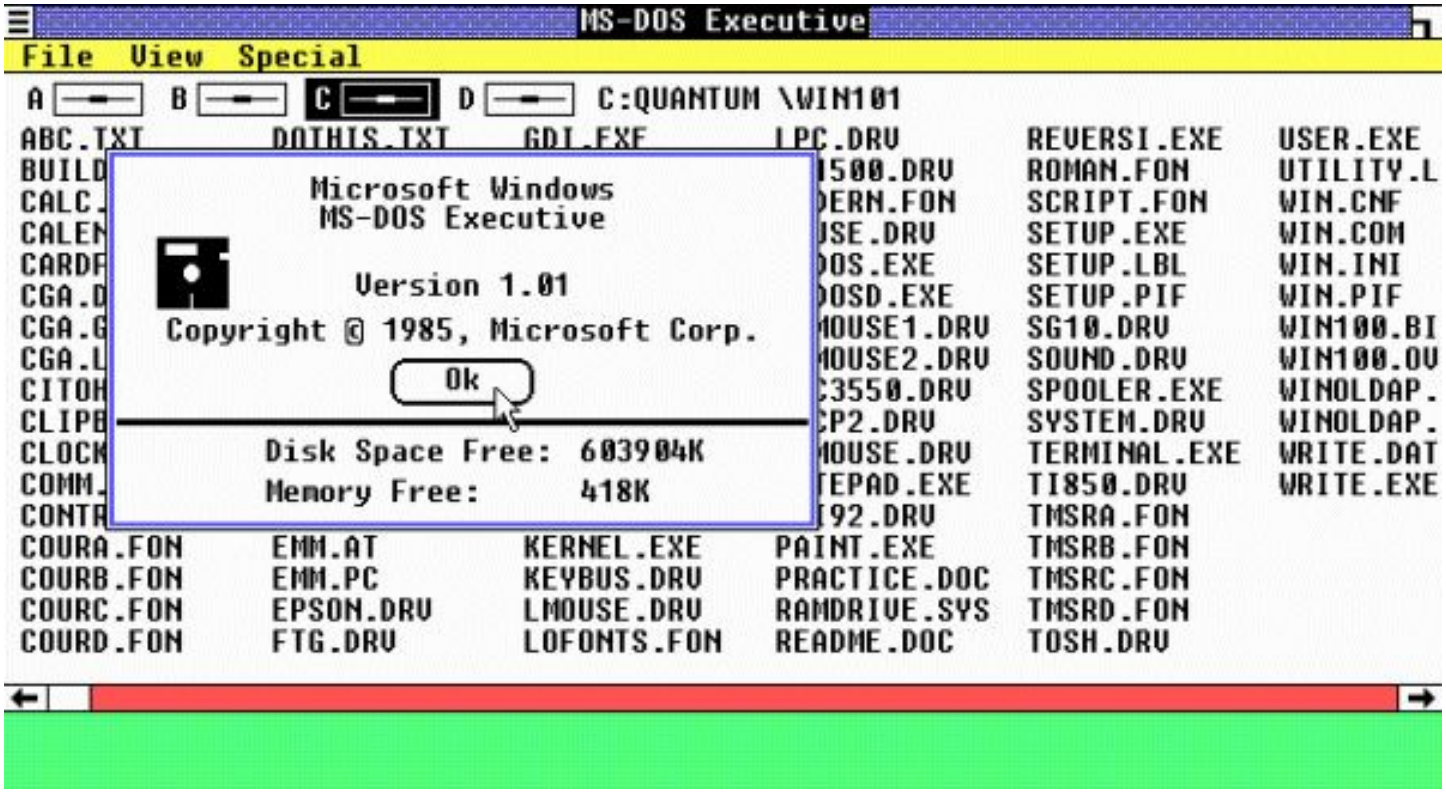
3.3. Операционные системы

80-90-е годы: настольные ОС

Логическим развитием Mac OS стала объектно-ориентированная ОС NeXTSTEP для компьютера NeXT (1987 г.)

3.3. Операционные системы

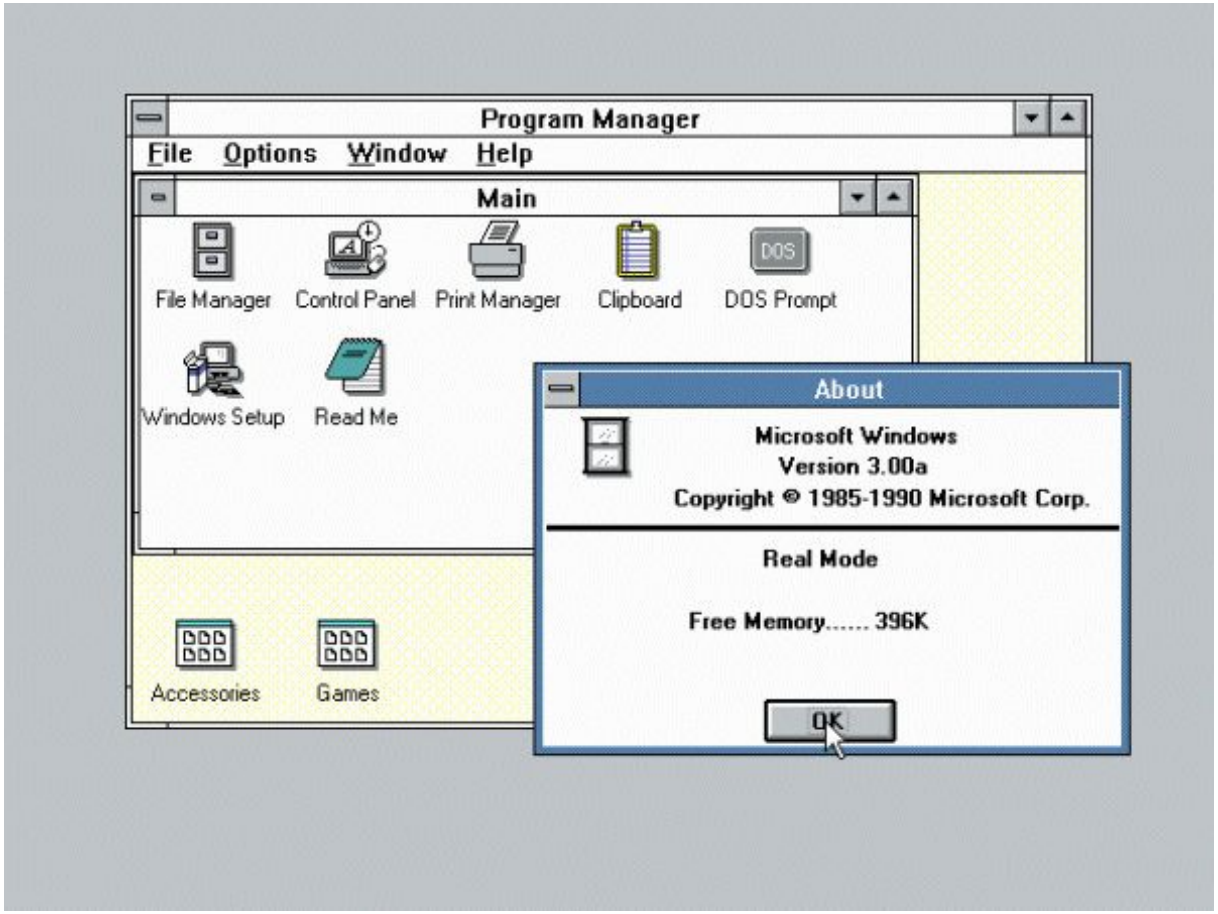
80-90-е годы: настольные ОС



Windows 1.0, выпущенная Microsoft в 1985 г., еще не была полноценной операционной системой. Она работала под DOS как обычная резидентная программа и была очень медленной

3.3. Операционные системы

80-90-е годы: настольные ОС



Начиная с версии 3.0 (1990 г.) Windows стала завоевывать рынок ОС для настольных ПК

3.3. Операционные системы

80-90-е годы: настольные ОС

Первая половина 1990-х годов отмечена противостоянием Microsoft Windows и операционной системой OS/2 фирмы IBM

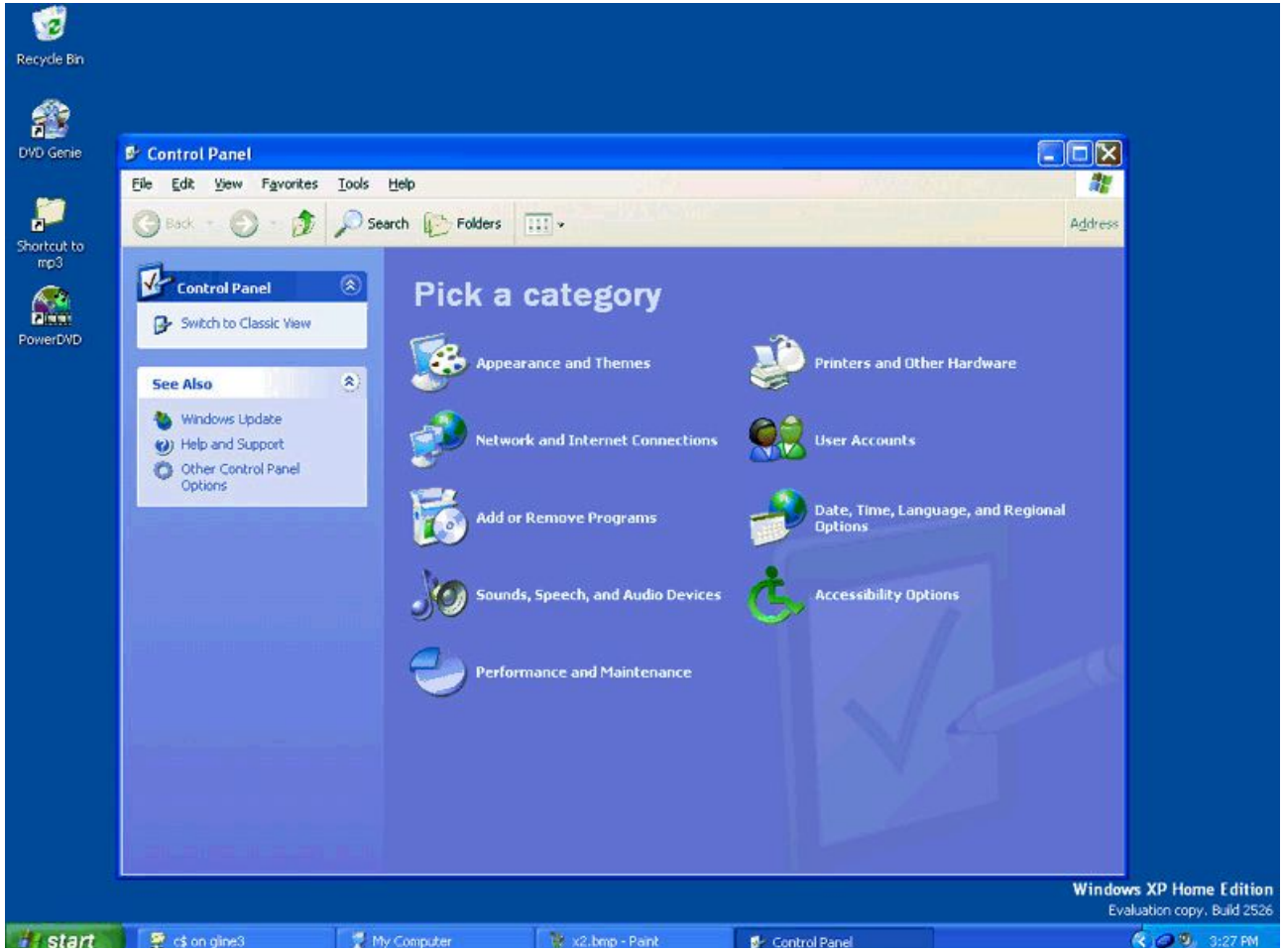
3.3. Операционные системы

80-90-е годы: настольные ОС

32-разрядная Windows-95 имела удобный графический дизайн. Она закрепила победу Microsoft на рынке настольных операционных систем и «убила» 486-й процессор

3.3. Операционные системы

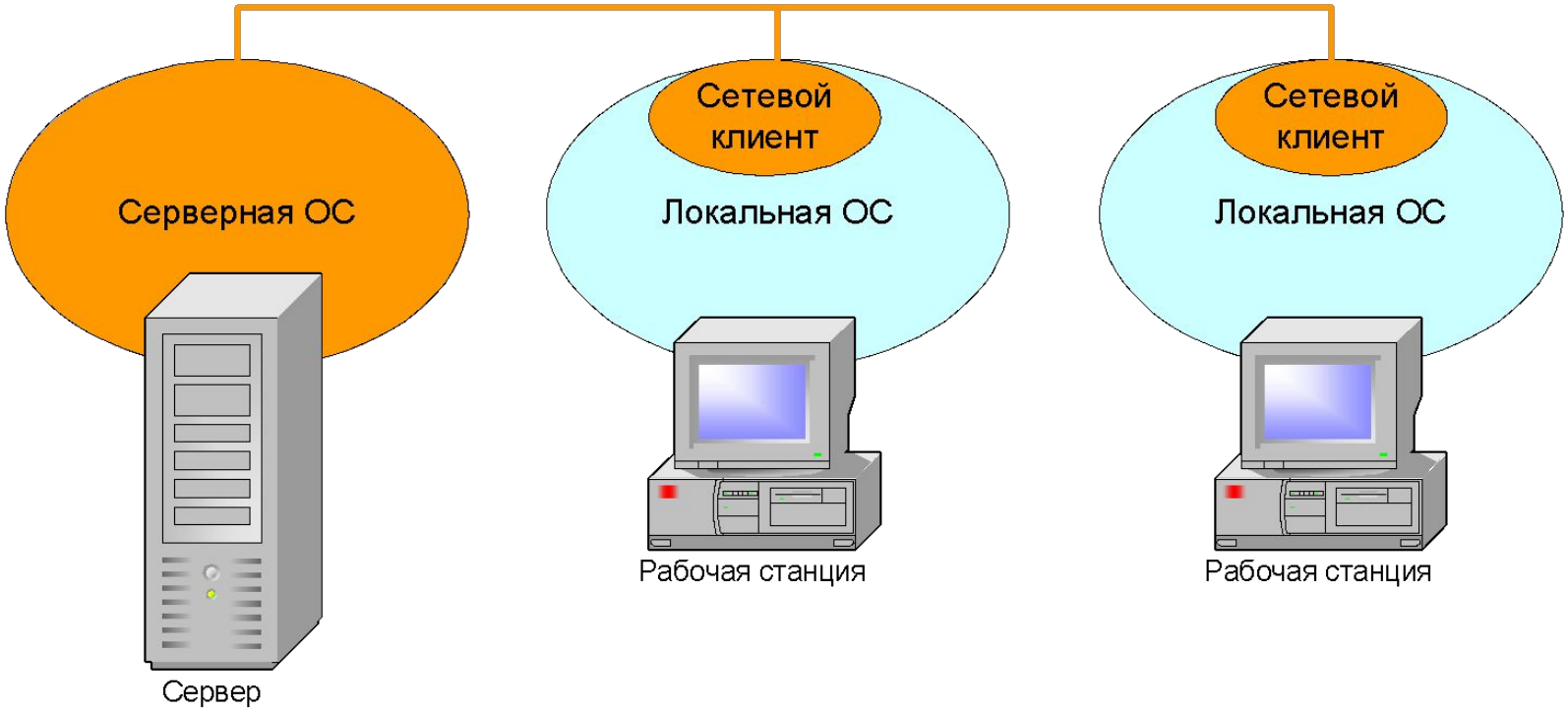
80-90-е годы: настольные ОС



Windows XP (2001 г.) – последняя из семейства операционных систем компании Microsoft

3.3. Операционные системы

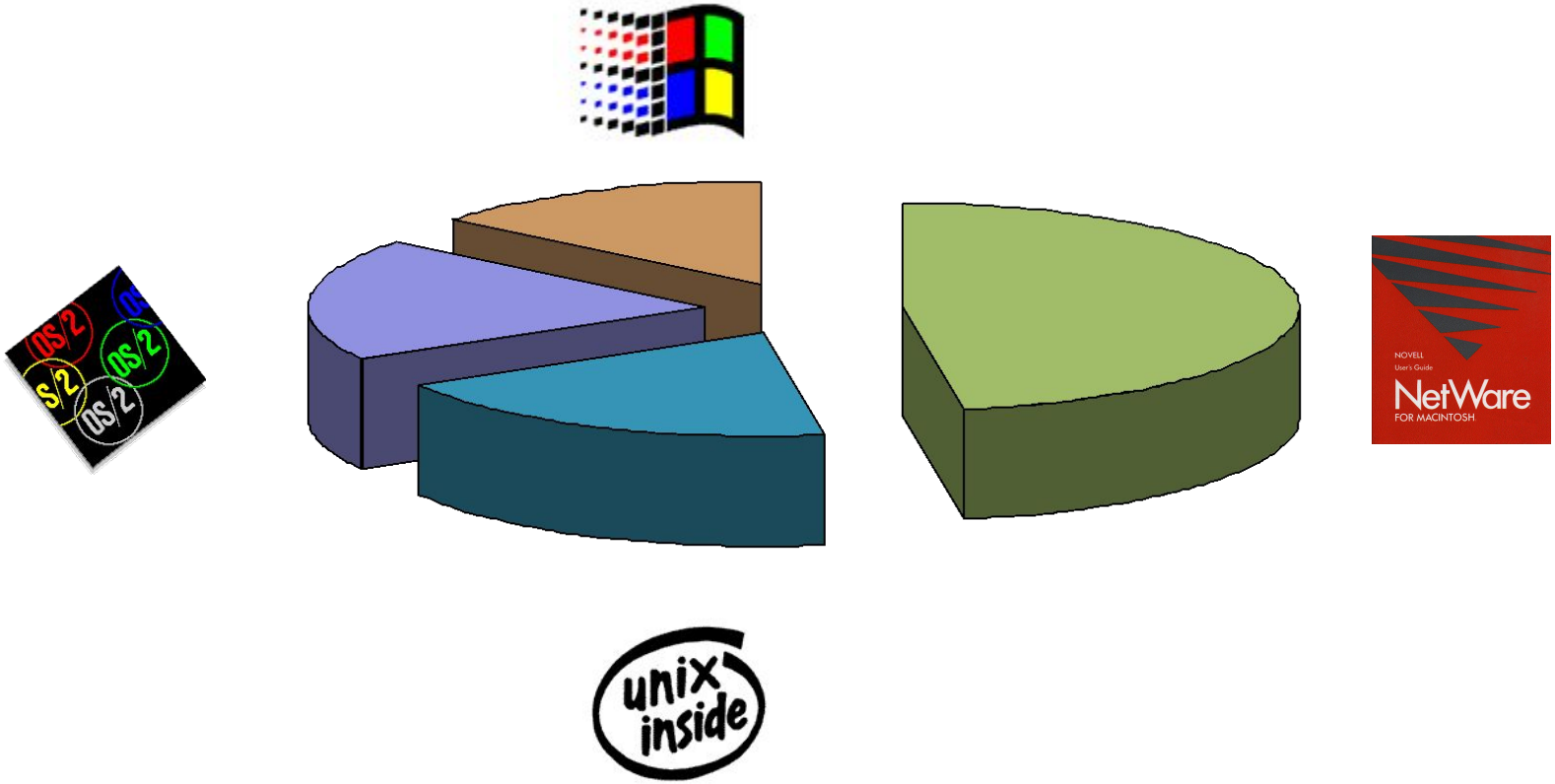
Сетевые ОС



На выделенных серверах устанавливаются операционные системы, специально оптимизированные для выполнения тех или иных сетевых функций, а у клиентов — специальные модули, встраиваемые в операционную систему рабочей станции и обеспечивающие связь с серверами.

3.3. Операционные системы

Сетевые ОС



В середине 1990-х годов рынок сетевых ОС был поделен между четырьмя основными системами: NetWare, Unix, OS/2 и Windows NT

3.3. Операционные системы

Сетевые ОС

В 1991 году произошло событие, круто повернувшее ход истории Unix. Финский студент из Университета Хельсинки Линус Торвальдс (Torvalds, Linus; p. 1971) бросил вызов всем гигантам-производителям коммерческих версий Unix, написав ядро Linux

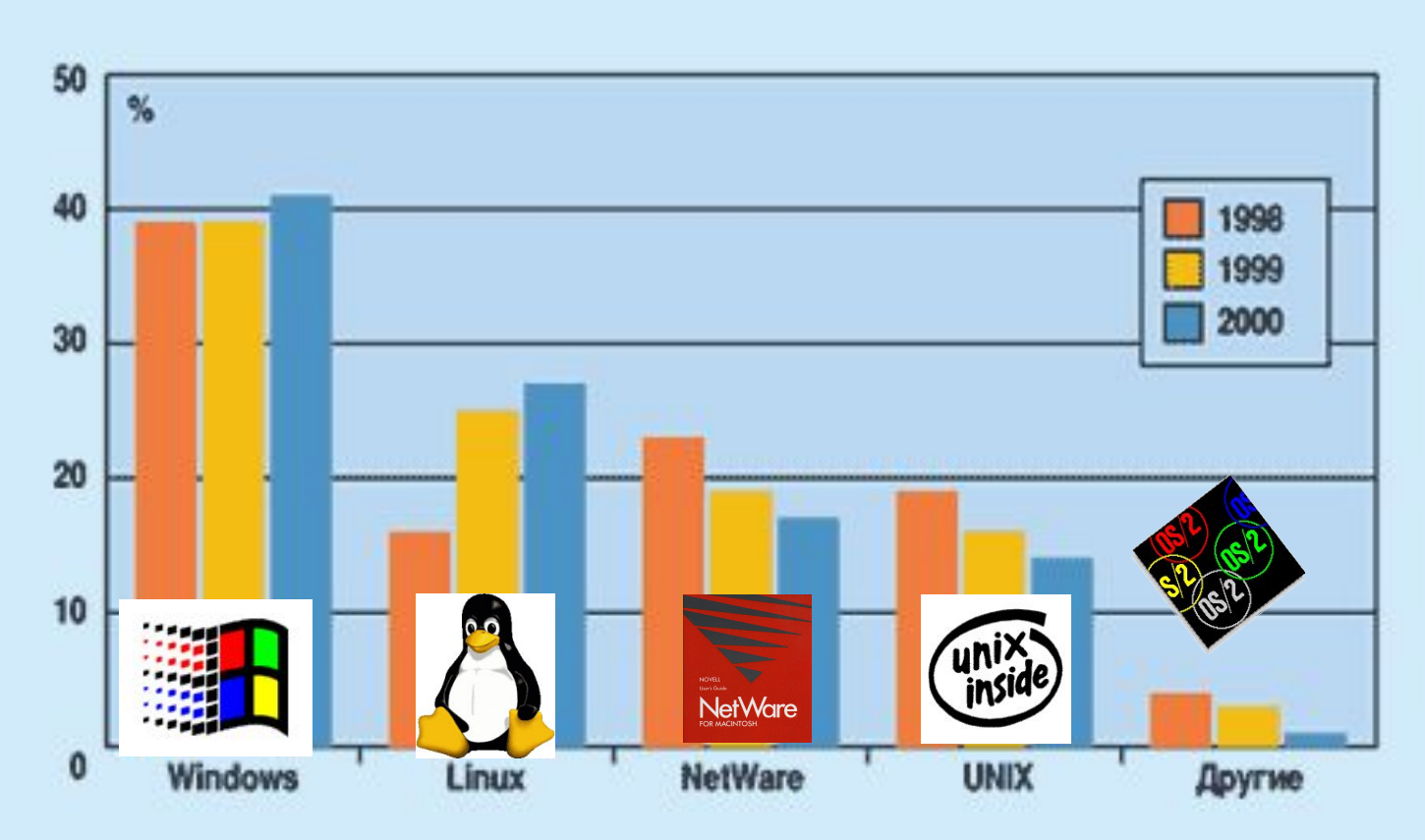
3.3. Операционные системы

Сетевые ОС

ОС Linux приобретает все большую популярность. Ей посвящаются специальные журналы, выпускаются коммерческие дистрибутивы (RedHat, BlackCat, Caldera и др.), ведущие производители ПО создают версии продуктов, работающие под Linux

3.3. Операционные системы

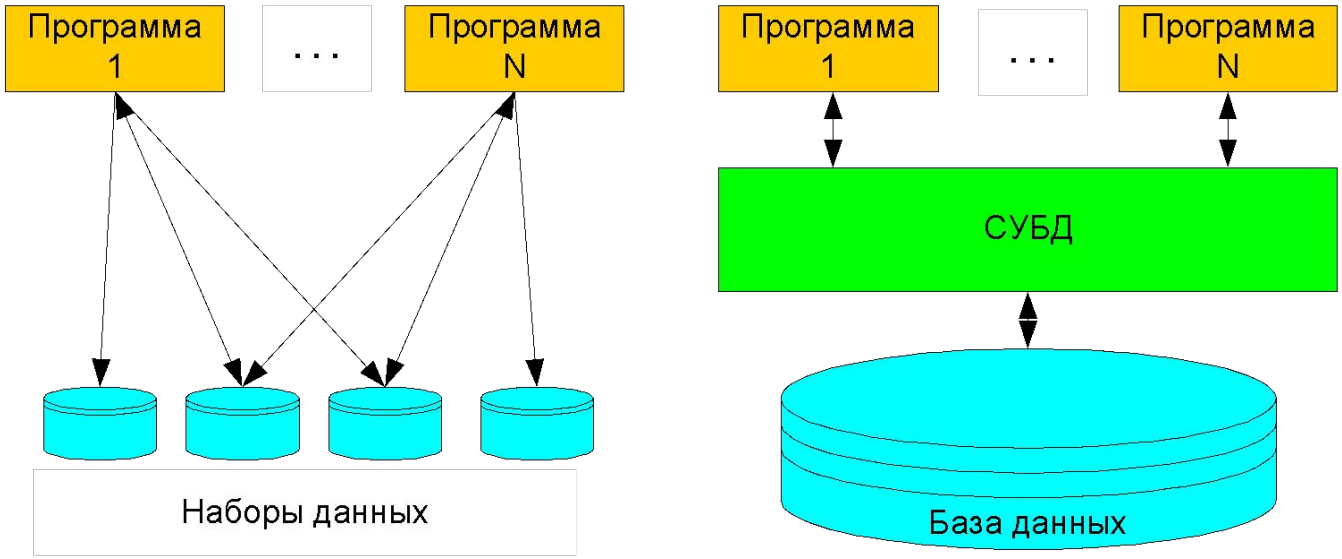
Сетевые ОС



В конце 1990-х годов на фоне преобладания Windows NT и снижения доли NetWare и OS/2 происходит резкий подъем популярности Linux

3.3. Системы управления базами данных

Предпосылки появления БД и СУБД



Взаимодействие программ с данными при независимой работе и при использовании СУБД

3.3. Системы управления базами данных

Предпосылки появления БД и СУБД

Преимущества использования СУБД:

- однократный ввод данных, что приводит к резкому сокращению затрат труда;
- независимость программ от данных, что дает возможность независимо менять те и другие;
- сокращение затрат на программирование

3.3. Системы управления базами данных

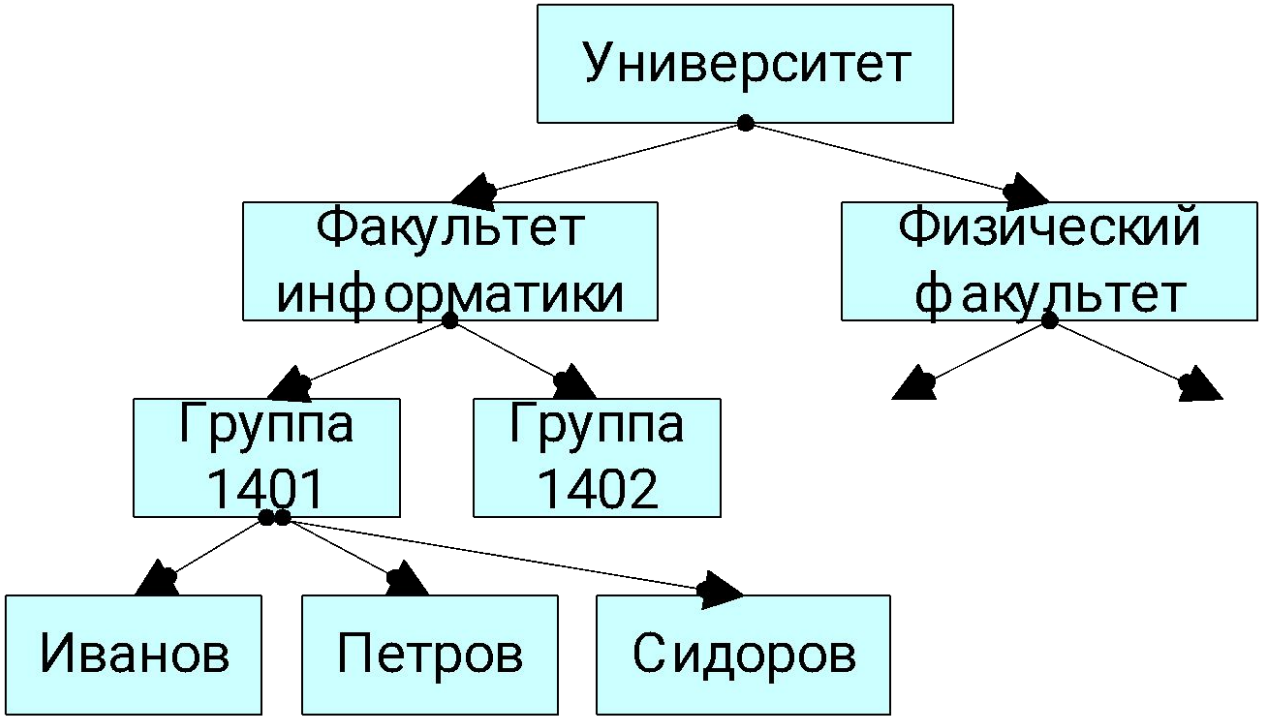
Предпосылки появления БД и СУБД

Основные функции СУБД:

- описание логической структуры базы данных
  - *с точки зрения администратора базы данных,*
  - *с точки зрения конкретной задачи;*
- манипулирование данными;
- обеспечение целостности БД;
- обеспечение многопользовательского доступа;
- защита данных от несанкционированного доступа.

3.3. Системы управления базами данных

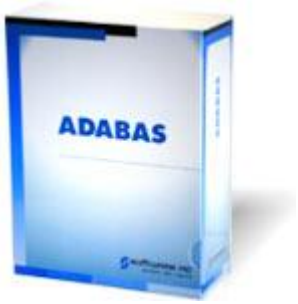
Основные типы и история развития СУБД

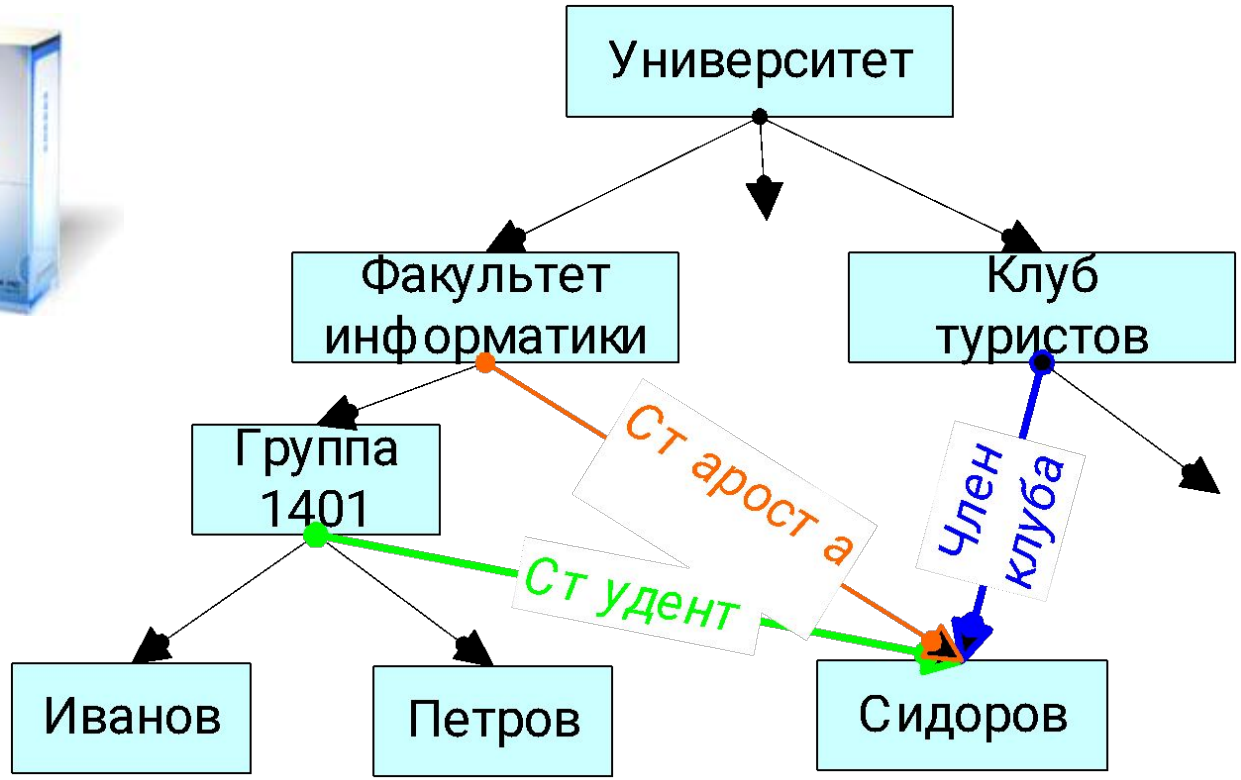


Иерархическая модель данных является исторически первой и интуитивно понятной.

3.3. Системы управления базами данных

Основные типы и история развития СУБД





Сетевая модель данных, разработанная рабочей группой CODASYL (1971 г.), описывает одновременно несколько иерархий. Сетевая модель реализована в ряде промышленных СУБД

3.3. Системы управления базами данных

Основные типы и история развития СУБД

ГРУППЫ

Группа	Факультет	Староста
1401	Информатики	Сидоров
1402	Информатики	Михайлов
501	Физический	Самойлов
...		

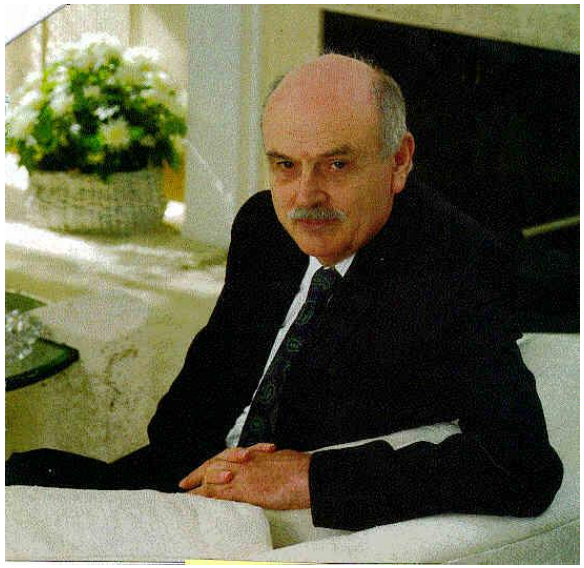
СТУДЕНТЫ

Фамилия	Группа
Иванов	1401
Петров	1401
Сидоров	1401
Трофимов	1402
...	

КЛУБЫ

Клуб	Член клуба
Туристов	Сидоров
Туристов	Пяткин
Водников	Ломакин
...	

Реляционная модель данных была предложена в 1970 году сотрудником IBM Эдгаром Коддом (Codd, Edgar F.; р. 1923). Использует формальный аппарат теории множеств и математической логики. Долгое время реляционная модель существовала только в теории

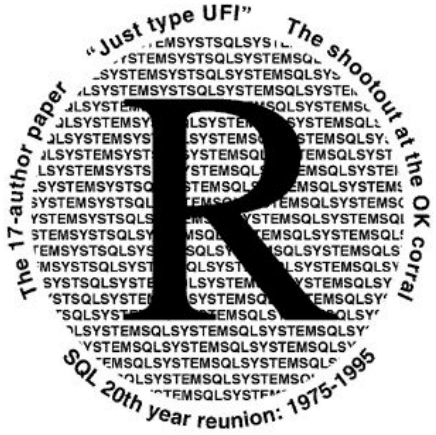


3.3. Системы управления базами данных

Основные типы и история развития СУБД

Для работы с данными, представленными в реляционной форме, Дональд Чемберлен (Chamberlin, Donald) и Рэй Бойс (Boyce, Ray) в 1974 году предложили язык запросов SEQUEL, который впоследствии был доработан и переименован в SQL — Structured Query Language.

Дэн Чемберлен



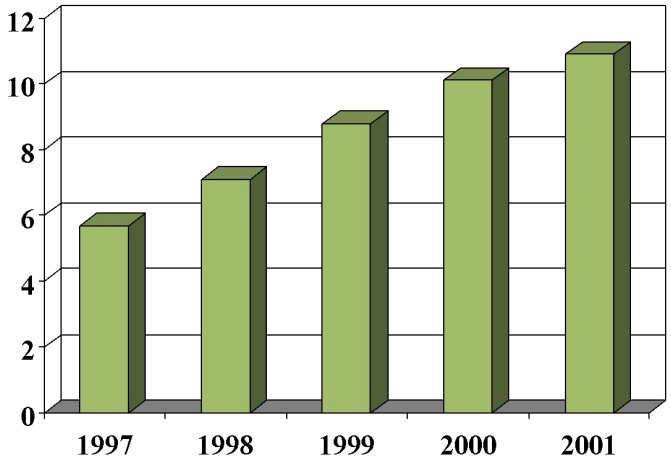
В ноябре 1976 года в журнале «IBM Journal of Research and Development» была опубликована статья Чемберлена с соавторами о проекте System/R, предлагающая путь эффективной реализации РСУБД и SQL.

Экспериментальная реляционная система System/R фирмы IBM закончена в 1979 г., коммерческая версия выпущена рынок в 1983 г. под названием DB2.

3.3. Системы управления базами данных

Основные типы и история развития СУБД

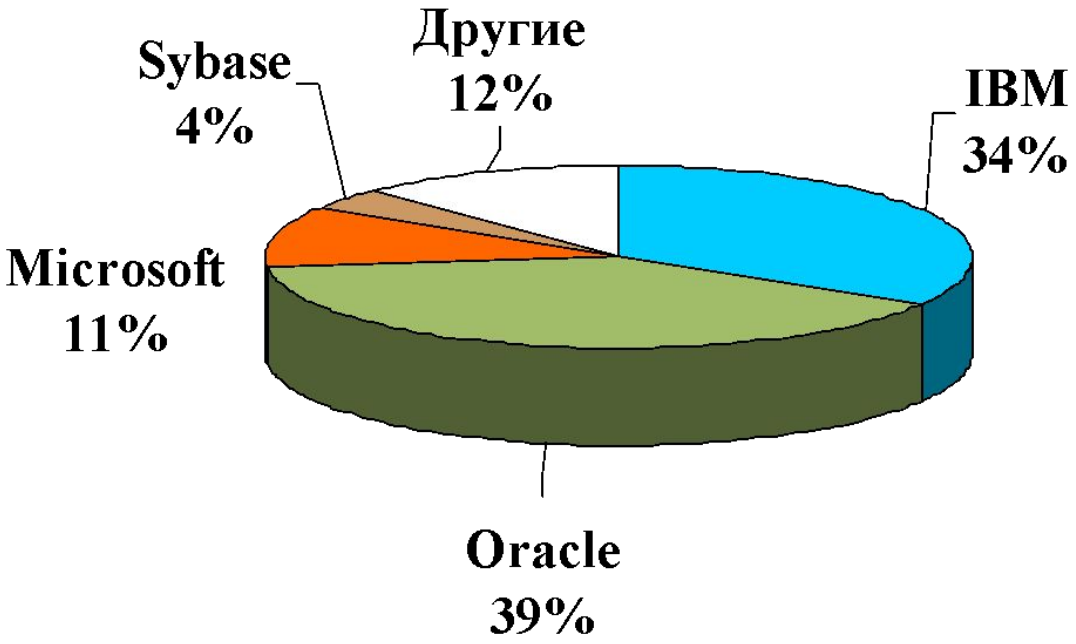
Прочитав в 1976 г. отчет о System/R, Ларри Эллисон (Ellison, Lawrence; p. 1944) организовал в 1977 фирму Relation Software Incorporated. Система Oracle v.2 для PDP-11 была создана в 1979. Фирма, переименованная в Oracle, к концу века превратилась в одну из крупнейших софтверных компаний мира.



Доходы, млрд. $

3.3. Системы управления базами данных

Основные типы и история развития СУБД



Современный мировой рынок СУБД составляет почти 10 млрд. долл. и сильно монополизирован. Около 85% его занимают три СУБД – Oracle, IBM DB2, MS SQL Server (данные 2002 г).

3.4. Пакеты прикладных программ для персональных компьютеров

Программы как товар массового спроса

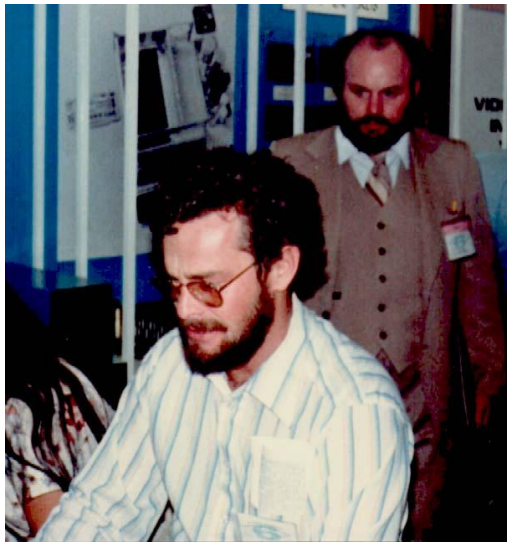
Компьютерный магазин в начале 1980-х годов

3.4. Пакеты прикладных программ для персональных компьютеров

Обработка текстов

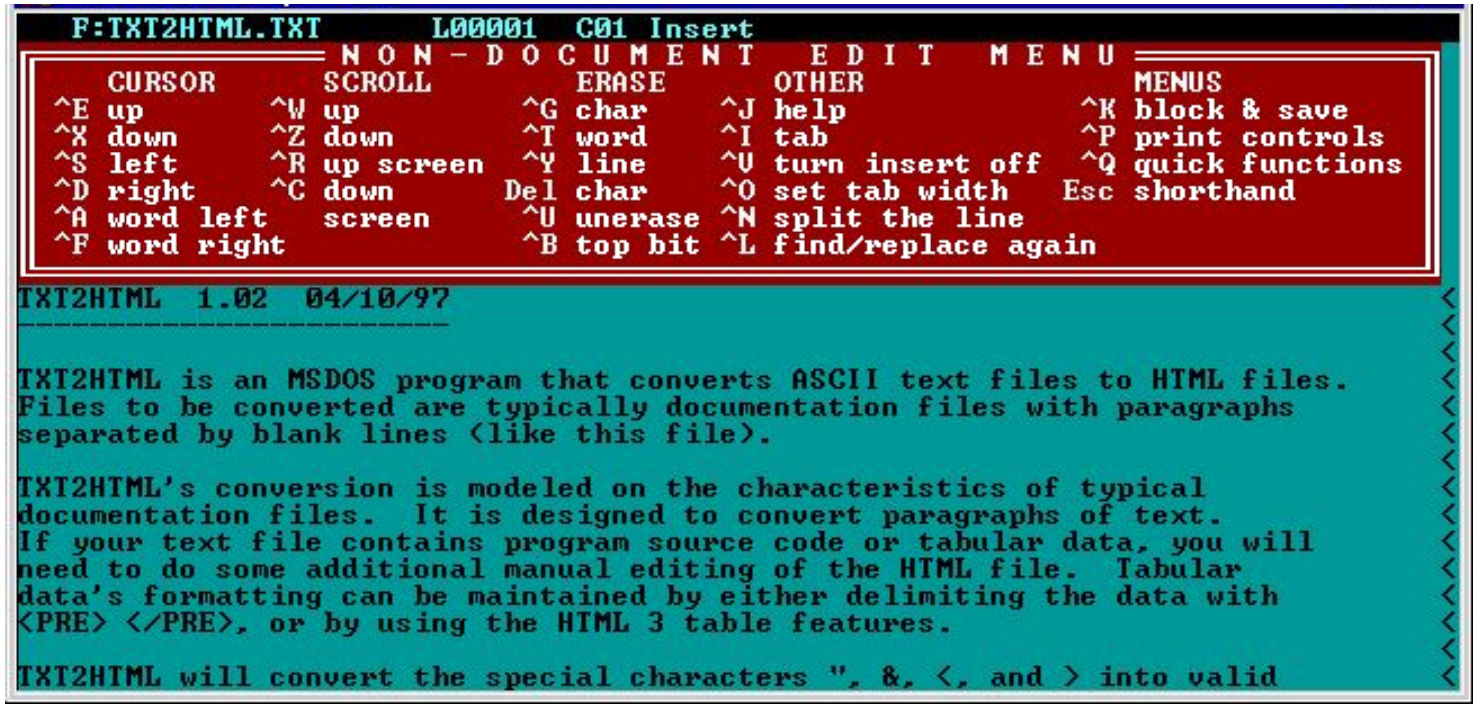
В 1978 году за разработку текстового редактора взялся Сеймур Рубенштейн (Rubenstein, Seymour) из фирмы IMSAI. Прежде чем создавать пакет, он собрал агентов по продаже программного обеспечения и выяснил, что хочет от программного продукта будущий пользователь.

После этого Рубенштейн пригласил талантливого программиста Роба Барнэби (Barnaby, Rob) и сформулировал ему четкое техническое задание.



3.4. Пакеты прикладных программ для персональных компьютеров

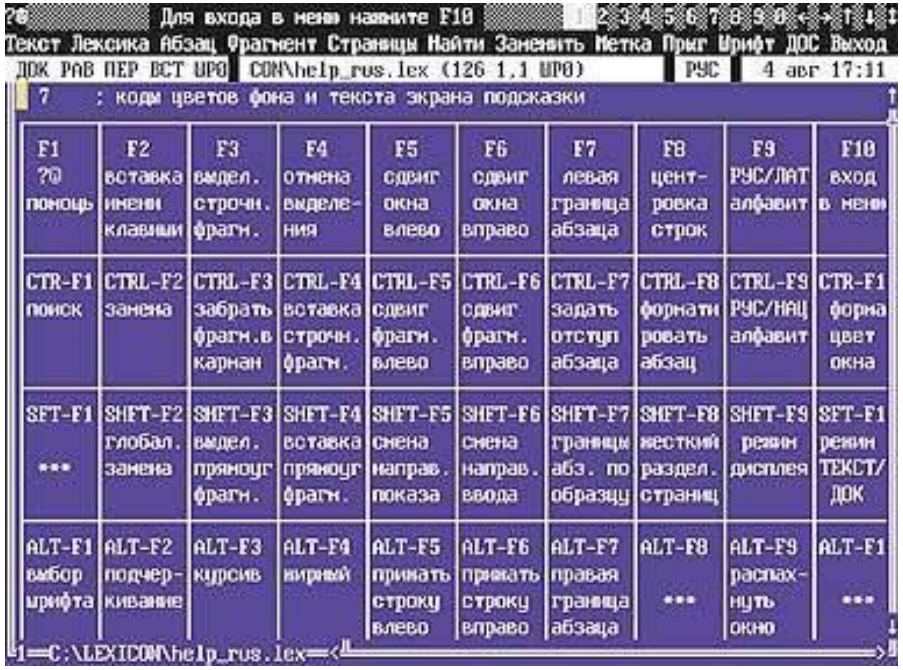
Обработка текстов



В результате появился WordStar — текстовый процессор, ставший фактическим стандартом для нескольких поколений последующих разработок

3.4. Пакеты прикладных программ для персональных компьютеров

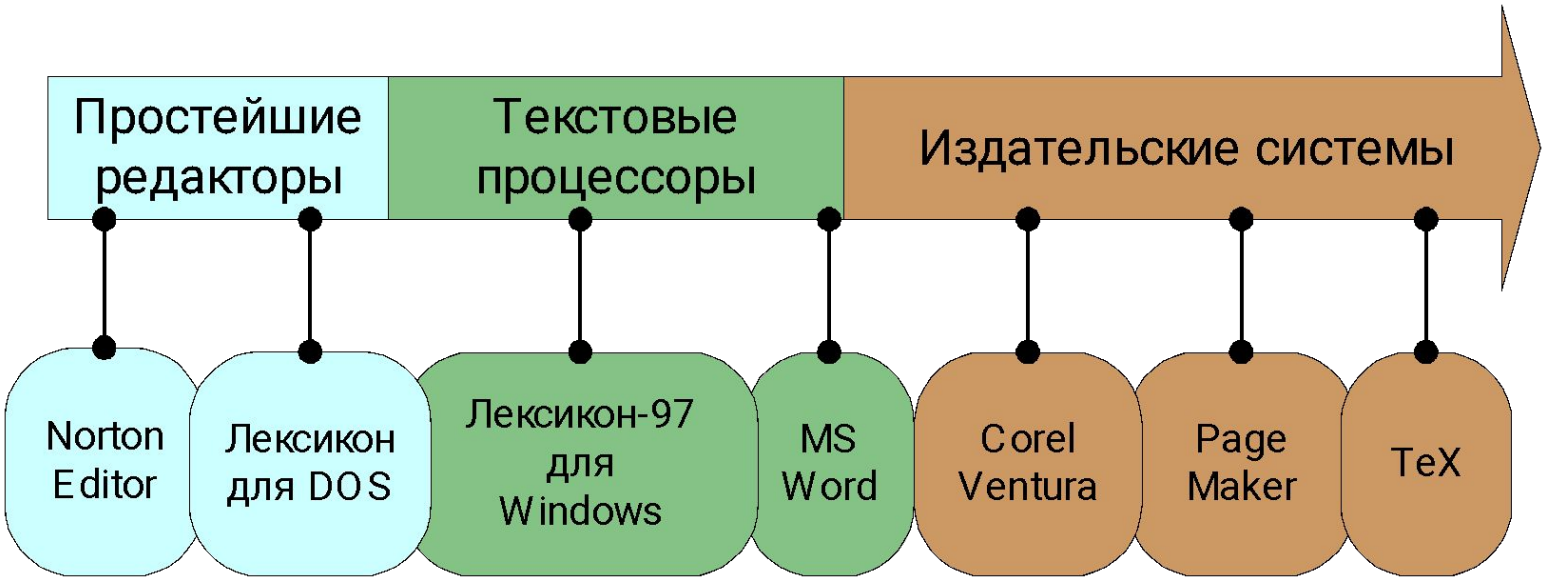
Обработка текстов



Популярнейшим русским текстовым редактором в эпоху DOS был «Лексикон», разработанный в 1985 г. сотрудником Вычислительного центре Академии наук СССР Евгением Веселовым. Первые версии распространялись бесплатно

3.4. Пакеты прикладных программ для персональных компьютеров

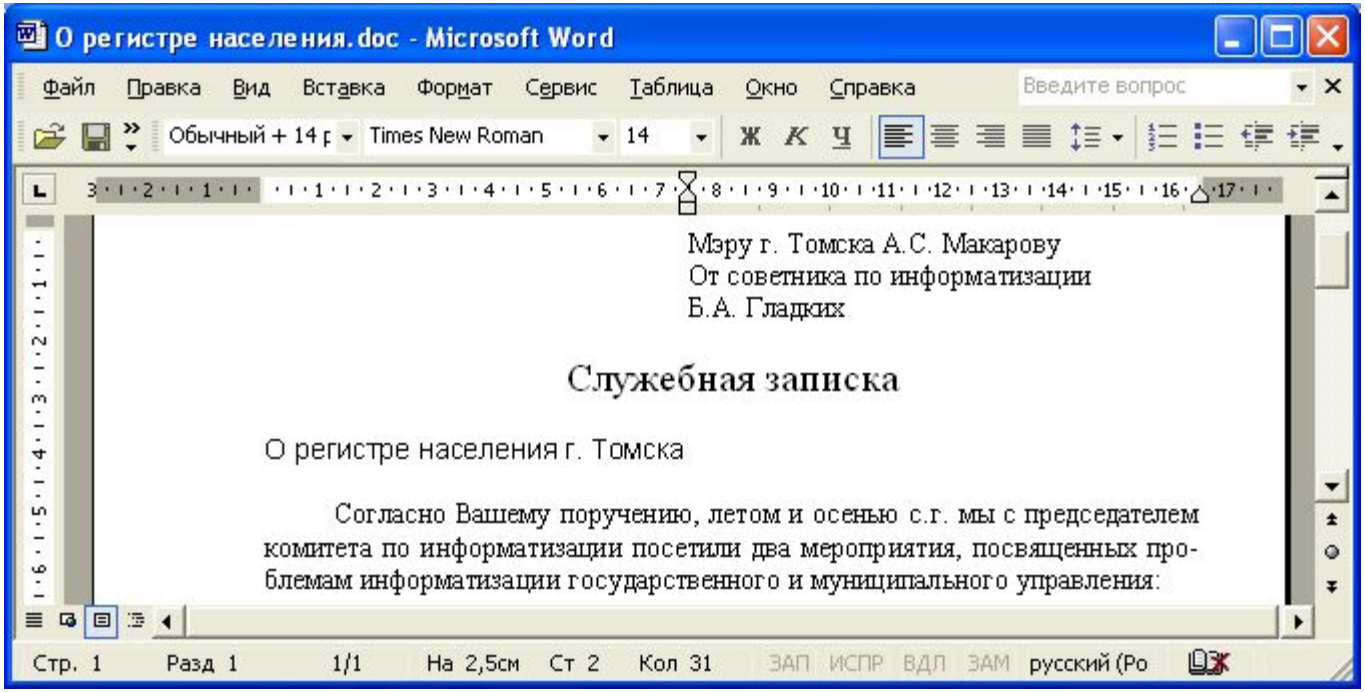
Обработка текстов



Рынок текстовых процессоров

3.4. Пакеты прикладных программ для персональных компьютеров

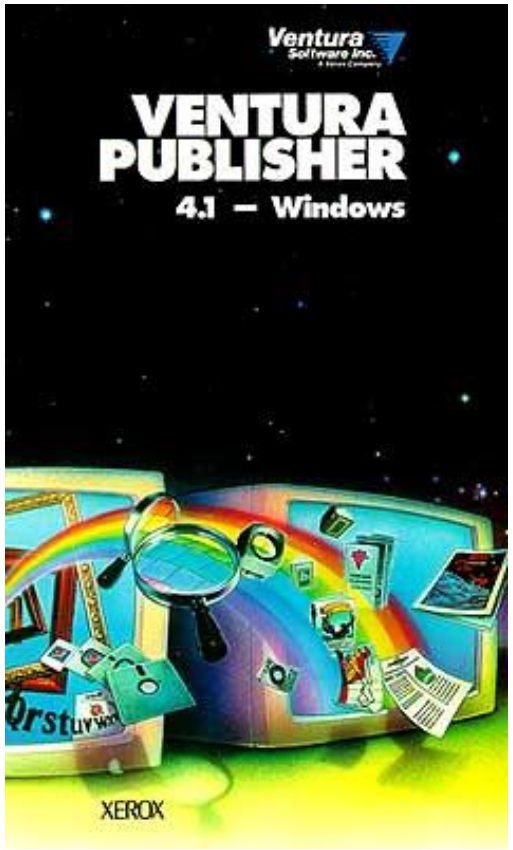
Обработка текстов



*Текстовые процессор*ы предназначены для подготовки документов в режиме *WYSIWIG — What You See Is What You Get* — Что видите, то и получаете. Лидерство в данном секторе рынка удалось захватить Microsoft с пакетом MS Word. Версия 1.0, работающая на платформах Intel и Macintosh была выпущена в 1983 году, с тех пор пакет постоянно совершенствуется

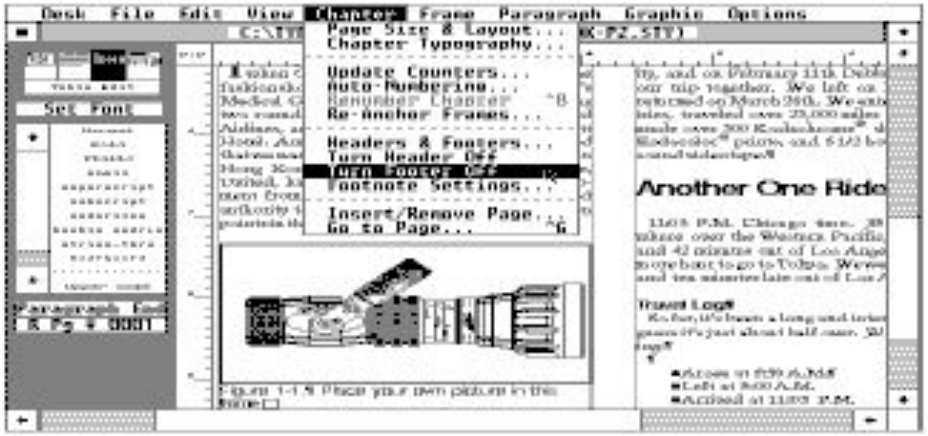
3.4. Пакеты прикладных программ для персональных компьютеров

Обработка текстов



Высший сектор рынка составляют *издательские системы*, предназначенные для подготовки оригинал-макетов печатных изданий (технология *desktop publishing – DTP*).

Первая полнофункциональная издательская система Ventura Publisher была разработана в Xerox PARC, но не получила там коммерческого развития



3.4. Пакеты прикладных программ для персональных компьютеров

Обработка текстов

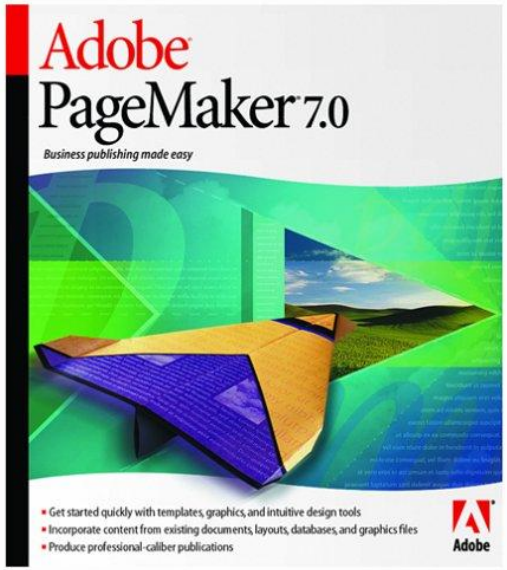
Стив Джобс (слева) с Чарльзом Гешке и Джоном Варноком (1982 г.)

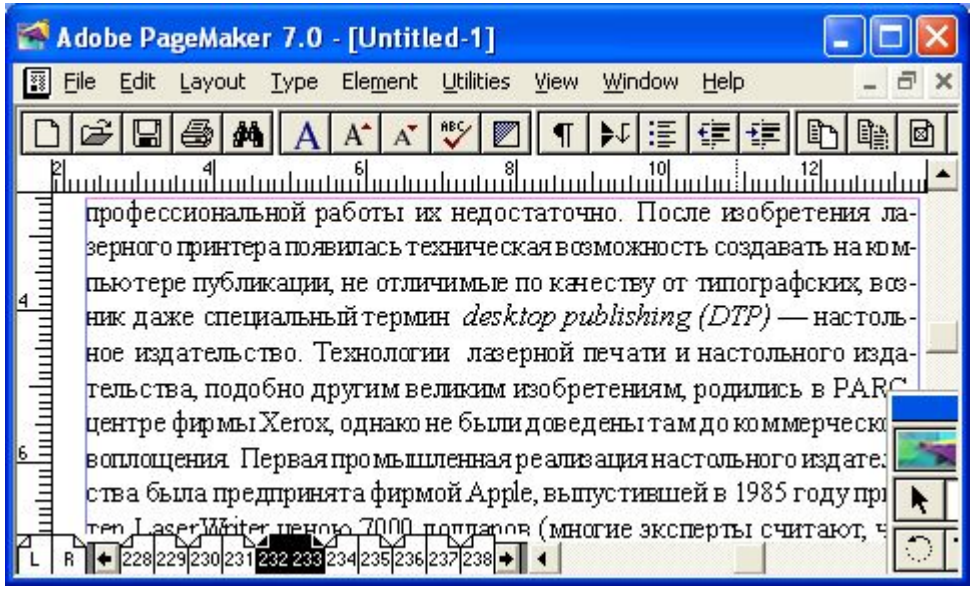
Лидером рынка ПО desktop publishing является компания Adobe Systems Inc., образованная бывшими сотрудниками PARC Джоном Варноком (Warnock, John) и Чарльзом Гешке (Geschke, Charles).

Штаб-квартира фирмы Adobe в г. Сан-Хосе

3.4. Пакеты прикладных программ для персональных компьютеров

Обработка текстов



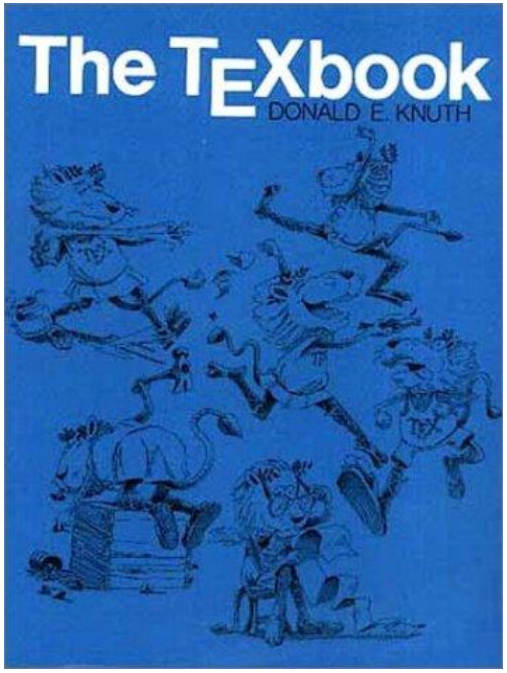


Для программного обеспечения издательского процесса Пол Бренер (Brainerd, Paul) в 1985 г. разработал пакет PageMaker, учитывающий все тонкости пятисотлетнего искусства полиграфии. Основанная Бренером фирма Aldus (название дано в честь средневекового венецианского издателя по имени Aldus Manutius, 1452-1516) в 1994 году присоединилась к компании Adobe Systems Inc.

3.4. Пакеты прикладных программ для персональных компьютеров

Обработка текстов

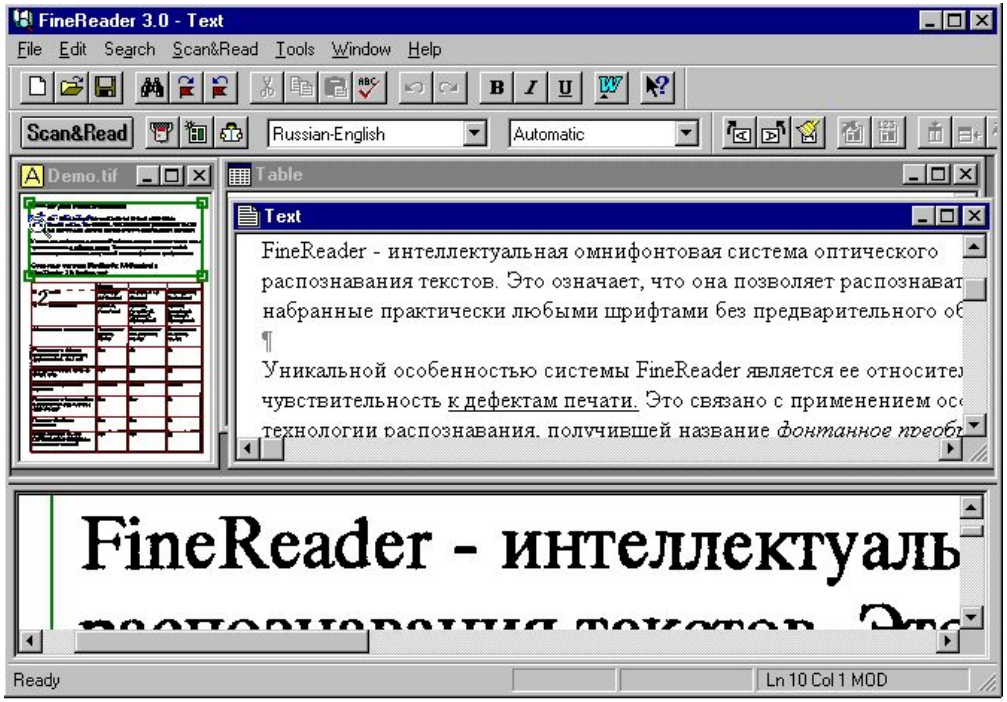
Знаменитый специалист в области информатики Дональд Кнут (Knuth, Donald; р. 1938) из Стенфордского университета в 1977 – 1987 годах создал и подарил мировому сообществу компьютерную систему TeX, которая синтезирует математические тексты любой сложности на уровне высоко-профессиональ-ного типографа.



Обложка книги с описанием системы TeX, 1984 г. (в русском переводе 1993 г. книга называется «Все про TeX»)

3.4. Пакеты прикладных программ для персональных компьютеров

Обработка текстов



Отечественная система распознавания текстов (*Optical Character Recognition — OCR*) Fine Reader компании ABBYY основана на глубоких математических исследованиях в области распознавания образов и получила мировое признание

3.4. Пакеты прикладных программ для персональных компьютеров

Электронные таблицы

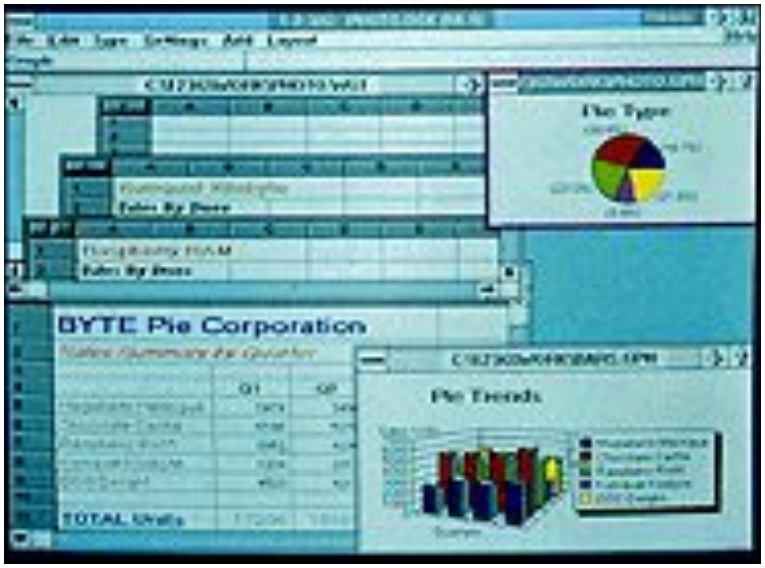
Автором идеи электронной таблицы был сотрудник DEC Дэниэл Бриклин (Bricklin, Daniel), на фото сидит

Вместе программистом Робертом Фрэнкстоном (Frankston, Robert) весной 1979 года они написали программу для Apple II, названную VisiCalc — Visible Calculator.

3.4. Пакеты прикладных программ для персональных компьютеров

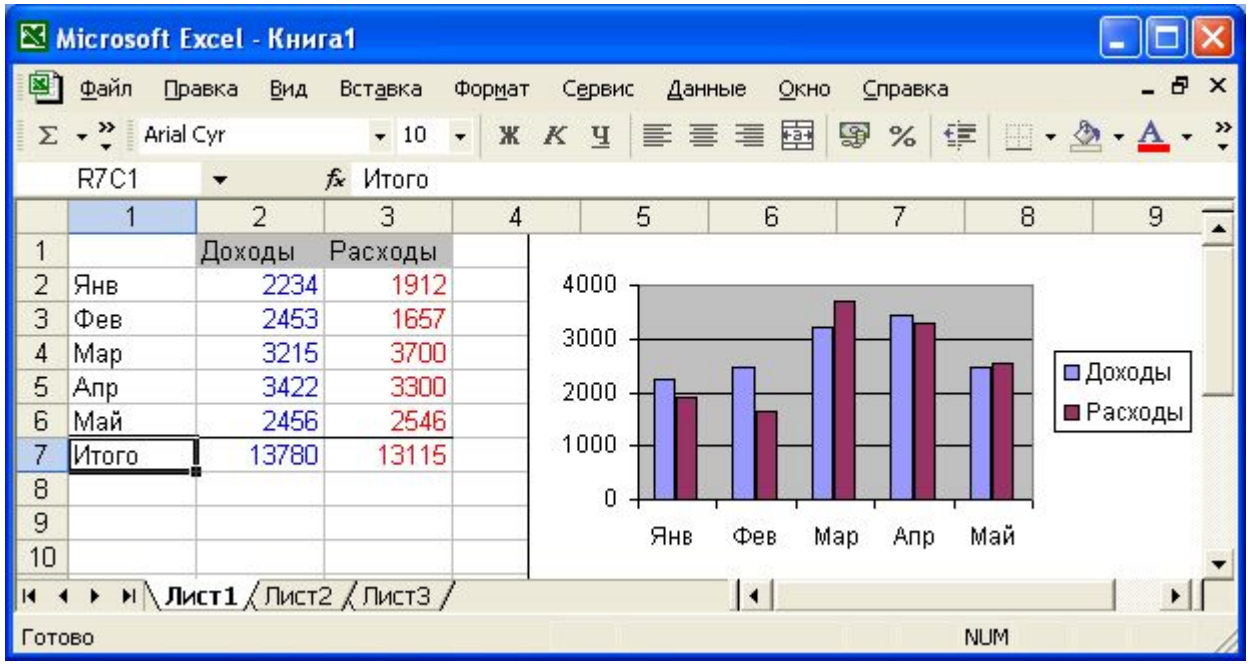
Электронные таблицы

Лидером рынка электронных таблиц для MS DOS был пакет Lotus 1-2-3 компании Lotus Development, образованной в 1982 году Митчелом Кэпором (Kapor, Mitchel) в 1982 г.



3.4. Пакеты прикладных программ для персональных компьютеров

Электронные таблицы



В 1984 году в соревнование с другими производителями вступила Microsoft с табличным процессором MultiPlan для IBM PC. Впоследствии он был существенно доработан в сторону графического интерфейса и получил название Excel. В 1987 году вышла версия Excel для Macintosh, а в 1990 году — для Windows. В течение нескольких последующих лет он отвоевал рынок у Lotus 1-2-3

3.4. Пакеты прикладных программ для персональных компьютеров

Настольные СУБД

В 1981 году инженер NASA Уэйн Рэтлифф (Ratliff, Wayne), на снимке слева, сделал простенькую СУБД для ведения футбольной статистики.

Глава фирмы Ashton-Tate Джордж Тейт (Tate, George), на снимке справа, выкупил авторские права и выпустил пакет в продажу под названием dBASE II

3.4. Пакеты прикладных программ для персональных компьютеров

Интегрированные системы

Став одним из лидеров рынка, фирма Ashton-Tate выпустила в 1984 году интегрированный пакет Framework. Еще до появления Windows, под управлением DOS, Framework моделировал графическую оконную среду