

# Лекция 1

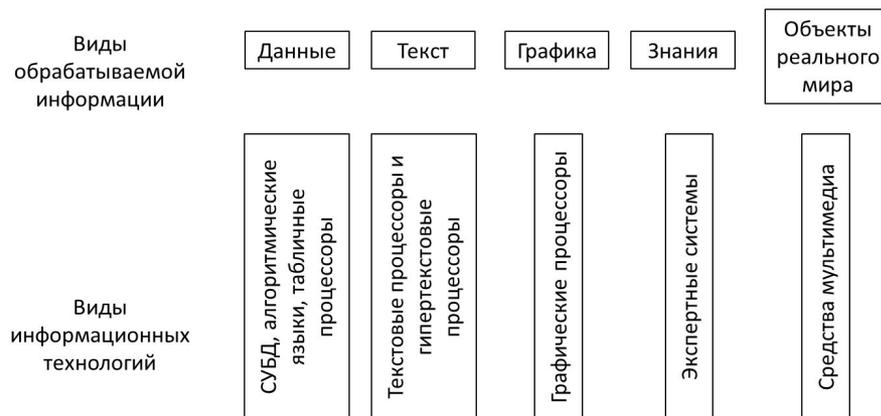
## Базовые понятия информационных технологий

**Информационные технологии** (ИТ, англ. information technology, IT) — широкий класс дисциплин и областей деятельности, относящихся к технологиям создания, сохранения, управления и обработки данных, в том числе с применением вычислительной техники. В последнее время под информационными технологиями чаще всего понимают **компьютерные технологии**. В частности, ИТ имеют дело с использованием компьютеров и программного обеспечения для создания, хранения, обработки, ограничения к передаче и получению информации.

© Wikipedia

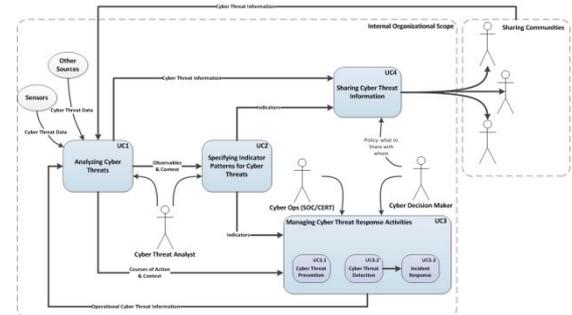


Классификация информационных технологий в зависимости от вида обрабатываемой информации

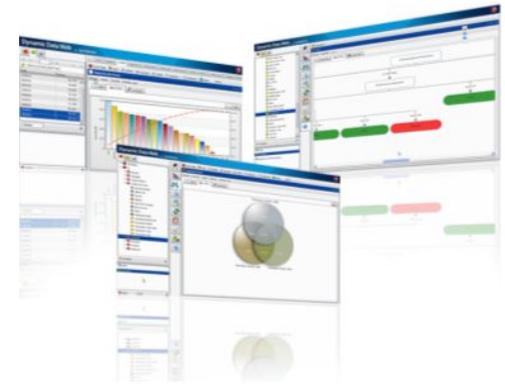


# Основные черты современных ИТ

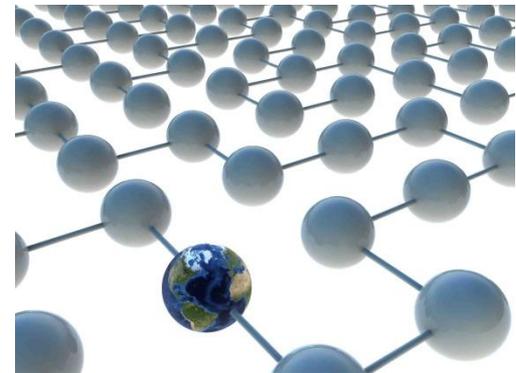
1. Структурированность стандартов цифрового обмена данными и алгоритмов



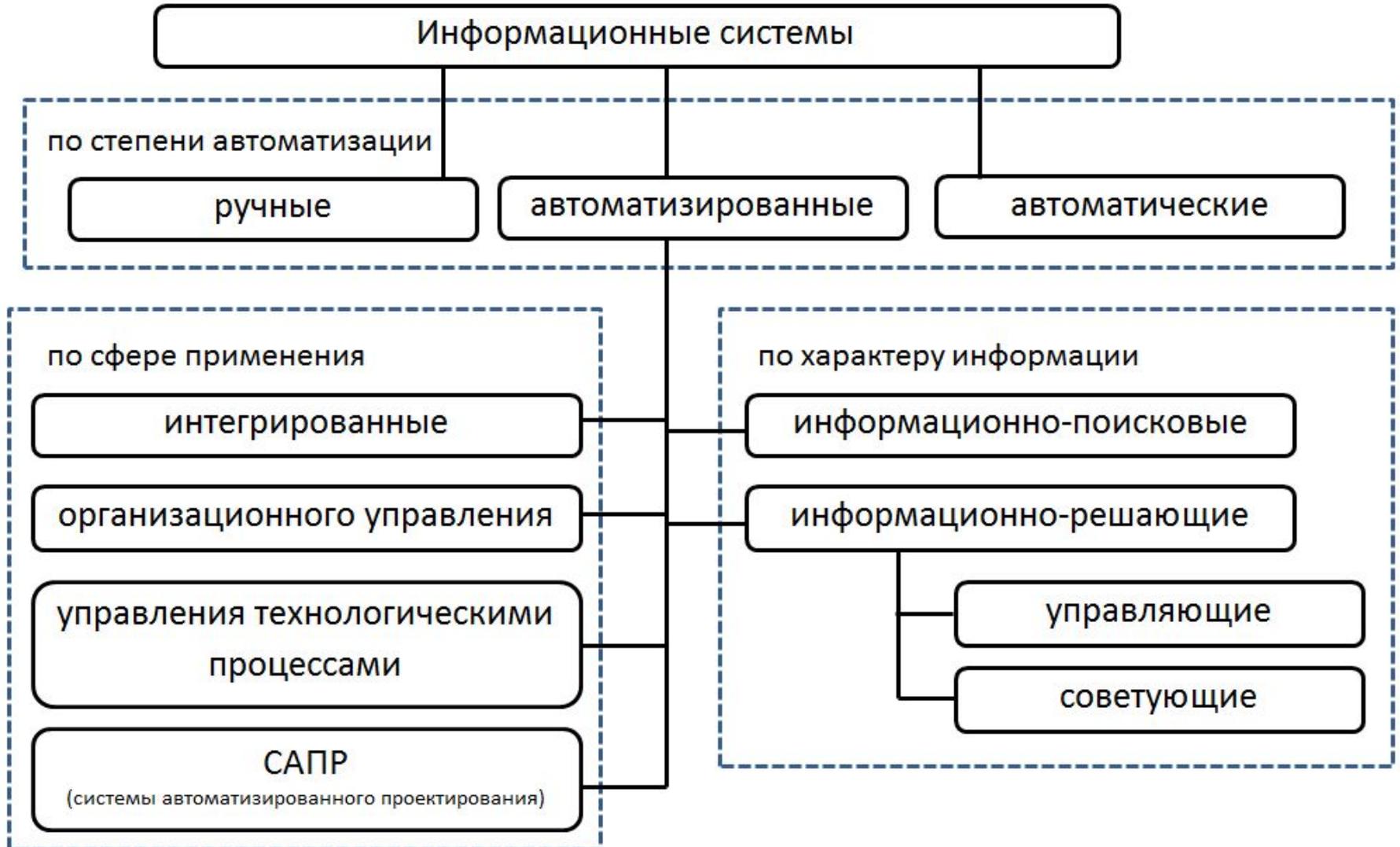
2. Широкое использование компьютерного хранения и предоставление информации в необходимом виде



3. Передача информации посредством цифровых технологий на практически безграничные расстояния



# Классификация современных информационных систем



# Классификация информационных

## Информационные технологии

Степень централизации технологического процесса

Централизованная

Децентрализованная

Комбинированная

Тип предметной области

Бухгалтерский учет

Налоговый учет

Управленческий учет

Аудит

.....

Степень охвата задач управления

Автоматизированная обработка данных

Автоматизация функций управления

Поддержка принятия решений

Электронный офис

Экспертная поддержка

Тип пользовательского интерфейса

Пакетный

Диалоговый

Сетевой

Способ построения сетевой структуры

Локальная

Иерархическая

Распределенная

# Архитектура информационной



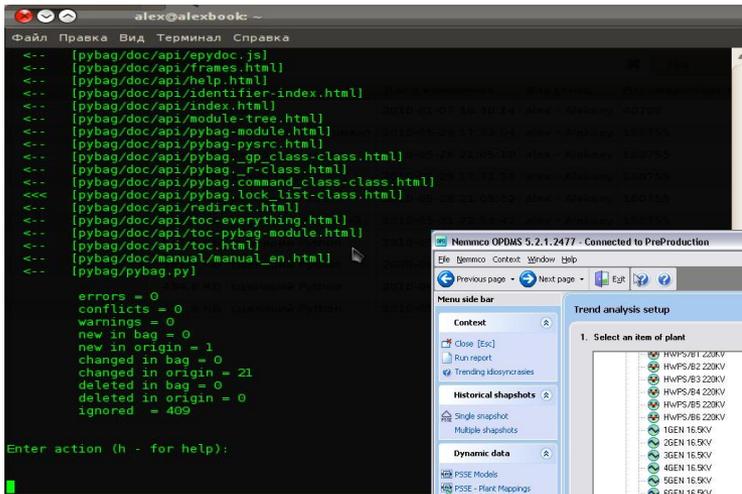
# Классификация информационных технологий в зависимости от вида обрабатываемой информации



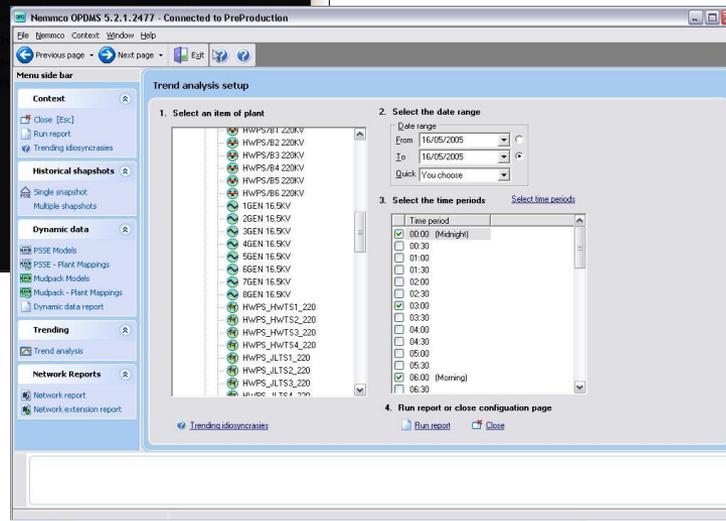
# Классификация информационных технологий

## по виду пользовательского интерфейса

- 1 поколение: командный интерфейс (диалоговый или пакетный режим)
- 2 поколение: WIMP-интерфейс (**W**indow / **I**mage / **M**enu / **P**ointer)
- 3 поколение: SILK-интерфейс (**S**peech / **I**mage / **L**anguage / **K**nowledge)

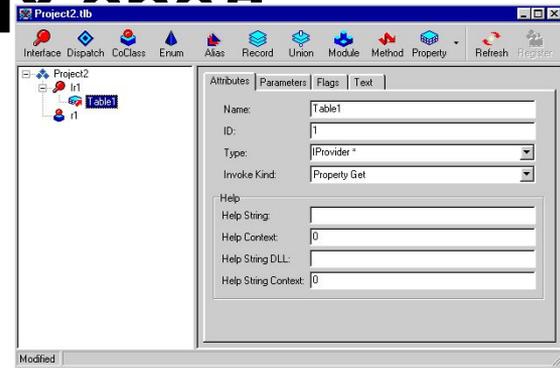


```
alex@alexbook: ~  
Файл Правка Вид Терминал Справка  
<-- [pybag/doc/api/epydoc.js]  
<-- [pybag/doc/api/frames.html]  
<-- [pybag/doc/api/help.html]  
<-- [pybag/doc/api/identifier-index.html]  
<-- [pybag/doc/api/index.html]  
<-- [pybag/doc/api/module-tree.html]  
<-- [pybag/doc/api/pybag-module.html]  
<-- [pybag/doc/api/pybag-pysrc.html]  
<-- [pybag/doc/api/pybag_gp_class-class.html]  
<-- [pybag/doc/api/pybag_r_class.html]  
<-- [pybag/doc/api/pybag_command_class-class.html]  
<<< [pybag/doc/api/pybag_lock_list-class.html]  
<-- [pybag/doc/api/redirect.html]  
<-- [pybag/doc/api/toc-everything.html]  
<-- [pybag/doc/api/toc-pybag-module.html]  
<-- [pybag/doc/api/toc.html]  
<-- [pybag/doc/manual/manual_en.html]  
<-- [pybag/pybag.py]  
  
errors = 0  
conflicts = 0  
warnings = 0  
new in bag = 0  
new in origin = 1  
changed in bag = 0  
changed in origin = 21  
deleted in bag = 0  
deleted in origin = 0  
ignored = 499  
  
Enter action (h - for help):
```



# Классификация современных вычислительных сред

- Традиционные  
(Traditional computing)
- WEB-ориентированные  
(Web-Based Computing)
- Встроенные  
(Embedded Computing)



# Формы представления

Область ИТ

Научно-методическая сфера

Производственная сфера и рынок

Базовые концепции

- Глобальная информационная инфраструктура
- Концепция открытых систем
- Взаимосвязь открытых систем
- Открытая распределенная обработка
- ...

Методическая база

- Термины и модели
- Методы спецификации
- Языки и парадигмы программирования

- стандартизация
  - профилирование
  - таксономия
  - конформность
- универсальные методы*

Информативная база (стандарты)

- Эталонные модели
- Базовые спецификации
- Профили (OSI, API)
- OSE-профили
- Стратегические профили
- Сценарии

Индустрия продуктов ИТ

Индустрия услуг ИТ

Маркетинг ИТ

произдуцирует

произдуцирует

Пространство спецификаций

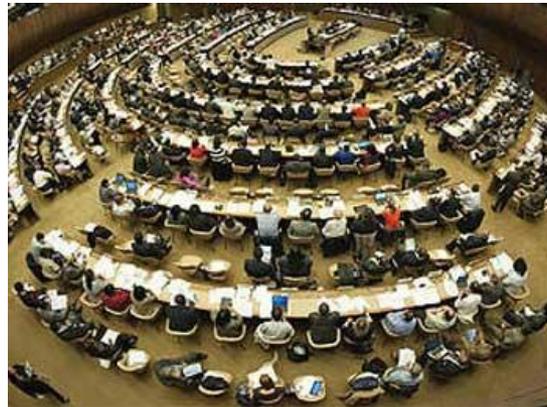
- Спецификации ИТ (IT specification)
- Профили ИТ (IT profiles)
- Сценарии ИТ (IT scenarios)
- ...

Пространство ИТ-продуктов (ИТ-систем)

- Продукты ИТ (IT products)
- Системы ИТ (IT systems)
- Протоколы ИТ (IT protocols)
- Сервисы ИТ (IT services)

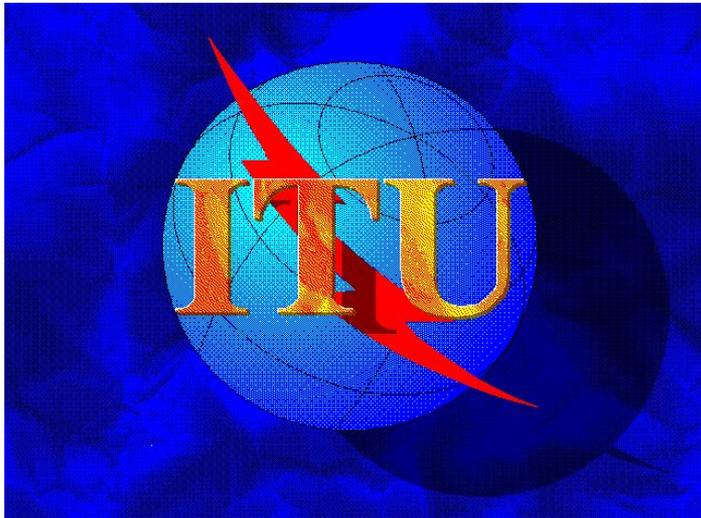
# Базовые понятия современных информационных технологий

**1. Стандарт** (по определению ISO). Технический стандарт или другой документ, доступный и опубликованный, коллективно разработанный или согласованный и общепринятый в интересах тех, кто им пользуется, основанный на интеграции результатов науки, технологии, опыта, способствующий повышению общественного блага и принятый организациями, признанными на национальном, региональном и международном уровнях.



# Базовые понятия современных информационных технологий

2. **Базовый стандарт** (base standard), также иногда используются термины **формальный стандарт** или **стандарт de-ure**. **Международный стандарт**, принятый ISO (международной организацией по стандартизации), или **рекомендация** организации ITU-T (до 1993 г. - CCITT) - международного союза по телекоммуникации.

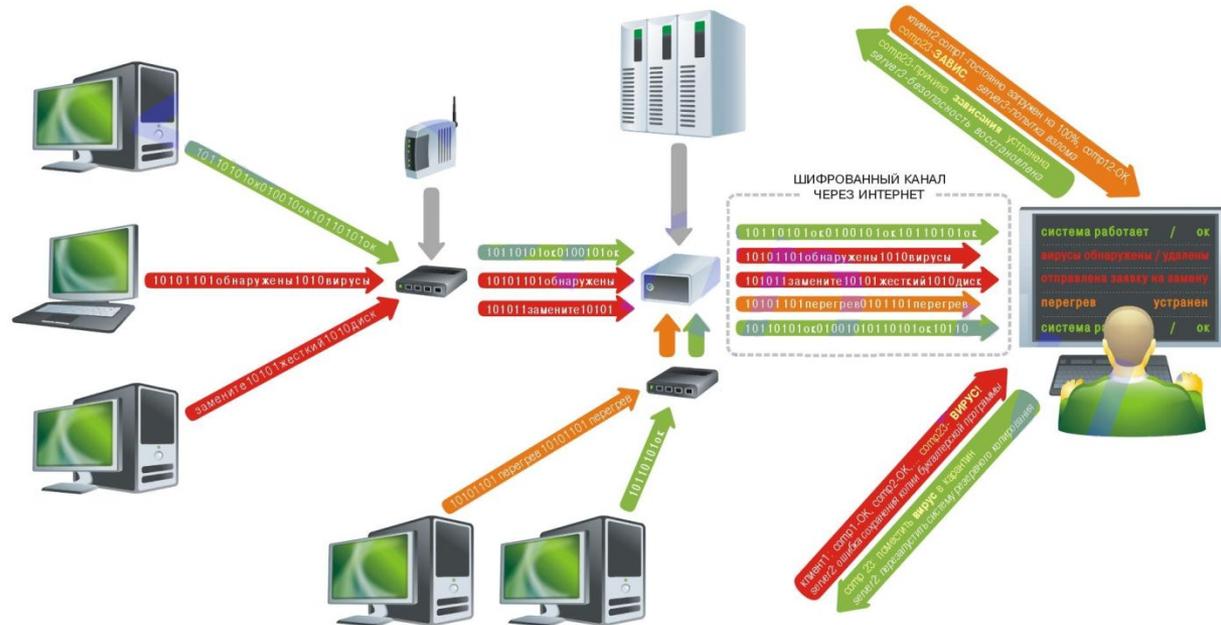


	8	7	6	5	4	3	2	1	
0	Идентификатор информационного элемента								
	0	0	1	1	0	0	0		Байт 1
	Длина содержимого								Байт 2
1 Ext	Идентификация интерфейса	Тип интерфейса	0 Резервный	Идентификатор предпочтения	Индикация D-канала	Выбор информационного канала			Байт 3
1 Ext	Стандарт кодирования		Номер Таблица		Тип канала/тип элемента таблицы				Байт 3.2
	Номер канала								Байт 3.3



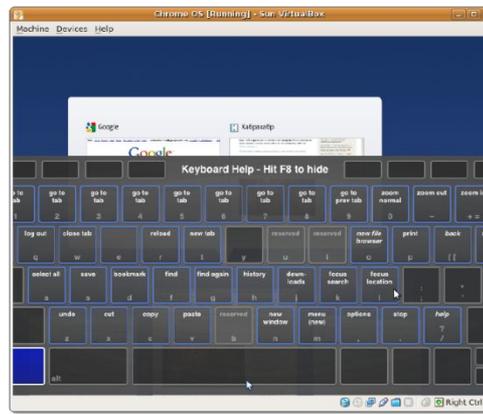
# Базовые понятия современных информационных технологий

**4. Система ИТ или ИТ-система (IT system)** (или по тексту просто система, если это не вызывает двусмысленности). Совокупность ресурсов информационных технологий, предоставляющая сервис (услуги) на одном или большем числе интерфейсов в соответствии с заданными ИТ-спецификациями



# Базовые понятия современных информационных технологий

5. **OSE** (Open Systems Environment - Окружение или среда открытых систем). Исчерпывающий набор интерфейсов, сервисов, форматов, а также пользовательских аспектов, которые обеспечивают **интероперабельность** и/или **переносимость** приложений (программ), данных, людей в контексте заданных базовых стандартов и профилей ИТ. (Еще одним важным и почти обязательным свойством открытости является свойство **масштабируемости ИТ**).



# Базовые понятия современных информационных технологий

**5.1 Переносимость** (portability). Свойство ИТ-системы, способствующее уменьшению накладных расходов, связанных с переносом прикладного программного обеспечения и данных (а также пользователей) с одной системы ИТ на другую.

**5.2 Интероперабельность** (interoperability). Способность ИТ-систем обмениваться информацией друг с другом и совместно использовать информацию, которой они обмениваются.

**5.3 Масштабируемость** (scalability). Свойство системы, позволяющее ей эффективно работать в широком диапазоне параметров, определяющих технические и ресурсные характеристики системы и/или поддерживающей среды (примерами таких характеристик могут

# Базовые понятия современных информационных технологий

**6. PAS** (Publicly Available Specifications - Общедоступные спецификации). По существу это хорошо отработанные спецификации, как правило, являющиеся стандартам de-facto, которые принимаются ISO в качестве кандидатов в международные стандарты для проведения соответствующих процедур по их стандартизации. Важным требованием к PAS является то, что их сопровождение осуществляется известными профессиональными организациями посредством прозрачного публичного процесса, основанного на консенсусе. (Примерами PAS могут служить спецификации организации OSF, известные под наименованием DCE и спецификации Java-технологий фирмы Sun).

# Базовые понятия современных информационных технологий

**7. RFC** (Request For Comments) - документ из серии пронумерованных информационных документов Интернета, содержащих технические спецификации и стандарты, широко применяемые во Всемирной сети. Название «Request for Comments» ещё можно перевести как «заявка (запрос) на отзывы» или «тема для обсуждения». В настоящее время первичной публикацией документов RFC занимается IETF под эгидой открытой организации Общество Интернет (англ. Internet Society, ISOC). Правами на RFC обладает именно Общество Интернет.

# Базовые понятия современных информационных технологий

**8. Профиль (Profile)** - набор, состоящий из одного или большего числа базовых стандартов и/или стандартизованных на международном уровне профилей (ISPs), содержащий указание области применимости, а также указание выбранных классов сервиса, опций и параметров, тестовых наборов, относящихся к входящим в состав профиля спецификациям. Таким образом, профиль задает некоторую функциональность, необходимую для выполнения конкретной прикладной задачи, агрегируя избирательным образом функциональные возможности стандартизованных спецификаций (стандартов или ISPs).

# Базовые понятия современных информационных технологий

**8.1 ISP** (International Standardized Profile - Международный стандартизованный профиль). Официально принятый и согласованный на международном уровне документ, описывающий один или несколько профилей.

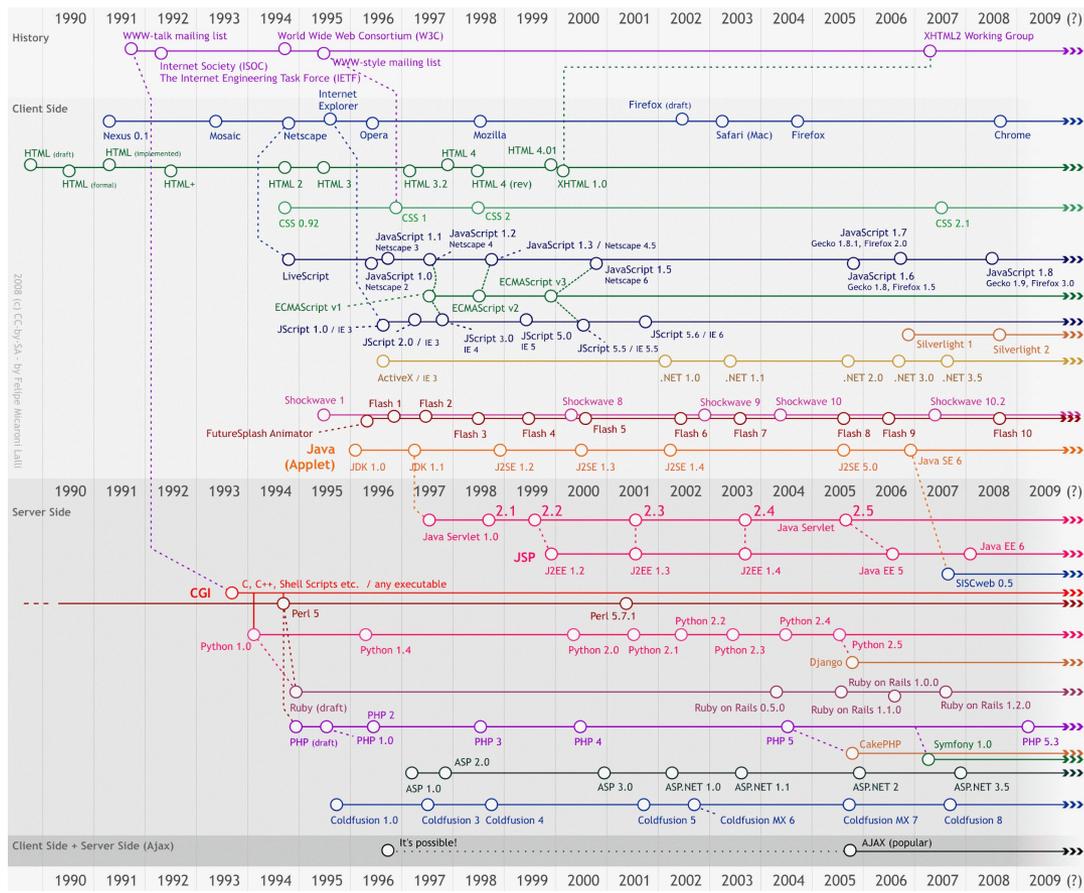
**8.2 OSE-профиль** (OSE-profile). Профиль, который специфицирует все поведение ИТ-системы или часть ее поведения на одном или большем числе интерфейсов OSE.

**8.3 OSI-профиль** (OSI-profile). Профиль, составленный из базовых спецификаций, соответствующих модели RM OSI, возможно дополненных базовыми стандартами и/или профилями для представления обмениваемых данных и их форматов (так называемыми F-профилями).

**8.4 API-профиль** (API-profile). Профиль, определяющий конкретную комбинацию базовых спецификаций прикладного пользовательского интерфейса в соответствии с моделью RM OSE, возможно дополненных базовыми стандартами и/или профилями для представления данных и их форматов.

# Базовые понятия современных информационных технологий

9. Таксономия (Taxonomy). Классификационная схема, применяемая для однозначной идентификации профилей или наборов профилей.





# Перспективные информационные

## технологии

Искусственный интеллект

Погружение в виртуальную  
реальность

Мобильная связь (4G)

Беспроводная связь

Машинный перевод

Машинное зрение

Твердотельные накопители

Стереодисплеи

3D-принтеры



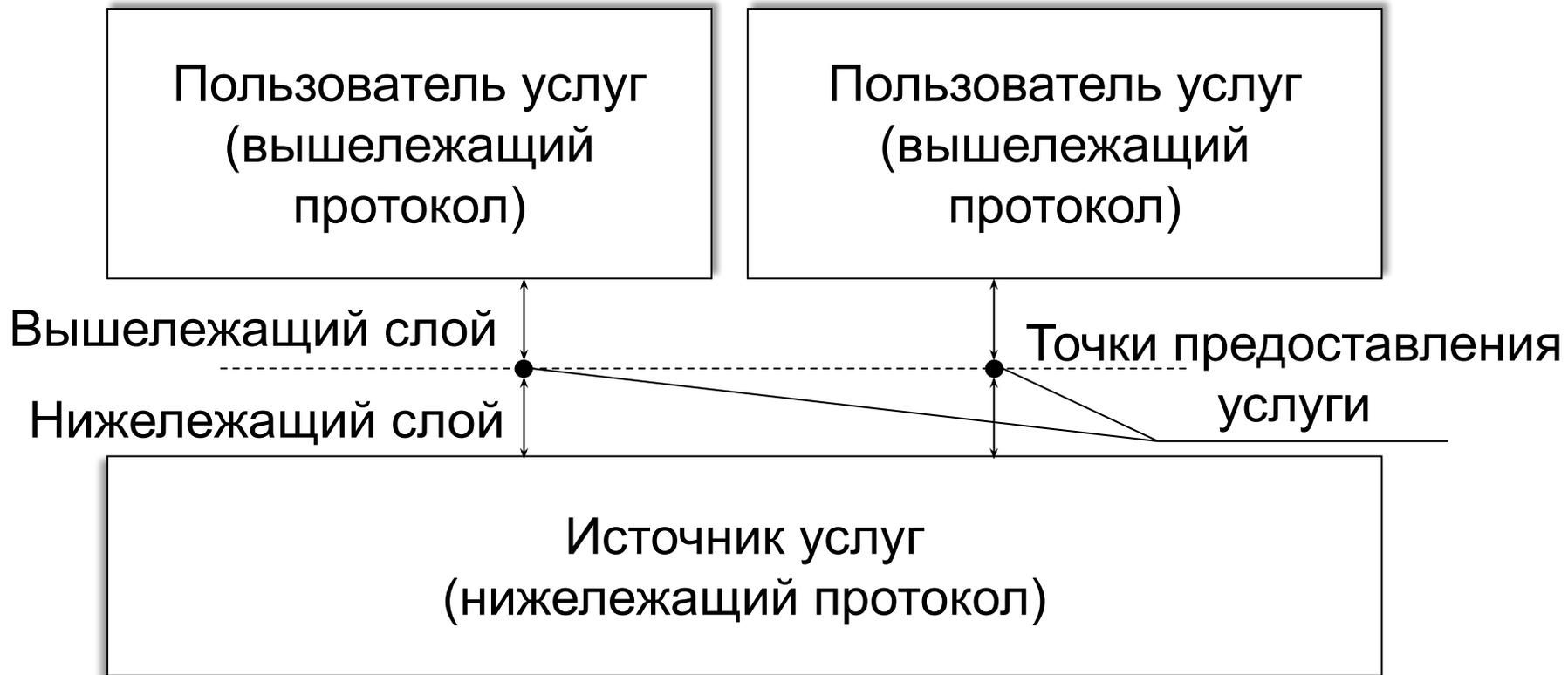
# Лекция 2

Современные технологии и  
интерфейсы прикладного  
программирования

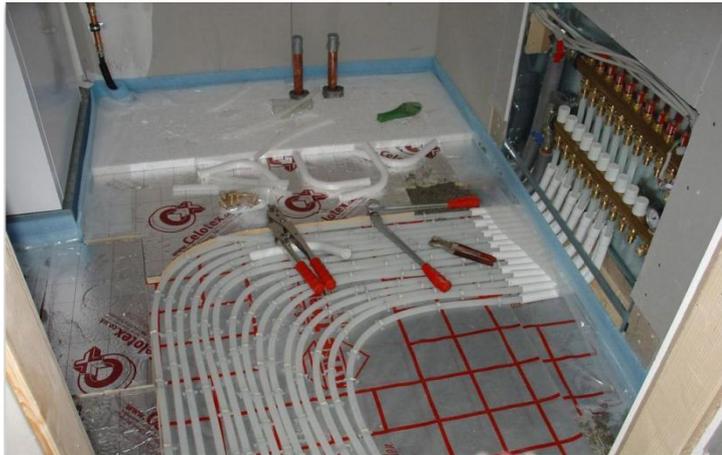




# Структура предоставления функций API



# Типичный функционал ДРІ





# POSIX

POSIX (Portable Operating System Interface for Unix — Переносимый интерфейс операционных систем Unix) — набор стандартов, описывающих интерфейсы между операционной системой и прикладной программой. Стандарт создан для обеспечения совместимости различных UNIX-подобных операционных систем и переносимости прикладных программ на уровне исходного кода, но может быть использован и для не-Unix систем.

Серия стандартов POSIX была разработана комитетом 1003 IEEE. Международная организация по стандартизации (ISO) совместно с Международной электротехнической комиссией (IEC) приняли данный стандарт (POSIX) под названием ISO/IEC 9945.

# Основные задачи серии стандартов POSIX

1. содействовать облегчению переноса кода прикладных программ на иные платформы
2. способствовать определению и унификации интерфейсов заранее при проектировании, а не в процессе их реализации
3. сохранить по возможности и учитывать все главные, созданные ранее и используемые прикладные программы
4. определять необходимый минимум интерфейсов прикладных программ, для ускорения создания, одобрения и утверждения документов
5. развивать стандарты в направлении обеспечения коммуникационных сетей, распределенной обработки данных и защиты информации
6. рекомендовать ограничение использования бинарного (объектного) кода для приложений в простых системах

# Windows API

Windows API — общее наименование целого набора базовых функций интерфейсов программирования приложений операционных систем семейств Microsoft Windows. Является самым прямым способом взаимодействия приложений с Windows. Для создания программ, использующих Windows API, «Майкрософт» выпускает комплект разработчика программного обеспечения, который называется Platform SDK, и содержит документацию, набор библиотек, утилит и других инструментальных средств для разработки.

**SDK – Software Development Kit**

**DDK – Driver Development Kit (устарел) → WDK – Windows Driver Kit**

# Windows API

Windows API

Графический интерфейс

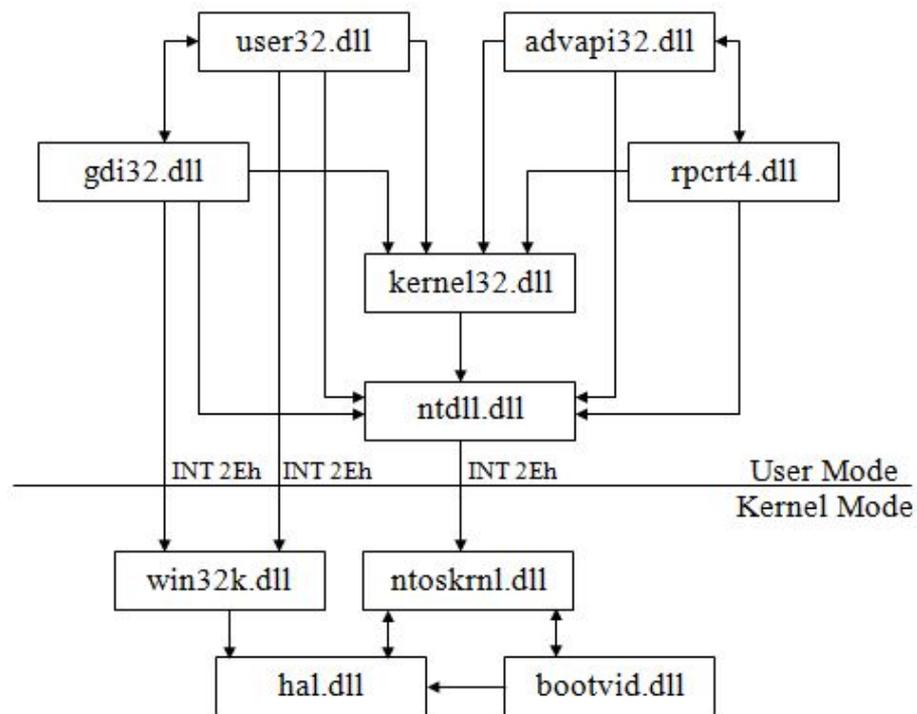
csrss.exe

*Client/Server Runtime Subsystem*

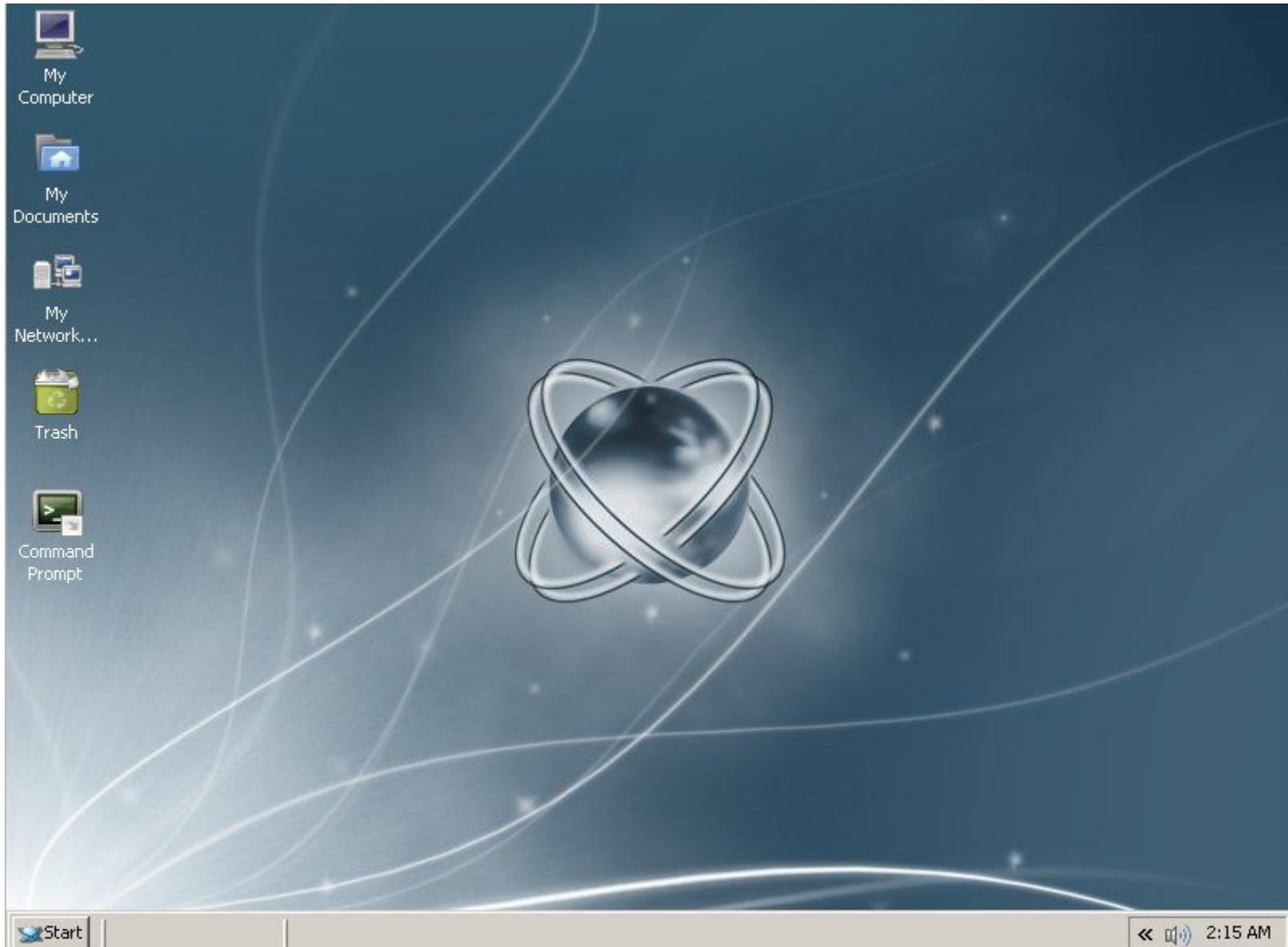
CSRSS выполняется как системный сервис пользовательского режима. Когда процесс пользовательского режима вызывает функцию с участием консольных окон, создания процесса или потока, библиотеки Win32 (kernel32.dll, user32.dll, gdi32.dll) вместо запроса системного вызова обращаются к процессу CSRSS

Функции ядра

win32k.sys

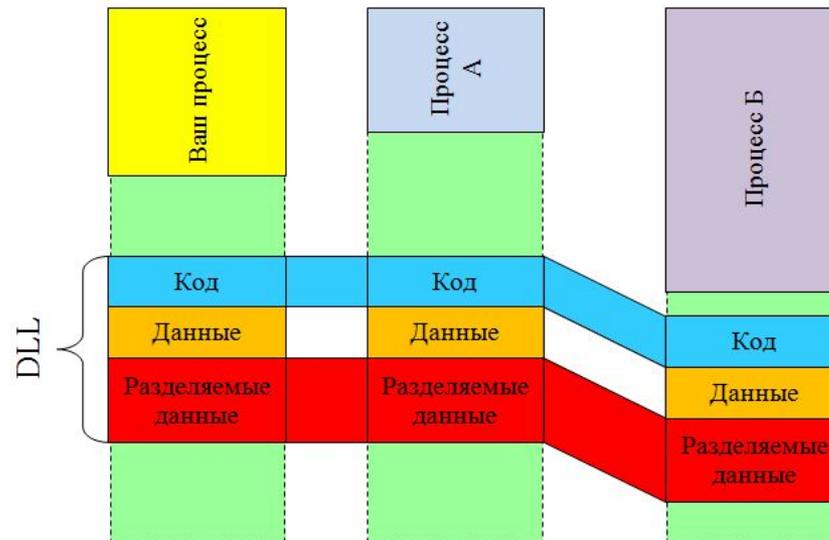


# Проект ReactOS совместим с Windows API



# DLL

DLL (dynamic-link library — «библиотека динамической компоновки», «динамически подключаемая библиотека») — понятие операционной системы Microsoft Windows, динамическая библиотека, позволяющая многократное использование различными программными приложениями. К DLL относятся также элементы управления ActiveX и драйверы. В UNIX аналогичные функции выполняют так называемые общие объекты (англ. shared objects).



# DLL

## Технология DLL

COM – Component Object Model

Microsoft OLE Automation, ActiveX,  
DCOM, COM+, DirectX, XPCOM

### Проблемы COM-технологии:

- «Хрупкий базовый класс»
- «Ад DLL» (DLL hell)

SOM – System Object Model



# Active X

Управляющие элементы ActiveX могут быть использованы для создания распределённого приложения, работающего через браузер.

Управляющие элементы ActiveX сравнимы с технологией Java-апплетов: программисты разрабатывают оба механизма, чтобы браузер мог не только скачать, но и обработать их. Однако, Java-апплеты могут работать под любой платформой, тогда как управляющие элементы ActiveX официально обрабатываются только Microsoft Internet Explorer и операционной системой Microsoft Windows.

**Вредоносное ПО можно случайно установить с вебсайтов злоумышленников, используя технологию управляющих элементов ActiveX.**

# ODBC и производные стандарты

**ODBC** (Open Database Connectivity) — это программный интерфейс доступа к базам данных, разработанный фирмой Microsoft, в сотрудничестве с Simba Technologies на основе спецификаций Call Level Interface (CLI), который разрабатывался организациями SQL Access Group, X/Open и Microsoft.

**ADOdb** — программная библиотека, обеспечивающая прикладной интерфейс доступа к базам данных для языков программирования PHP и Python, основанная на некоторых концепциях Microsoft **ActiveX Data Objects**. Библиотека обеспечивает разработчика приложений абстрактным инструментарием, позволяющим создавать приложения без необходимости программирования поддержки каждого из конкретных возможных типов источников данных. В частности, у разработчиков появляется возможность изменить СУБД без необходимости вносить исправления в программный код.

# ODBC и производные стандарты

**JDBC** (Java DataBase Connectivity — соединение с базами данных на Java) — платформенно-независимый промышленный стандарт взаимодействия Java-приложений с различными СУБД, реализованный в виде пакета `java.sql`, входящего в состав Java.

**BDE** (Borland Database Engine — «движок баз данных Borland») — 32-битный движок баз данных под Microsoft Windows для доступа к базам данных из Borland Delphi, C++ Builder, IntraBuilder, Paradox for Windows и Visual dBASE for Windows. В настоящее время абсолютно устарел.

**dbExpress** — это архитектура создания драйверов данных от компании Embarcadero, которая заменяет устаревший BDE. Впервые была реализована в Borland Delphi 6 и C++Builder 6, и потерпев некоторые переработки, в последний раз использовалась в Delphi XE2.

# Мультимедийные API



Microsoft DirectX — это набор API, разработанных для решения задач, связанных с программированием под Microsoft Windows. Наиболее широко используется при написании компьютерных игр. Пакет средств разработки DirectX под Microsoft Windows бесплатно доступен на сайте Microsoft. Иногда обновленные версии DirectX поставляются вместе с игровыми приложениями.



Microsoft XNA — набор инструментов с управляемой средой времени выполнения (.NET), созданный Microsoft, облегчающий разработку и управление компьютерными играми. XNA стремится освободить разработку игр от написания «повторяющегося шаблонного кода» и объединить различные аспекты разработки игр в одной системе



OpenGL (Open Graphics Library — открытая графическая библиотека, графическое API) — спецификация, определяющая независимый от языка программирования платформонезависимый программный интерфейс для написания приложений, использующих двухмерную и трёхмерную компьютерную графику.

# OpenGL API

OpenGL ориентируется на следующие две задачи:

1. Скрыть сложности адаптации различных 3D-ускорителей, предоставляя разработчику единый API.
2. Скрыть различия в возможностях аппаратных платформ, требуя реализации недостающей функциональности с помощью программной эмуляции.

Основным принципом работы OpenGL является получение наборов векторных графических примитивов в виде точек, линий и многоугольников с последующей математической обработкой полученных данных и построением растровой картинка на экране и/или в памяти. Векторные трансформации и растеризация выполняются графическим конвейером (graphics pipeline), который по сути представляет собой дискретный автомат.

# OpenGL API

OpenGL является низкоуровневым процедурным API, что вынуждает программиста диктовать точную последовательность шагов, чтобы построить результирующую растровую графику (императивный подход). Это является основным отличием от дескрипторных подходов, когда вся сцена передается в виде структуры данных (чаще всего дерева), которое обрабатывается и строится на экране.

С одной стороны, императивный подход требует от программиста глубокого знания законов трёхмерной графики и математических моделей, с другой стороны — даёт свободу внедрения различных инноваций.

# Два слова о шейдерах

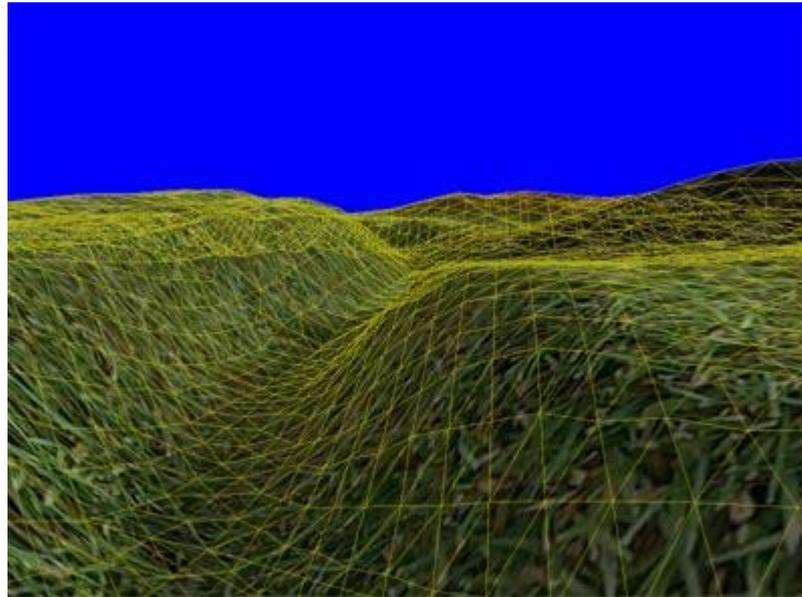
Шейдер (Shader; схема затенения, программа построения теней) — это программа для одной из ступеней графического конвейера, используемая в трёхмерной графике для определения окончательных параметров объекта или изображения. Она может включать в себя произвольной сложности описание поглощения и рассеяния света, наложения текстуры, отражение и преломление, затенение, смещение поверхности и эффекты пост-обработки.

До «изобретения» шейдеров разработчик писал всю цепочку рендеринга. Шейдеры же позволили составлять сложные видеоэффекты из атомарных операций.

# Два слова о шейдерах

## Вершинные шейдеры (Vertex Shader)

Вершинный шейдер оперирует данными, сопоставленными с вершинами многогранников. Вершинный шейдер может быть использован для видового и перспективного преобразования вершин, генерации текстурных координат, расчета освещения и так далее.



# Два слова о шейдерах

## Геометрические шейдеры (Geometry Shader)

Геометрический шейдер, в отличие от вершинного, способен обработать не только одну вершину, но и целый примитив (до шести вершин для треугольного примитива). Кроме того, геометрический шейдер способен генерировать примитивы «на лету», не задействуя при этом центральный процессор.



# Два слова о шейдерах

## Пиксельные шейдеры (Pixel Shader)

Пиксельный шейдер работает с фрагментами растрового изображения, под которыми понимается пиксел, которому поставлен в соответствие некоторый набор атрибутов, таких как цвет, глубина, текстурные координаты. Пиксельный шейдер используется на последней стадии графического конвейера для формирования фрагмента изображения.



# А что на выходе? Интерфейс!

Графический интерфейс пользователя или графический пользовательский интерфейс (Graphical User Interface, GUI) — разновидность пользовательского интерфейса, в котором элементы интерфейса (меню, кнопки, значки, списки и т. п.), представленные пользователю на дисплее, исполнены в виде графических изображений.

В отличие от интерфейса командной строки, в GUI пользователь имеет произвольный доступ (с помощью устройств ввода) ко всем видимым экранным объектам (элементам интерфейса) и осуществляет непосредственное манипулирование ими. Чаще всего элементы интерфейса в GUI реализованы на основе метафор и отображают их назначение и свойства, что облегчает понимание и освоение программ неподготовленными пользователями.

# Разновидности интерфейсов

Программы с классической (полноэкранной) организацией экрана могут использовать элементы оконного интерфейса (субокна) для организации меню, окон сообщений и диалогов.

Программы с полной реализацией оконного интерфейса отдельно работают с отдельными подзадачами в разных окнах. Такая программа может одновременно открывать/работать с несколькими документами, помещая их в отдельные субокна (например, многооконный редактор с документом в каждом окне). Организацию этих субокон в подобных программах реализуют несколькими способами:

- однооконный режим (SDI - single document interface)
- многооконный режим (MDI, TDI - multiple/tabbed document interface)
- псевдомногооконный режим (PMDI - pseudoMDI)

# Принципы организации интерфейсов пользователя

Одним из требований к хорошему графическому интерфейсу программной системы является концепция «делай то, что я имею в виду» или DWIM (Do What I Mean). DWIM требует, чтобы система работала предсказуемо, чтобы пользователь заранее интуитивно понимал, какое действие выполнит программа после получения его команды.

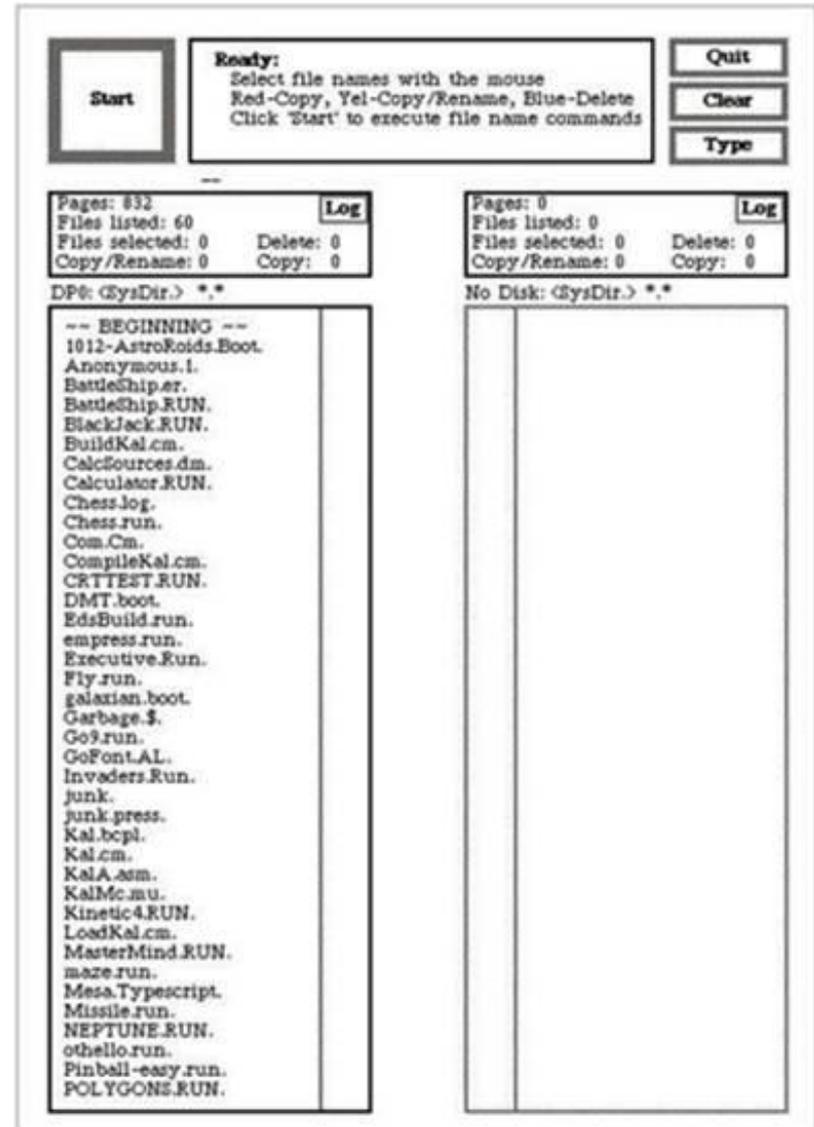
WYSIWYG (What You See Is What You Get, «что видишь, то и получишь») — свойство прикладных программ или веб-интерфейсов, в которых содержание отображается в процессе редактирования и выглядит максимально близко похожим на конечную продукцию, которая может быть печатным документом, веб-страницей или презентацией.



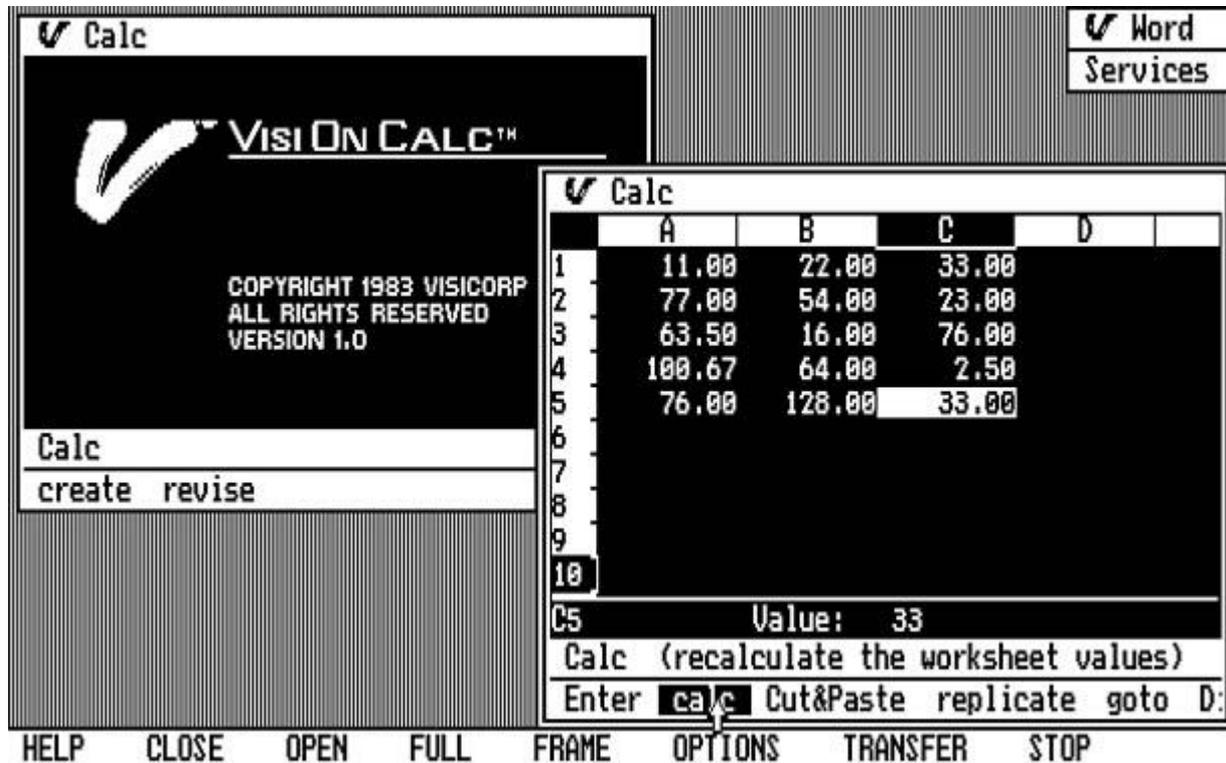
# История развития интерфейсов



1973 | Xerox Alto

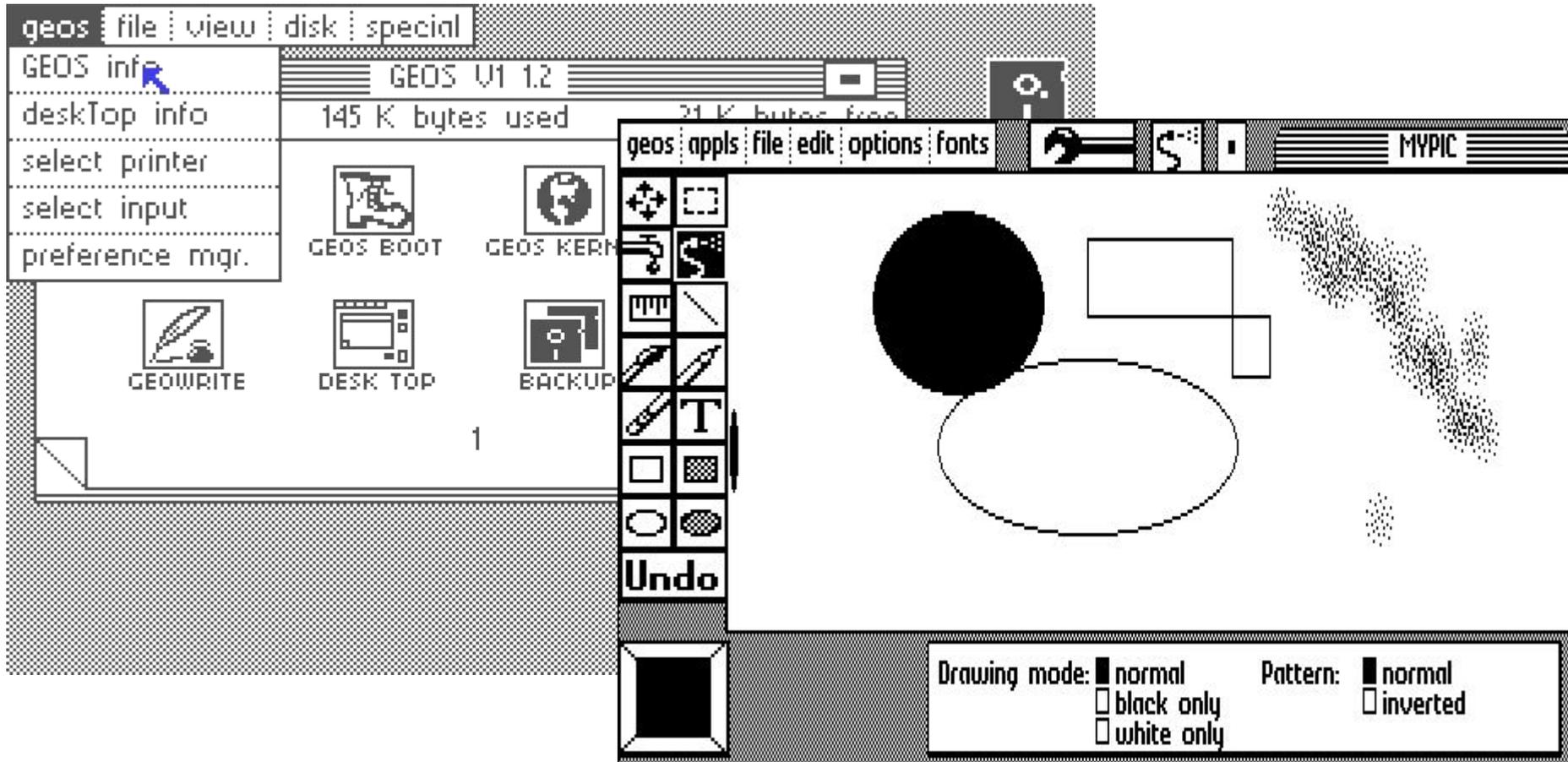


# История развития интерфейсов



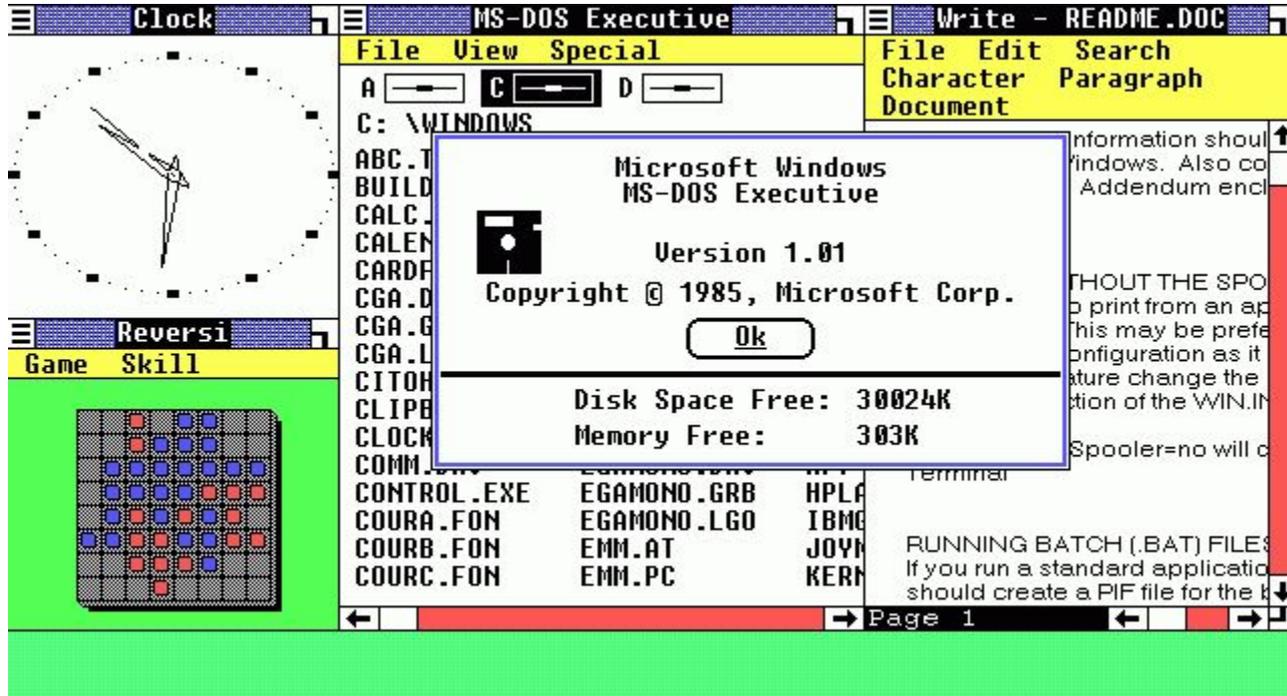
1983 | IBM Visi On

# История развития интерфейсов



1984 | Apple II GEOS

# История развития интерфейсов



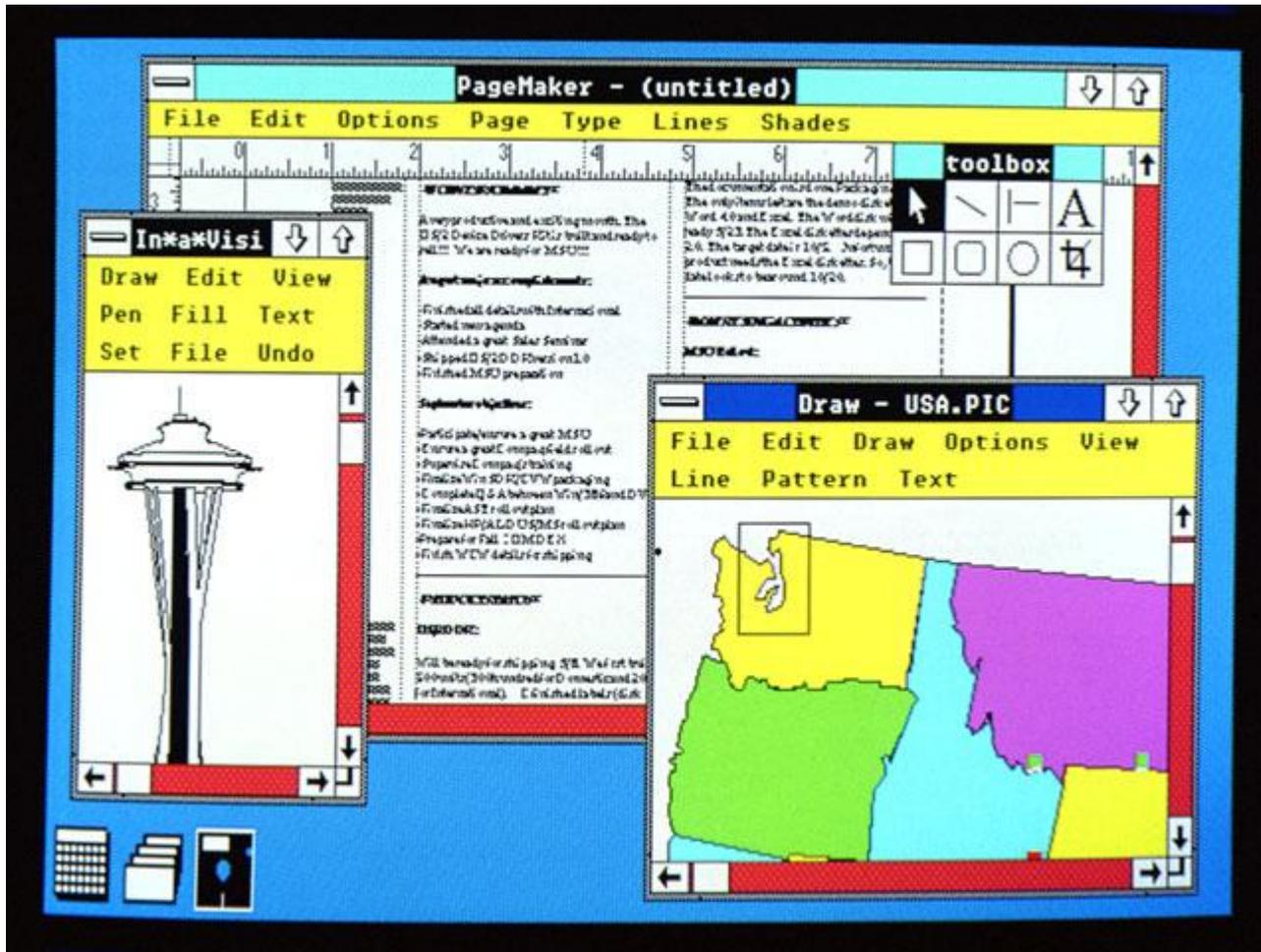
1985 | Microsoft Windows 1.0

# История развития интерфейсов



1985 | Commodore Amiga Workbench

# История развития интерфейсов



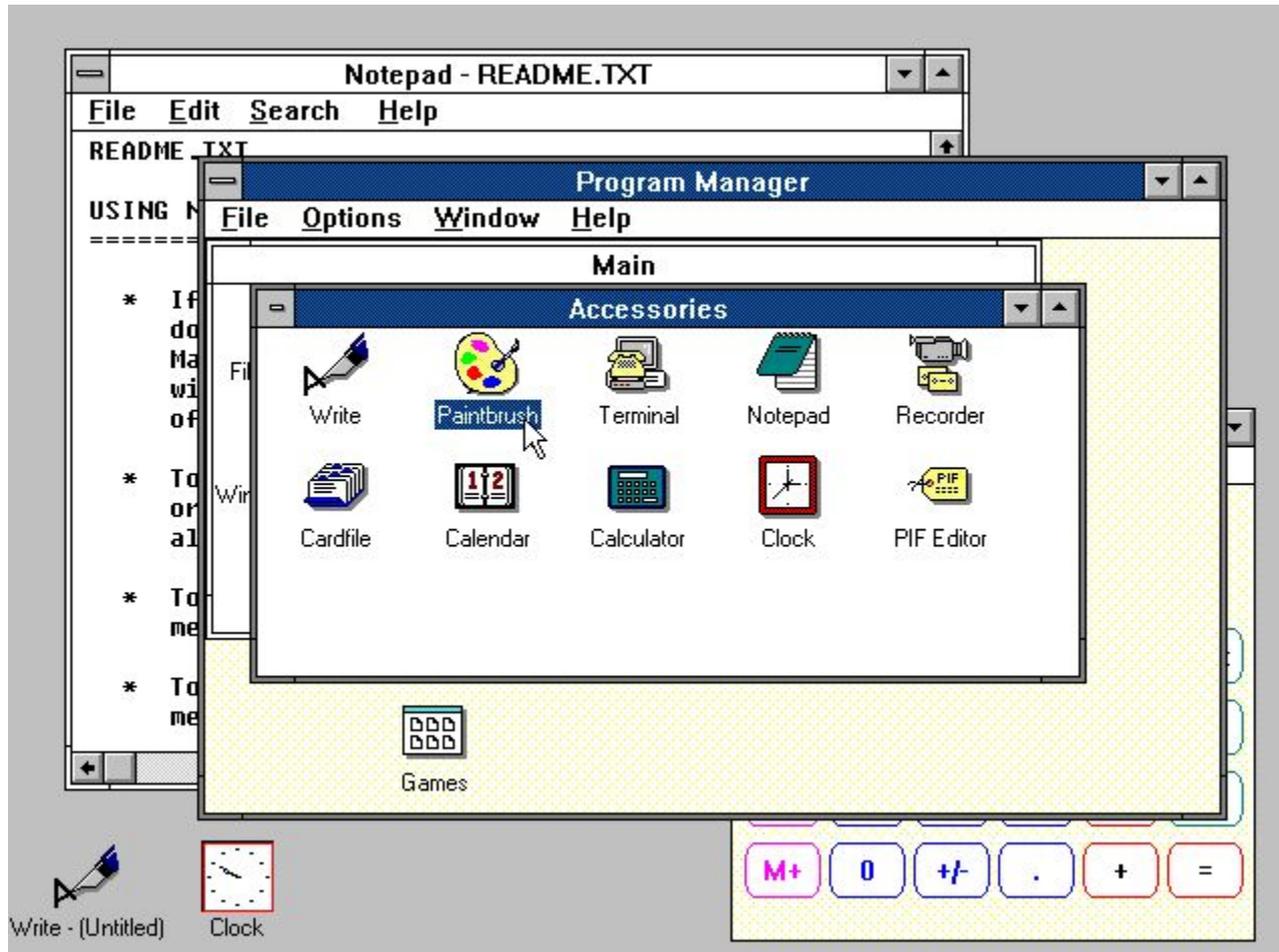
1987 | Microsoft Windows 2.0

# История развития интерфейсов



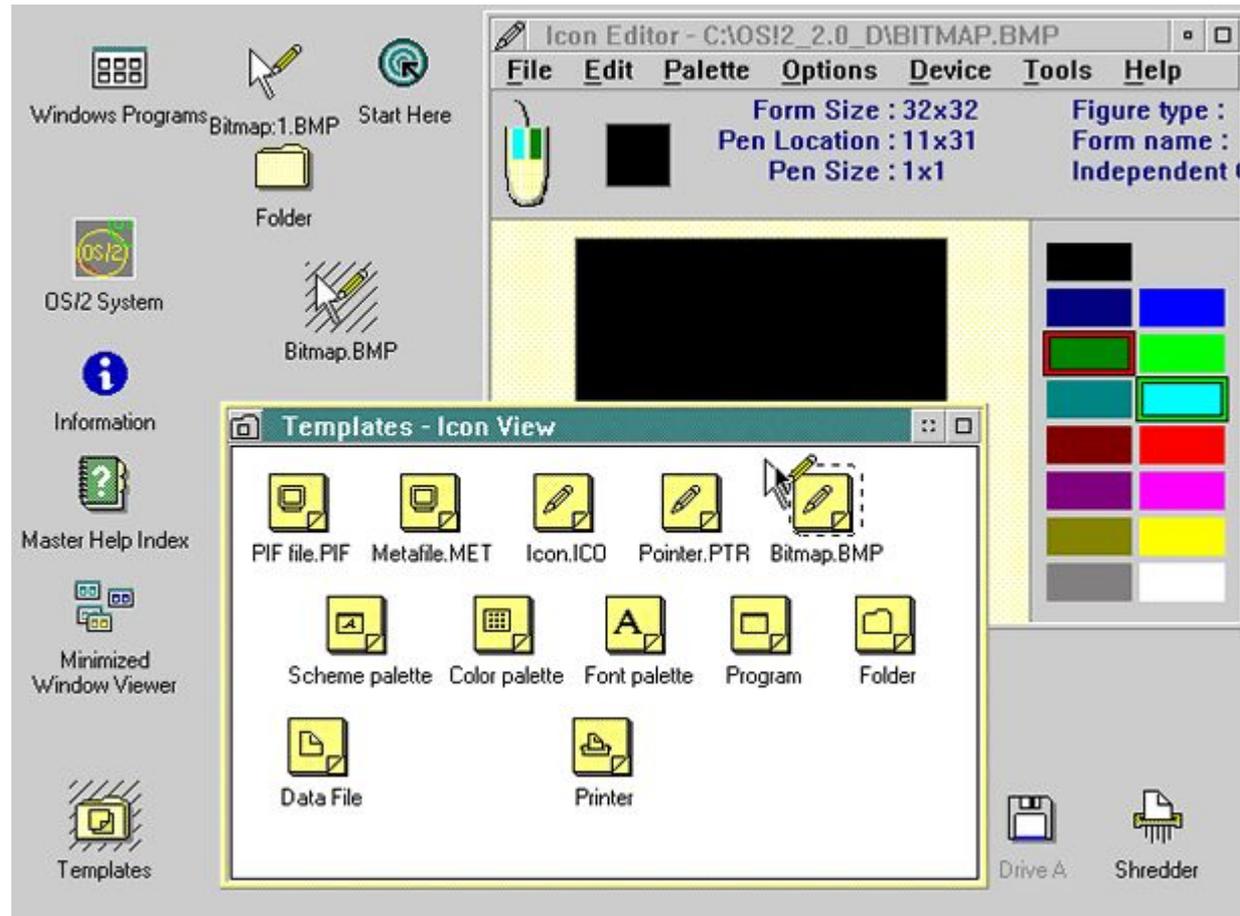
1990 | Amiga Workbench 2

# История развития интерфейсов



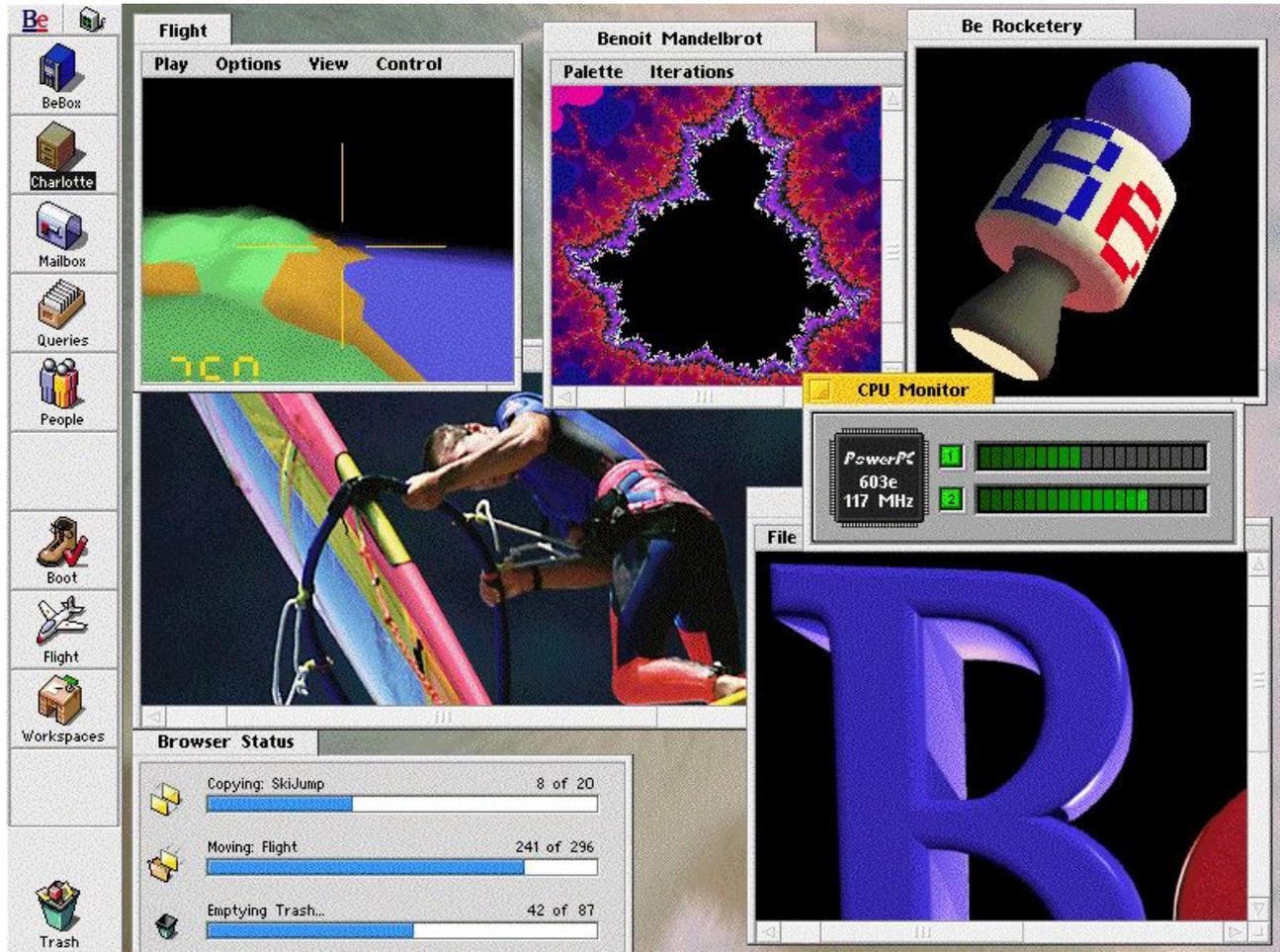
**1990 | Microsoft Windows 3.0**

# История развития интерфейсов



1992 | IBM OS/2 2.0

# История развития интерфейсов



1995 | BeOS