



# Перспективы развития индустрии ПО

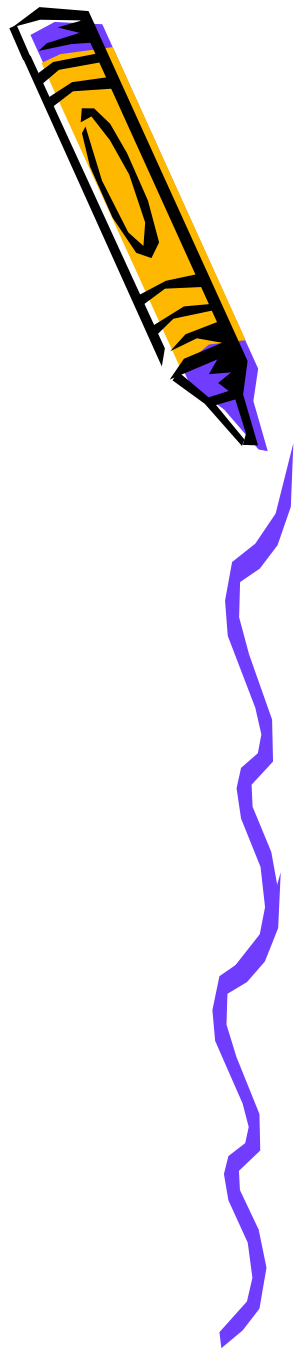
Дмитрий Маленко (dmal@isd.dp.ua)

MCAD, MCSD.NET



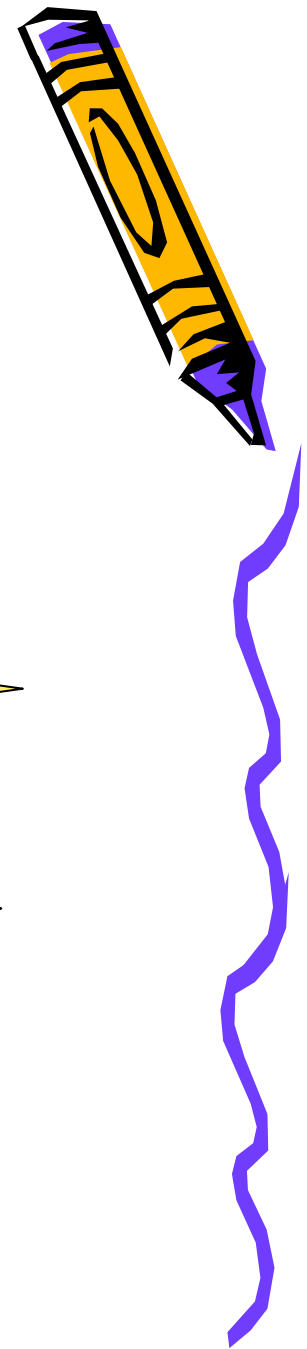
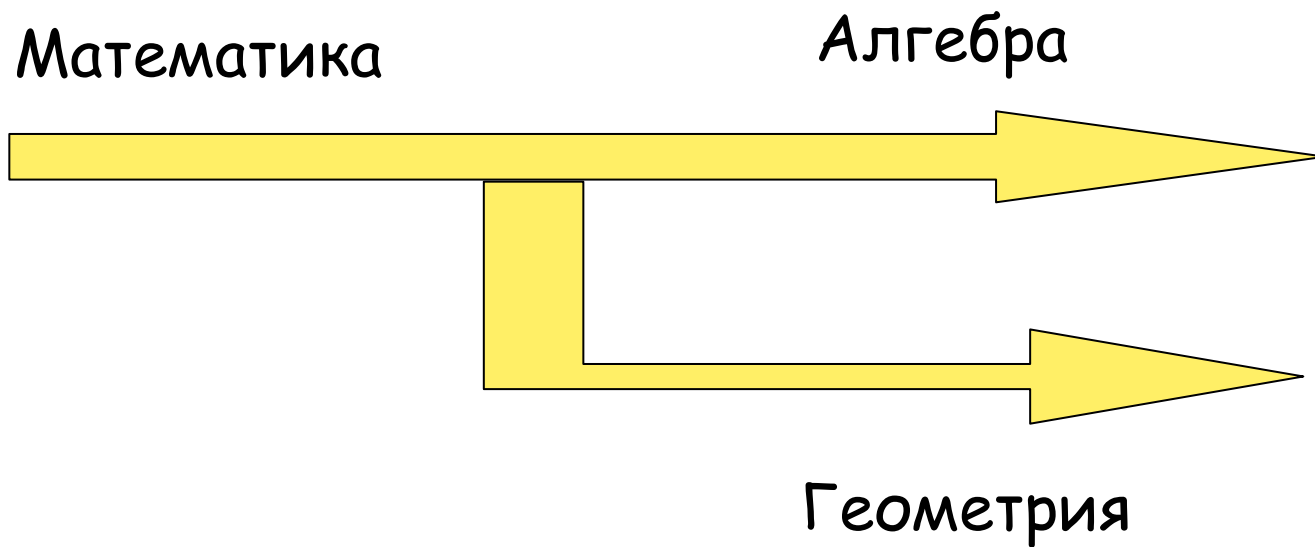
# О чем мы будем ГОВОРИТЬ...

- Индустрия ПО
  - Прошлое
  - Настоящее
  - Будущее



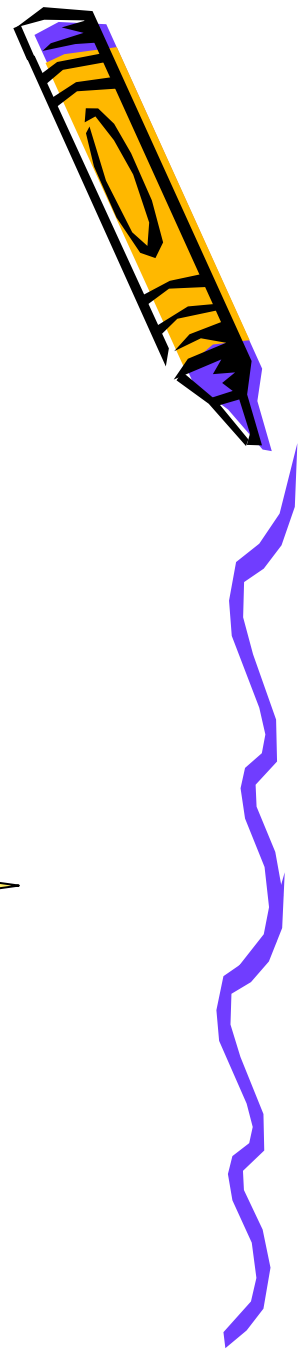
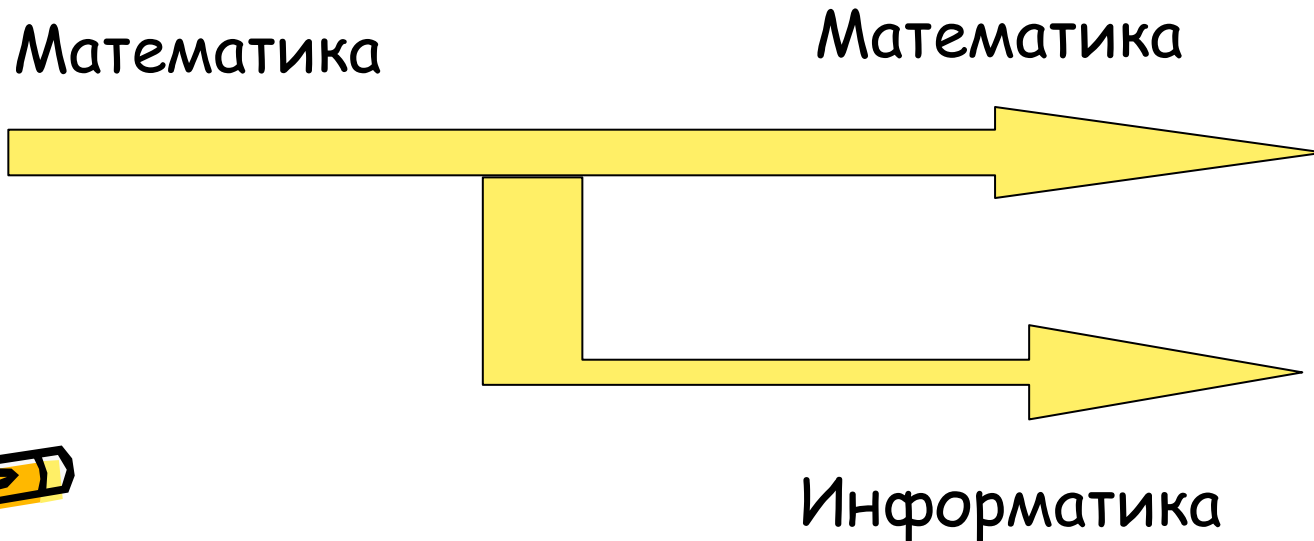
# Прошлое информатики

- Древняя Греция

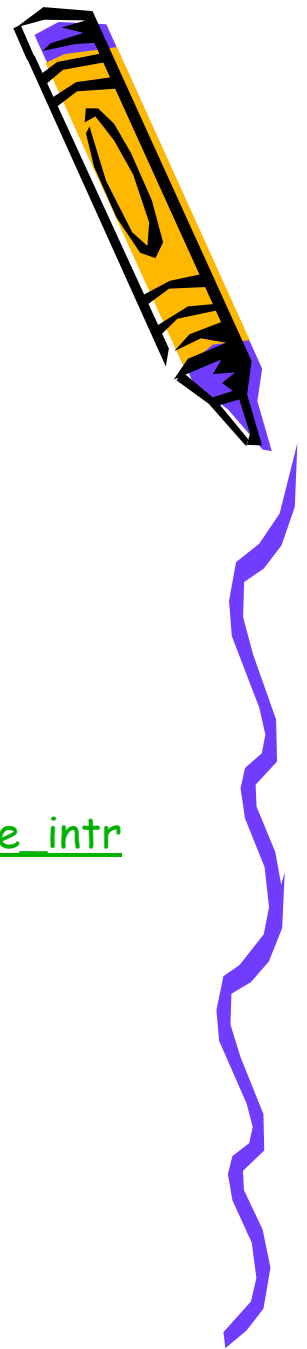


# Прошлое информатики

- XIX-XX век
  - Информатика выделяется в самостоятельную дисциплину



# Компьютеры начинают приносить деньги



- Середина XX века - начинается коммерческое использование компьютеров

- 1952 г. - IBM 701

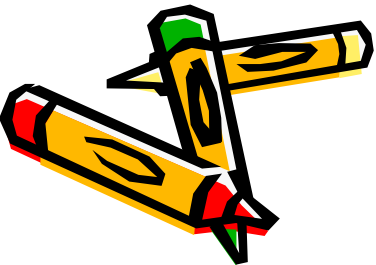
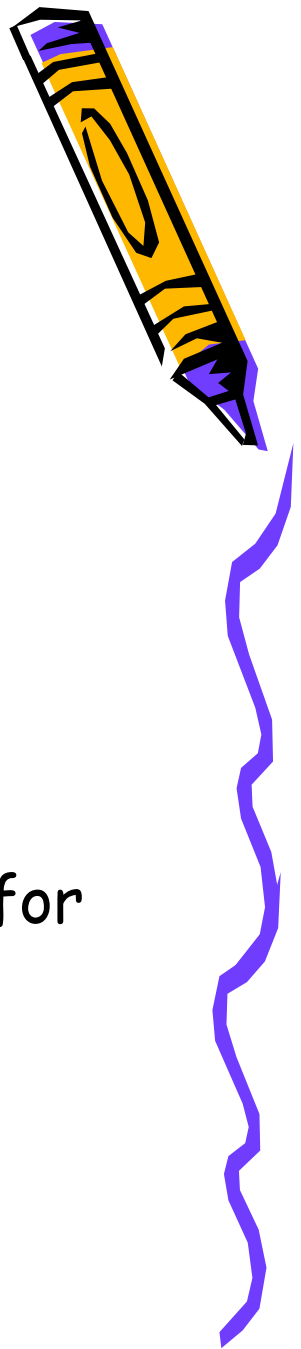
[http://www.ibm.com/ibm/history/exhibits/mainframe/mainframe\\_intro.html](http://www.ibm.com/ibm/history/exhibits/mainframe/mainframe_intro.html)

- Появляется потребность в специалистах

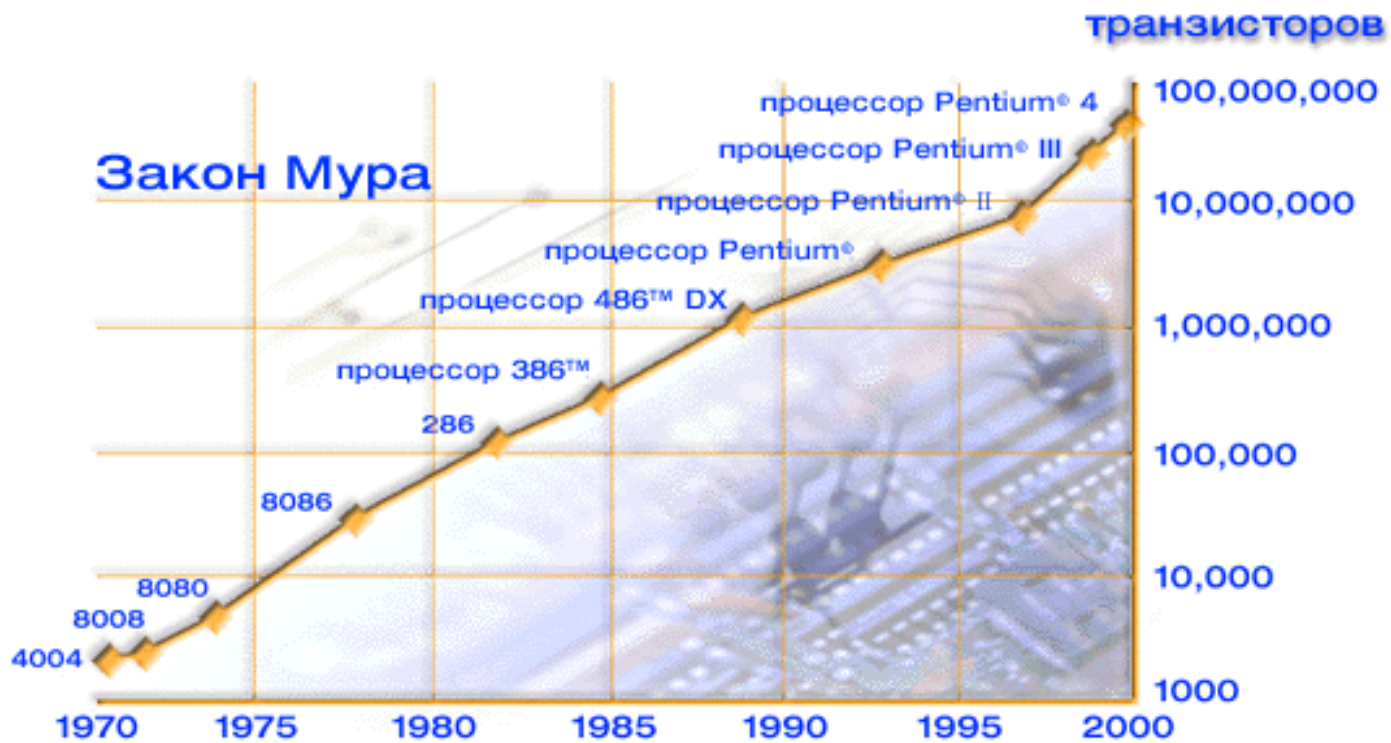


# Университеты

- 1960-е гг. - появляются первые факультеты по информатике и проектированию компьютеров
- Первый образовательный стандарт
  - ACM Curriculum Committee on Computer Science. Curriculum '68: Recommendations for the undergraduate program in computer science



# Закон Мура



<http://www.intel.com/ru/Intel/museum/history/hof/moore.htm>

# Развитие технологий

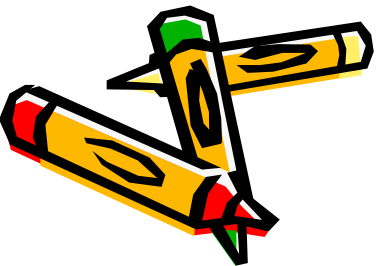
- Появление новых технологий и совершенствование старых приводят к изменению свода знаний, необходимых профессионалу
- Выходят последующие версии образовательных стандартов в 1978, 1983, 1991, 2001



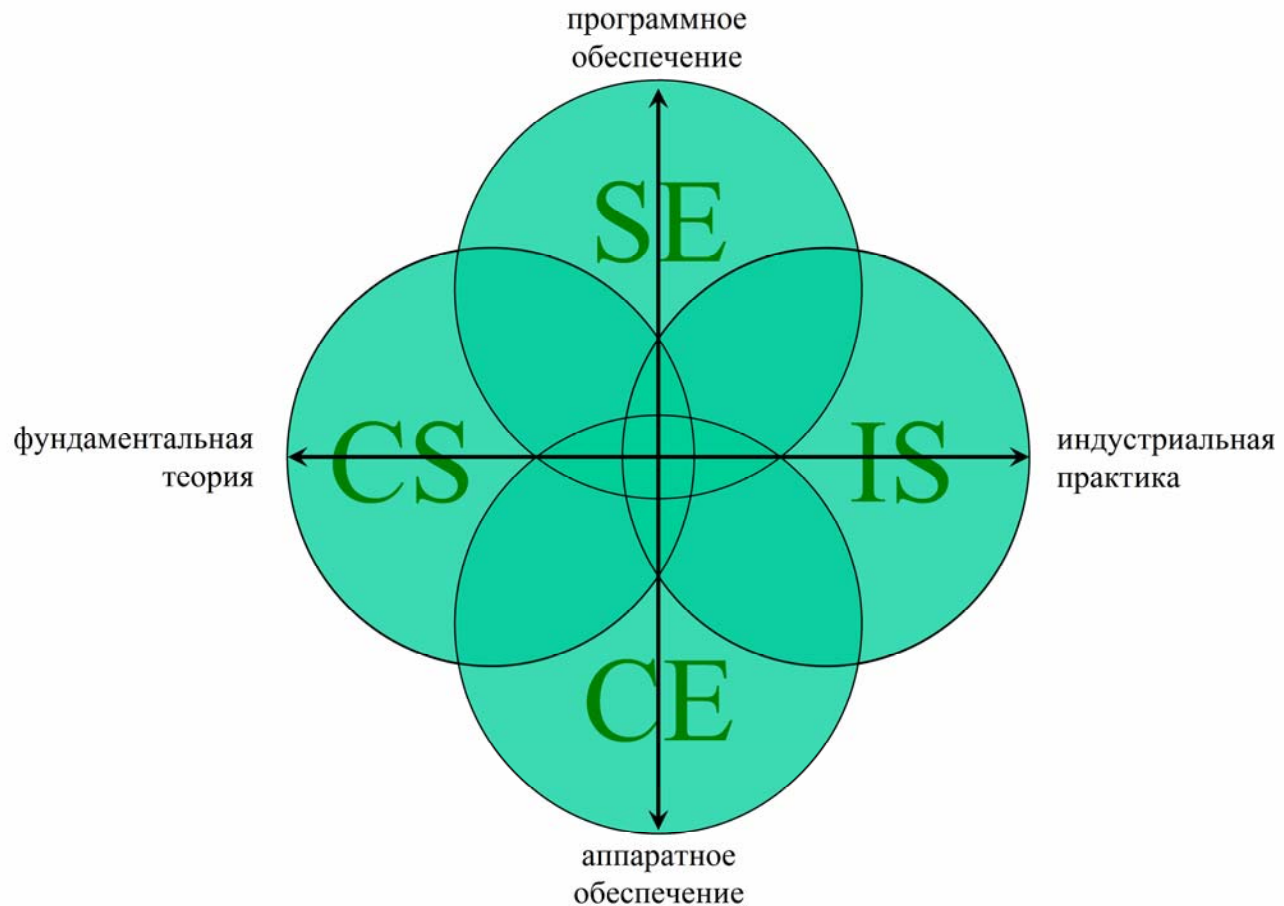


# 90-е гг. XX века

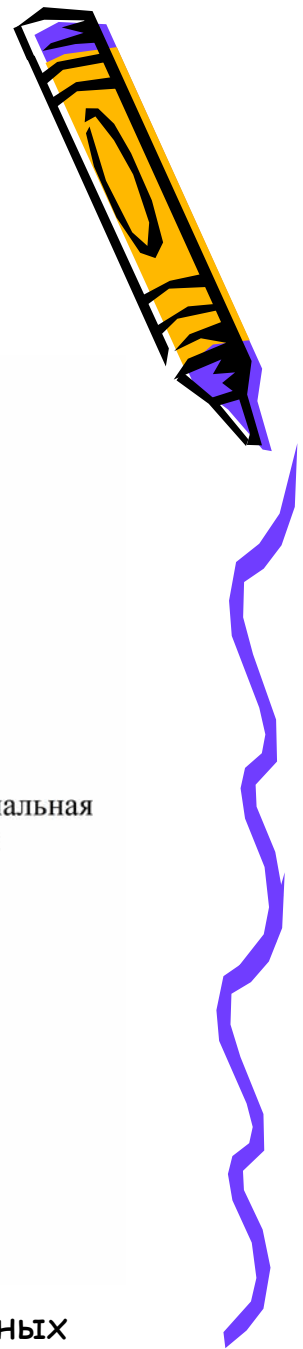
- Стремительное развитие информационных технологий приводит к выделению специализаций внутри самой информатики
  - Компьютерная наука
  - Программная инженерия
  - Проектирование компьютеров
  - Информационные системы



# Соотношение между дисциплинами

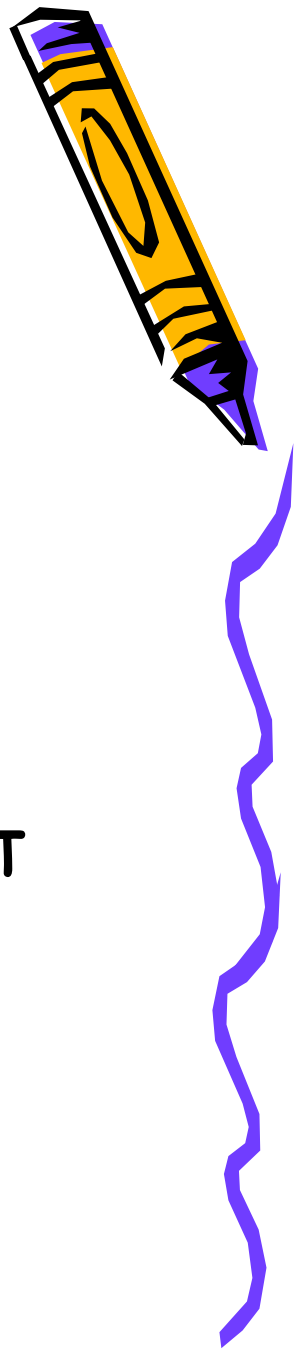


В.Л.Павлов, А.А.Терехов. О выборе карьеры в области информационных технологий.

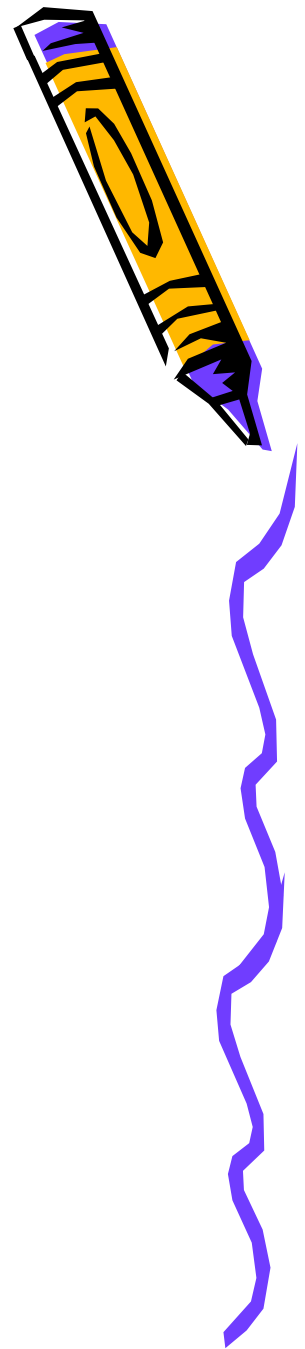


# Software Engineering

- 1952 г. - John Tukey - "software"
- 1968 г. - Заголовок конференции NATO включал слова "software engineering"
- 1972 г. - IEEE впервые публикует "Transactions on Software Engineering"



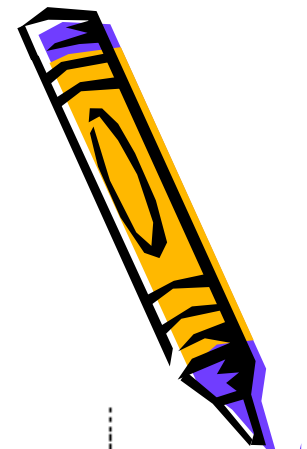
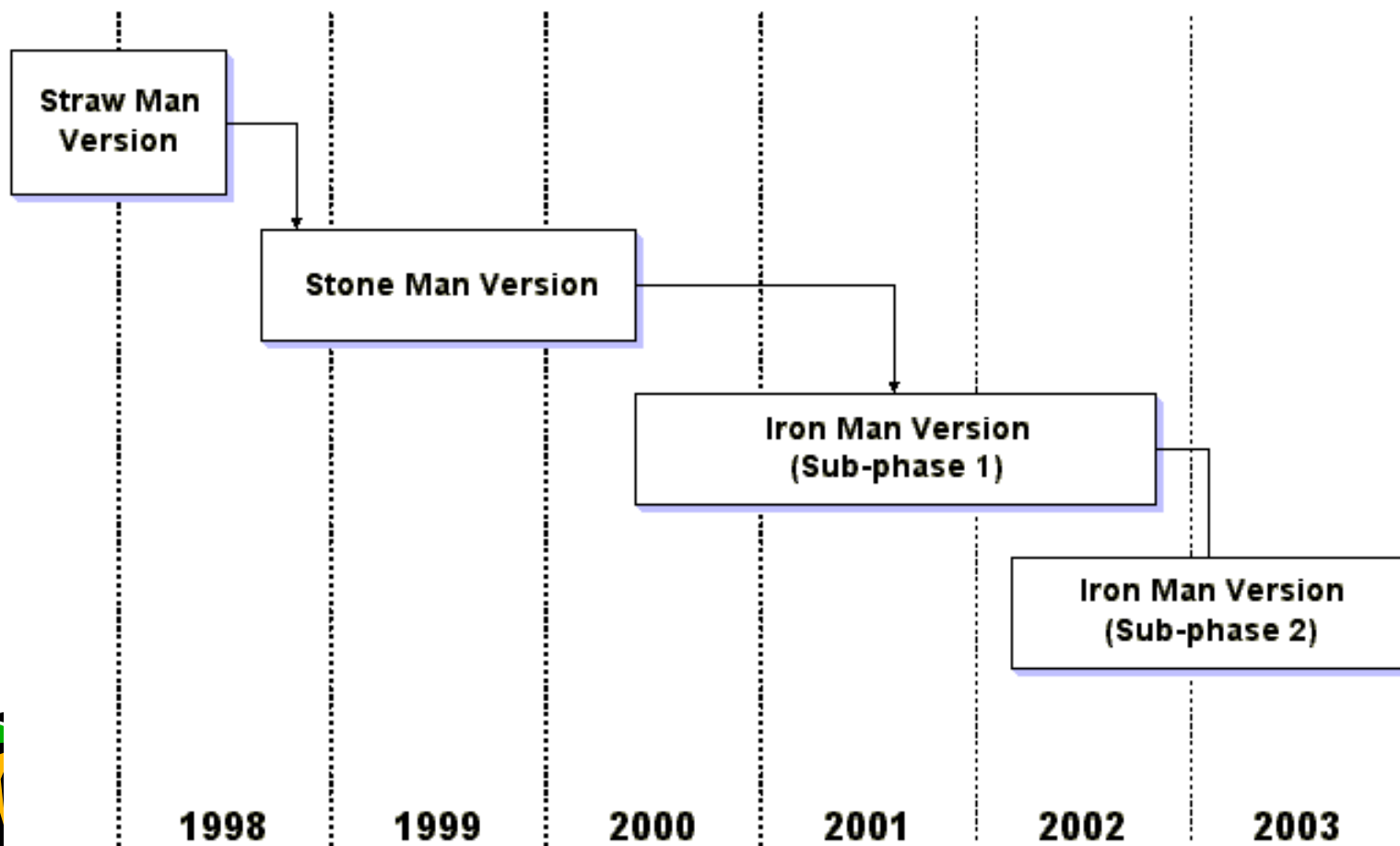
# Software Engineering Body of Knowledge



- Описывает рамки дисциплины «Программная инженерия»
- Категоризирует свод знаний, входящих в данную дисциплину
- <http://www.swebok.org>
- <http://computer.org>

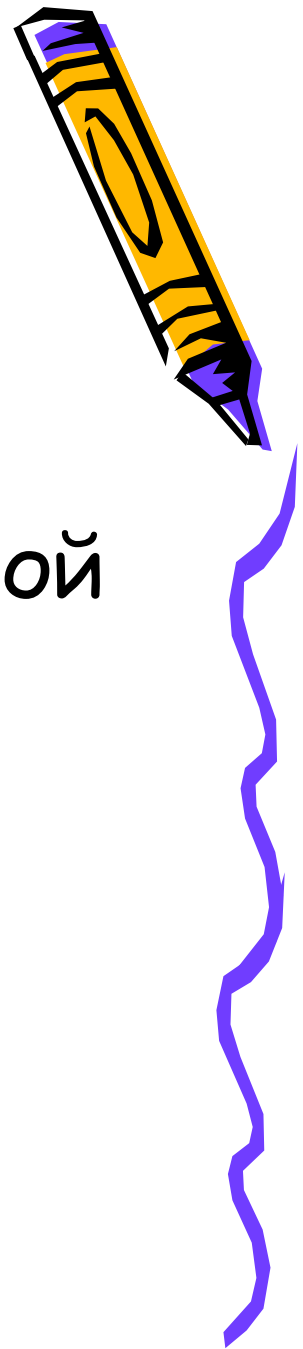


# Эволюция SWEВОК



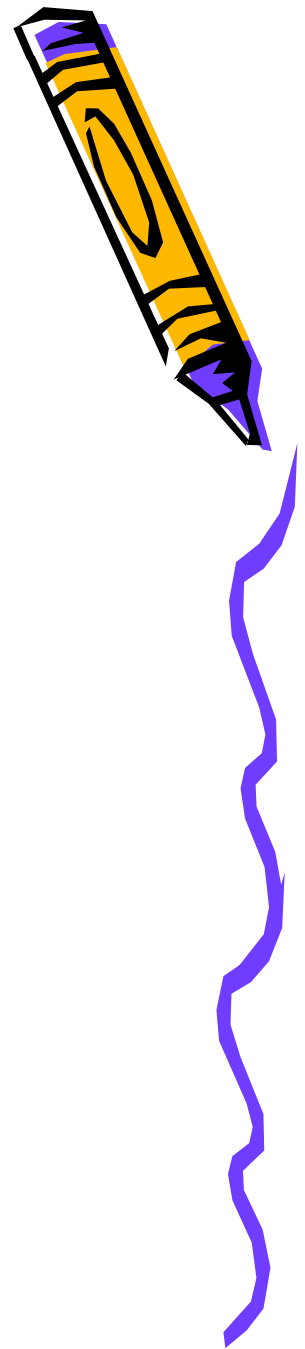
# Области знаний

- 10 областей знаний
- Первые 5 соответствуют каскадной модели жизненного цикла ПО
- Остальные 5 идут по алфавиту



# Требования к ПО (Software Requirements)

- Определение требований
- Анализ требований
- Спецификация требований к ПО
- Верификация требований
- Управление требованиями



# Проектирование ПО (Software Design)

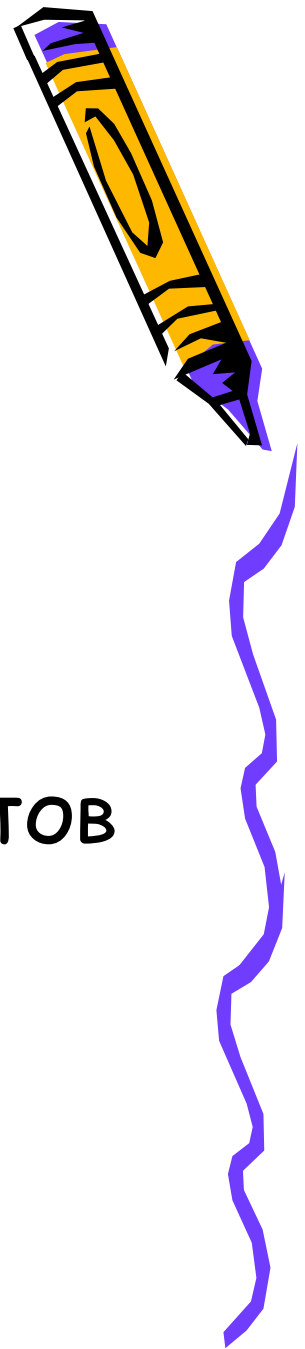


- Основные концепции проектирования ПО
- Ключевые аспекты проектирования ПО
- Структура и архитектура
- Оценка и качество проектирования
- Нотации проектирования
- Стратегии и методы разработки ПО

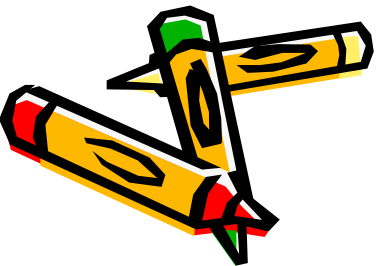




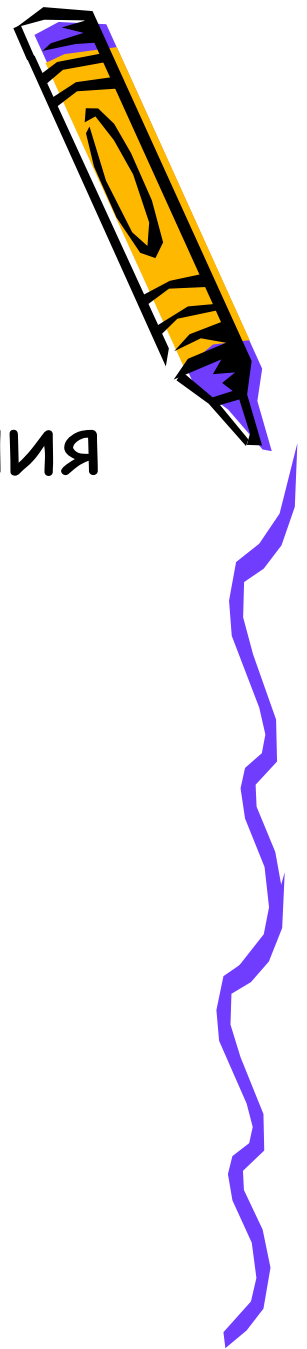
# Конструирование ПО (Software Construction)



- Декомпозиция задач
- Уменьшение сложности ПО
- Структуризация для проверки
- Использование внешних стандартов



# Тестирование ПО (Software Testing)



- Основные концепции тестирования
- Виды и уровни тестирования
- Способы тестирования
- Метрики тестирования
- Управление тестированием



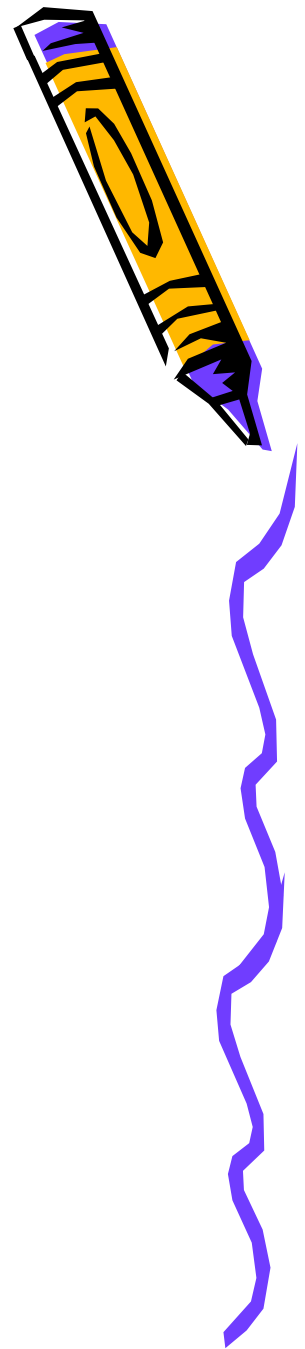
# Сопровождение ПО (Software Maintenance)



- Основные концепции сопровождения ПО
- Процесс сопровождения ПО
- Ключевые аспекты сопровождения
- Технологии сопровождения



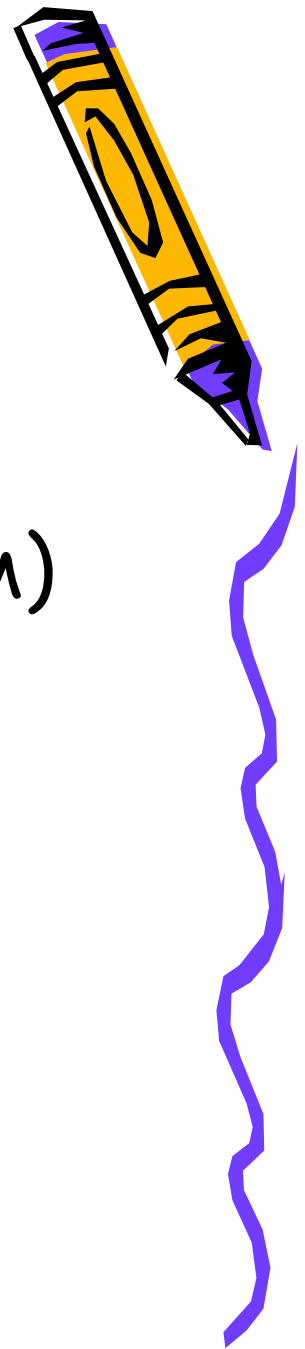
# Управление конфигурацией ПО (Software Configuration Management)



- Управление SCM
- Определение конфигурации ПО
- Контроль конфигурации
- Учет состояния конфигурации
- Аудит конфигурации ПО
- Управление выпуском ПО



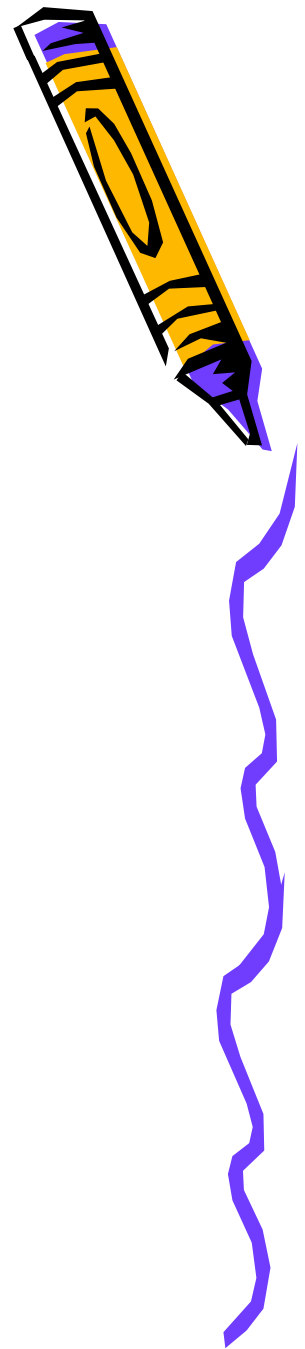
# Управление разработкой ПО (Software Engineering Management)



- Организационное управление
- Управление проектом(процессом)
- Метрики ПО



# Процесс разработки ПО (Software Engineering Process)

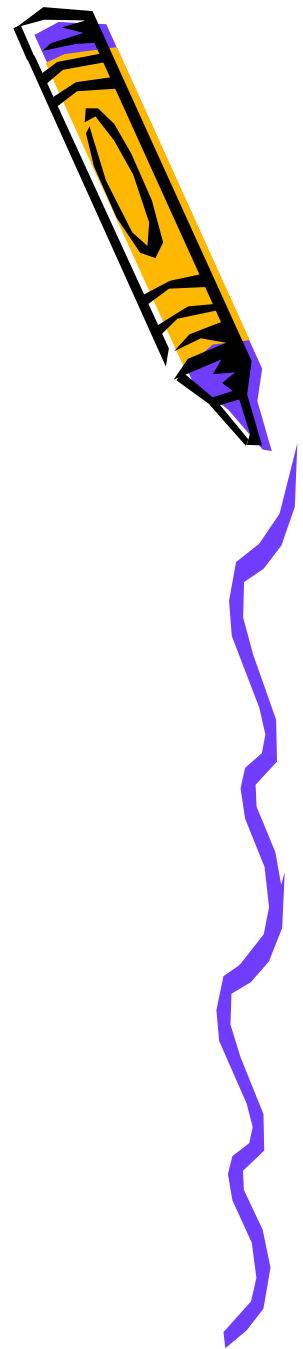


- Основные концепции
- Инфраструктура процесса
- Метрики специфичные для процесса разработки ПО
- Определение процесса
- Качественный анализ процесса
- Реализация и изменение процесса разработки



# Методы и средства разработки ПО (Software Engineering Tools and Methods)

- Среды разработки ПО
- Методы разработки ПО
  - Эвристические
  - Формальные
  - Основанные на прототипировании
  - Другие



# Качество ПО (Software Quality)



- Концепция качества ПО
- Цели и планирование обеспечения качества (QA) ПО
- Методы и техники для обеспечения качества
- Метрики, использующиеся при обеспечении качества





# Перспективы

*Я не знаю другого способа судить о  
будущем, как с помощью прошлого*

*ПАТРИК ГЕНРИ*

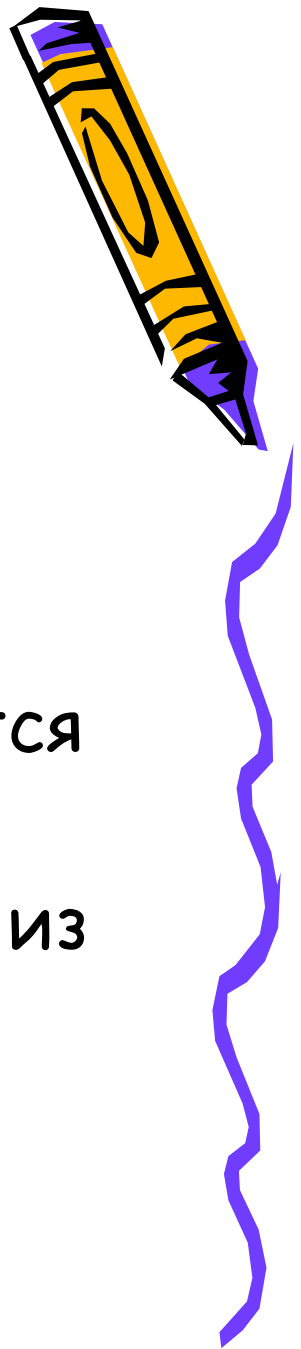
*Опираясь на прошлое, невозможно  
планировать будущее*

*ЭДМУНД БЕРК*

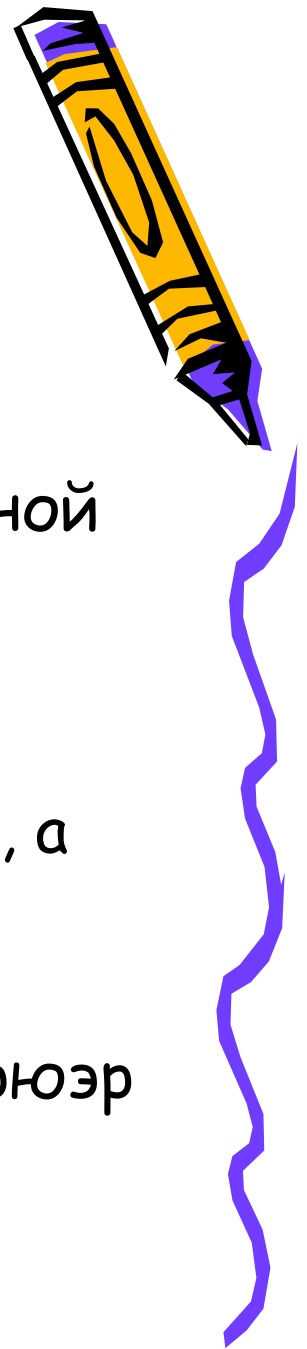


# Настоящее

- Программная инженерия стала более зрелой, чем была в момент своего возникновения в 60-х
- До сих пор выпускается и используется изобилующее ошибками ПО
- Пользователи не считают это чем-то из ряда вон выходящим!!!



# Процессы разработки



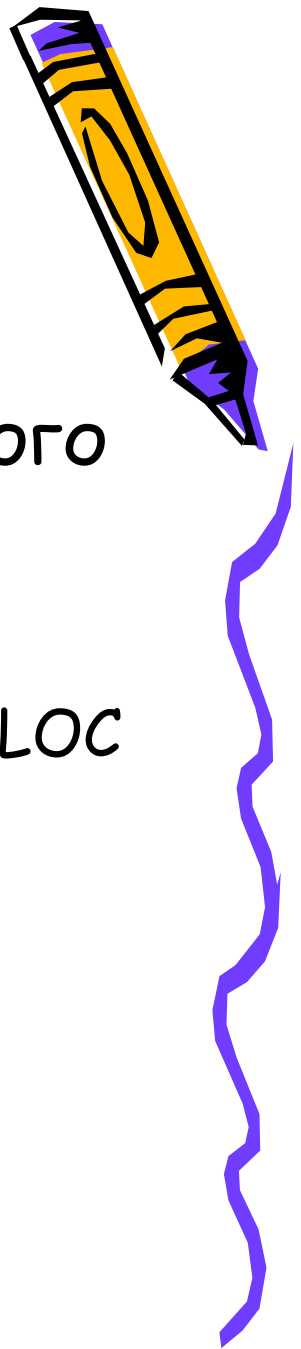
- Более половины опрошенных на Международной конференции по программной инженерии (2000г.) затрагивали проблемы самого процесса работы
- Нам необходимы упрощенные процессы, в которые пользователи действительно верят, а не инструменты и процессы, которые их вынуждают использовать

Эрик Брюэр



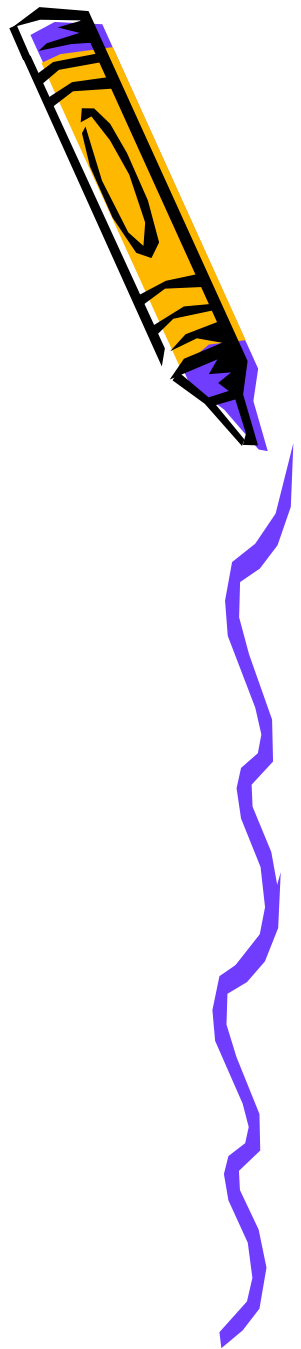
# Сложность ПО

- 80-е - ~10KLOC предел для структурного подхода
- Каков предел для ООП?
  - Объем исходного кода Windows XP ~50MLOC
- Аспектно-ориентированное программирование
  - <http://www.aosd.net>



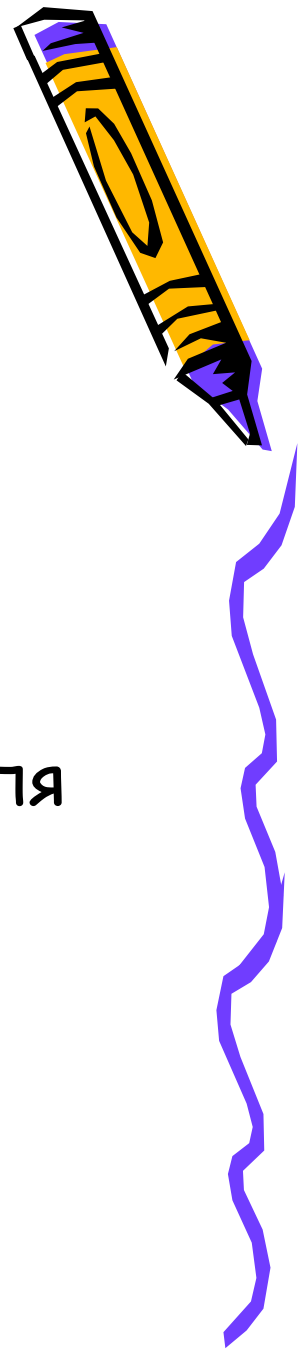
Что дальше?

Конвергенция...



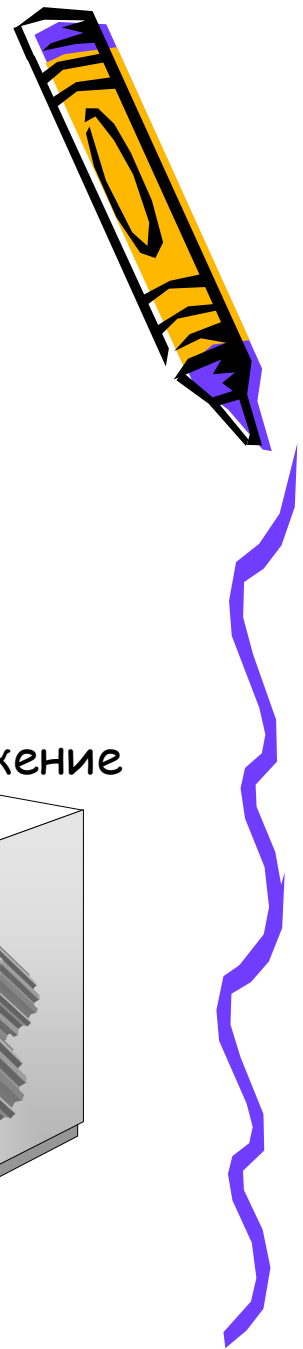
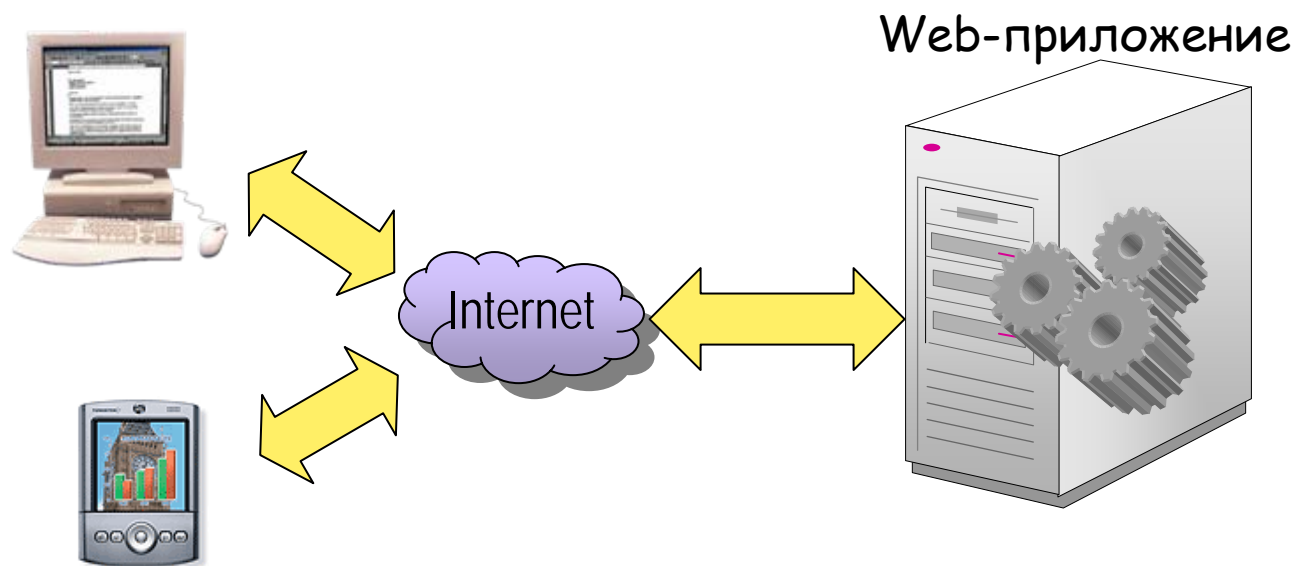
# ...различных платформ

- Взрывное развитие Интернета
  - 2000г. - 400 млн. пользователей
  - 2003г. - 800 млн. пользователей
- Java предоставила возможность объединить различные платформы для разработчика
- Открытые стандарты
  - XML, SOAP...



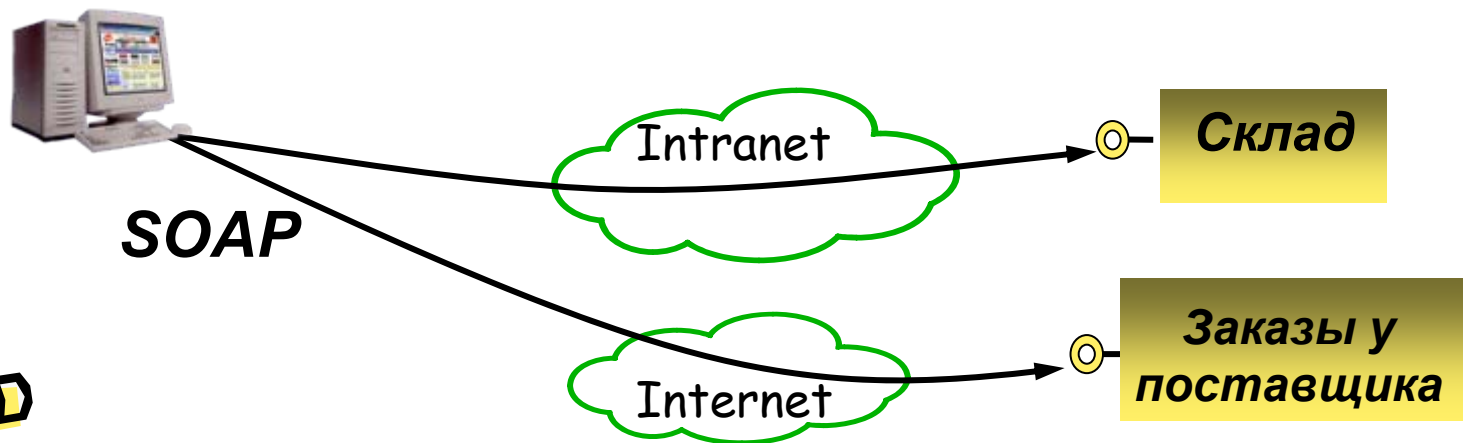
# ...различных устройств

- Почти каждое электронное устройство управляется программами



# ...различных систем

- С помощью ПО предприятия хотят автоматизировать не только внутреннюю деятельность, но и взаимодействия с другими предприятиями
- XML Web Services





# ...средств разработки

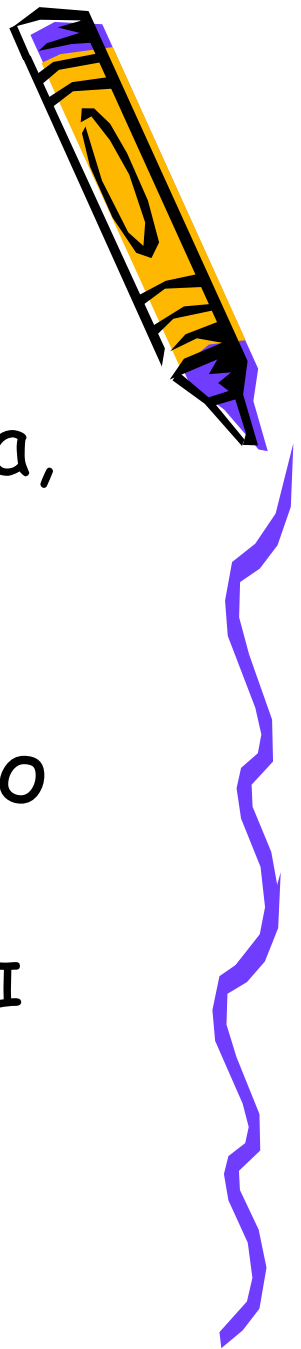


- .NET Framework
  - Поддерживается более 20 различных языков, которые могут абсолютно безболезненно взаимодействовать
- Jim Miller: Мне хотелось достичь две очень простые цели, о которых я мечтал более тридцати лет:
  - *Писать программы на языках, которые я люблю, но использовать при этом библиотеки, созданные другими (менее просвещенными) людьми на других языках*
  - *Писать библиотеки на языках, которые я люблю, но делать их доступными для других (менее просвещенных) людей, создающих программы на других языках*



## ...средств разработки

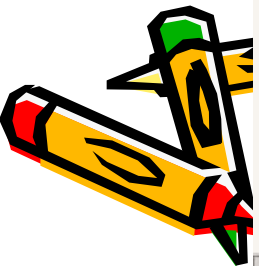
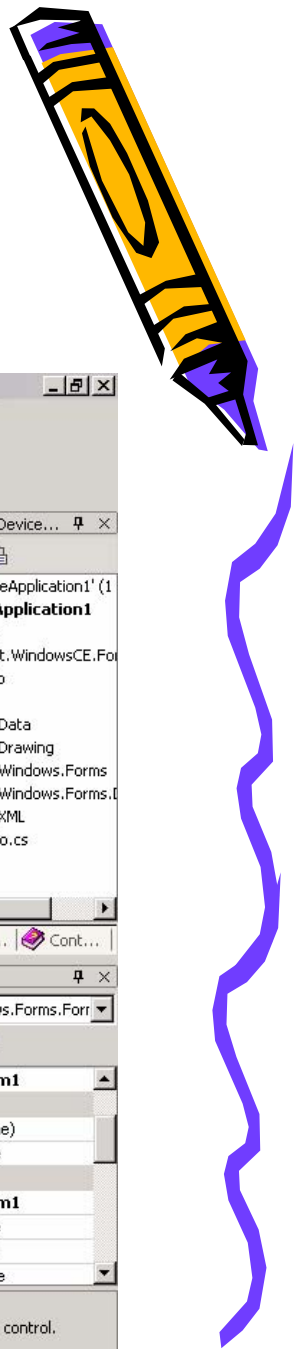
- Вы наверняка помните те времена, когда среда интегрированным отладчиком была роскошью
- Сейчас в Visual Studio .NET можно не только создавать, но и отлаживать хранимые процедуры



# ...средств разработки

The screenshot displays the Microsoft Visual Studio .NET development environment for a SmartDeviceApplication1 project. The main workspace is divided into several panes:

- Design View:** Shows the visual representation of the application form. It features a tree view with two main categories: "Sports" (containing Soccer, Basketball, and Baseball) and "Hobbies" (containing Hiking). Below the tree are two buttons labeled "Add category" and "Remove category".
- Emulator:** A central window titled "Pocket PC 2002" showing the application running on a virtual mobile device. The device screen displays the same form as the design view.
- Solution Explorer:** Located on the right, it shows the project structure for "SmartDeviceApplication1", including references to various system assemblies like Microsoft.Windows.CE.Forms, MSCorLib, System, System.Data, System.Drawing, System.Windows.Forms, System.XML, AssemblyInfo.cs, and Form1.cs.
- Properties Window:** Also on the right, it shows the properties for the selected "Form1" control, such as Name, DrawGrid, GridSize (8; 8), and Locked.
- Output Window:** At the bottom, it displays the build output, including the message "Launching Application" and an error: "Error while trying to run project: Unable to start debugging. Unable to start program 'C:\Documents and Settings\dima\Мои документы\Visual Studio Projects\SmartDeviceAppli...".



# ...моделирования и разработки



- Rational XDE - моделирование и разработка в одном флаконе

The screenshot displays the Rational XDE environment. The central workspace shows a UML class diagram with the following elements:

- SearchResultsBean** (Class):
  - Operations: `~ SearchResultsBean ()`, `+ setResult ()`, `+ getLength ()`, `+ getResults ()`, `+ getResult ()`
  - Attributes: `- results`
- SearchEngine** (Class):
  - Operations: `~ SearchEngine ()`, `+ find ()`, `+ doQuery ()`, `+ setupSearch ()`, `+ addEntries ()`
  - Attributes: `- resultsCount : int = 1`
  - Associations: `- resultsMap` (to `Map`), `- theSearchEngine` (to `SearchController`)
- SearchController** (Class):
  - Operations: `+ SearchController ()`, `+ doSearch ()`, `# doGet ()`, `# doPost ()`
  - Attributes: `- resultsPage : String = "ResultsView.jsp"`
- Map** (Interface):
  - Operations: `+ size ()`, `+ isEmpty ()`, `+ containsKey ()`, `+ containsValue ()`, `+ get ()`, `+ put ()`, `+ remove ()`, `+ removeAll ()`, `+ clear ()`, `+ keySet ()`, `+ values ()`, `+ entrySet ()`, `+ equals ()`, `+ hashCode ()`

The right-hand side of the interface features a **Model Explorer** showing a project structure:

- (TestDrive) Java Code Model\*
- Main (jdk\_min)
- (InstantUML) Java Code Model\*
- com
- mybiz
- searchapp
- SearchController
- SearchEngine
- SearchResultsBean
- SearchController.java
- SearchEngine.java
- SearchResultsBean.java
- Main (jdk\_min)
- jdk\_min

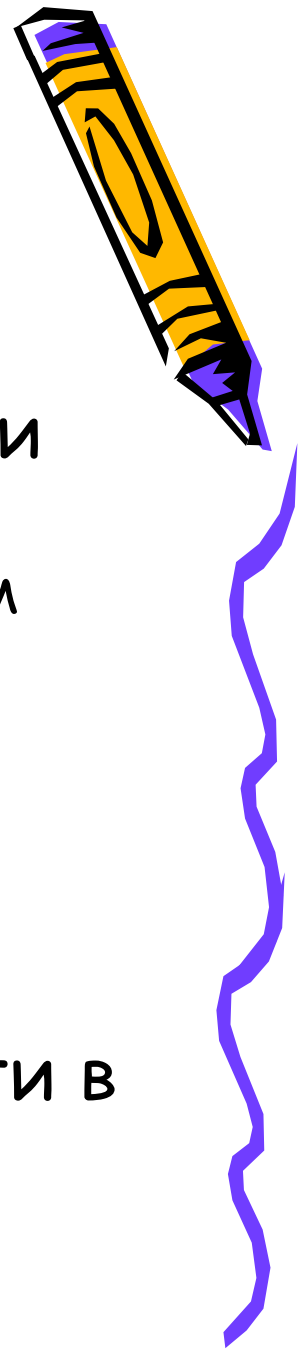
The bottom of the interface includes an **Output** window with the following log entries:

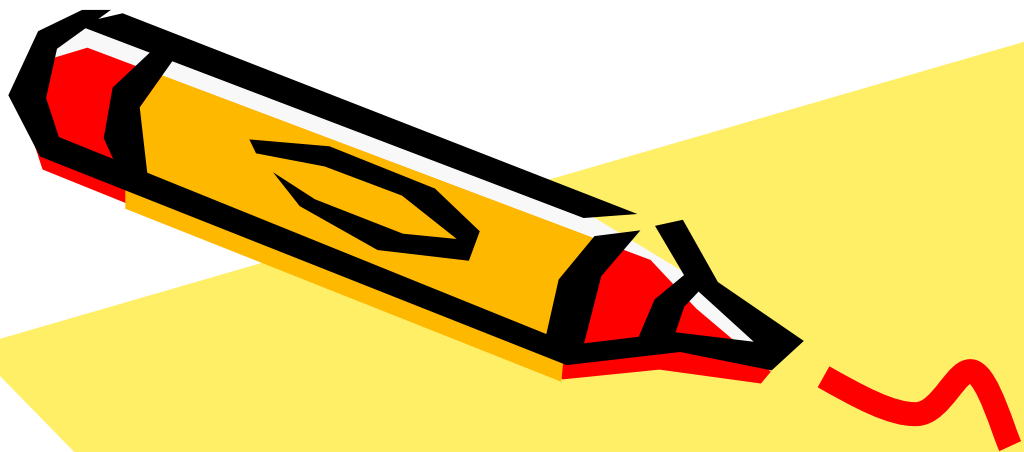
```
Rational XDE | Output Log
[Java : 2002-06-28 15:22:50] 'com::mybiz::searchapp::SearchResultsBean' - Added
[Java : 2002-06-28 15:22:50] Component item changes:
[Java : 2002-06-28 15:22:50] 'com::mybiz::searchapp::SearchResultsBean.java' - Added
[Java : 2002-06-28 15:43:27] Reverse engineering (Model: 'Java Code Model'):
```



## ...ВСЕГО СО ВСЕМ

- Баз данных с платформами разработки
  - Для SQL Server "Yukon" хранимые процедуры можно будет писать на любом языке .NET
- Файловых систем с базами данных
  - В Windows "Longhorn" файловая система будет построена на основе реляционной СУБД
- Нам не придется скучать читая новости в компьютерных журналах...





Вопросы?

