

Дисциплина:
Технология разработки программного обеспечения

Технология разработки программного обеспечения (ТРПО) -

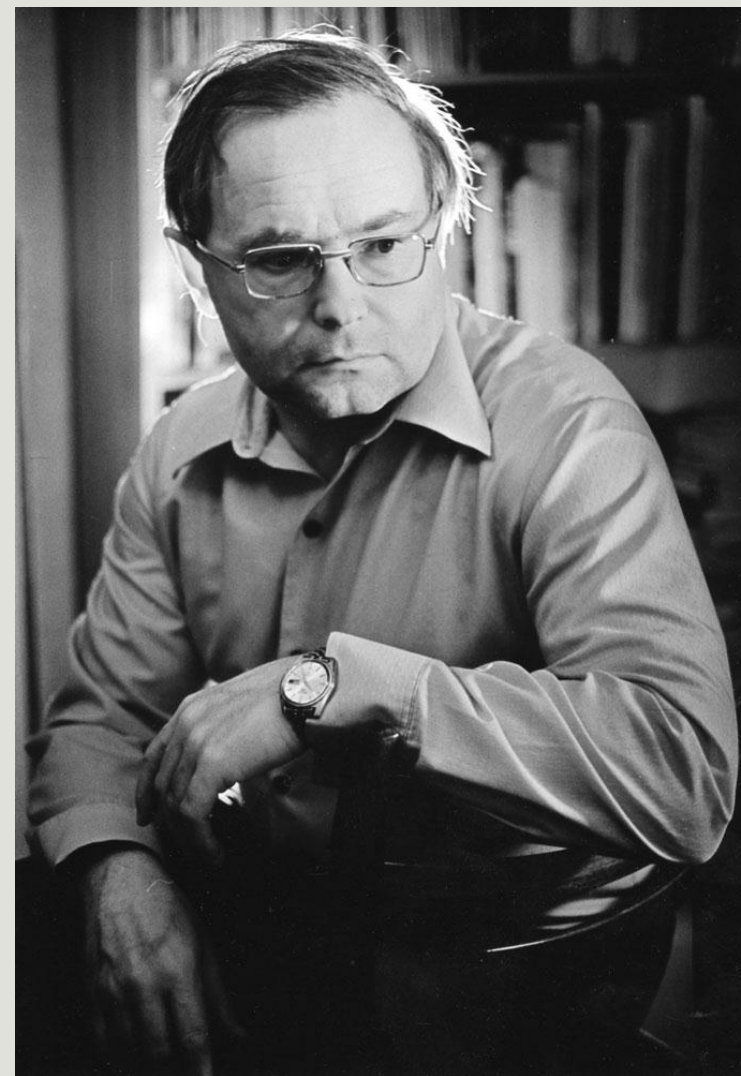
- это совокупность процессов и методов создания программного продукта.
- это система инженерных принципов для создания экономического ПО, которое надежно и эффективно работает в реальных компьютерах.

От организации процесса разработки программных систем напрямую зависят основные характеристики выполнения программного проекта – сроки выполнения, запланированный бюджет, качество выпускаемого продукта.



Русский термин
«**технология программирования**»
был введен русским
академиком

Андреем Петровичем Ершовым.



Разработка =

анализ + проектирование + программирование
(кодирование) + тестирование + отладка

Цели и задачи технологий разработки ПО

Современные крупные проекты ИС характеризуются следующими особенностями:

- **сложность описания** (достаточно большое количество функций, процессов, элементов данных и сложные взаимосвязи между ними), требующая тщательного моделирования и анализа данных и процессов;
- **наличие совокупности тесно взаимодействующих компонентов** (подсистем), имеющих свои локальные задачи и цели функционирования (например, традиционных приложений, связанных с обработкой транзакций и решением регламентных задач, и приложений аналитической обработки (поддержки принятия решений), использующих нерегламентированные запросы к данным большого объема);
- **отсутствие прямых аналогов**, ограничивающее возможность использования каких-либо типовых проектных решений и прикладных систем;
- **необходимость интеграции существующих и вновь разрабатываемых приложений;**
- **функционирование в неоднородной среде на нескольких аппаратных платформах;**
- **разобщенность и разнородность** отдельных групп разработчиков по уровню квалификации и сложившимся традициям использования тех или иных инструментальных средств;
- **существенная временная протяженность проекта**, обусловленная, с одной стороны, ограниченными возможностями коллектива разработчиков, и, с другой стороны, масштабами организации-заказчика и различной степенью готовности отдельных ее подразделений к внедрению ИС.

Основные понятия и определения

Программное обеспечение (Software)

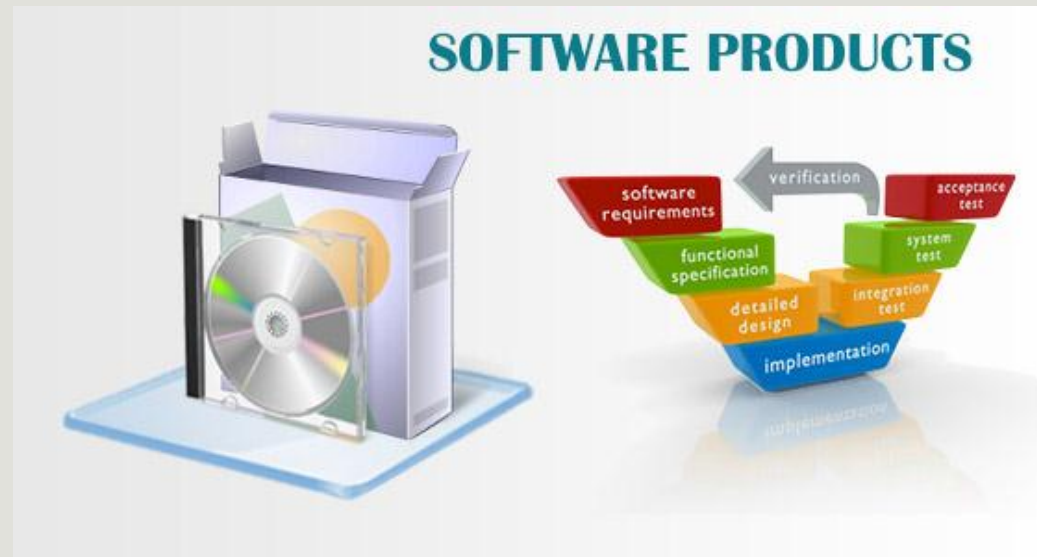
Полный набор или часть программ, процедур, правил и связанной с ними документации системы обработки информации. *ПО - интеллектуальный продукт, не зависящий от среды, на которой он записан.*



Программные средства (Software product) - набор компьютерных программ, процедур и, возможно, связанных с ними документации и данных. *Объем понятия, выражаемого термином "программные средства" включает в себя как частный случай объем понятия "программное обеспечение"*

Программный продукт (Software product) - набор компьютерных программ, процедур и, возможно, связанных с ними документации и данных, предназначенных для передачи пользователю.

Продукты включают промежуточные продукты и продукты, предназначенные для пользователей типа разработчиков и персонала сопровождения.



Программирование - это один из видов деятельности, входящих в цикл разработки программного обеспечения.

Проектирование программного обеспечения представляет собой процесс построения приложений реальных размеров и практической значимости, удовлетворяющих заданным требованиям функциональности и производительности, таких, например, как текстовый редактор, электронная таблица, операционная система или, скажем, программа контроля неисправностей космической станции.

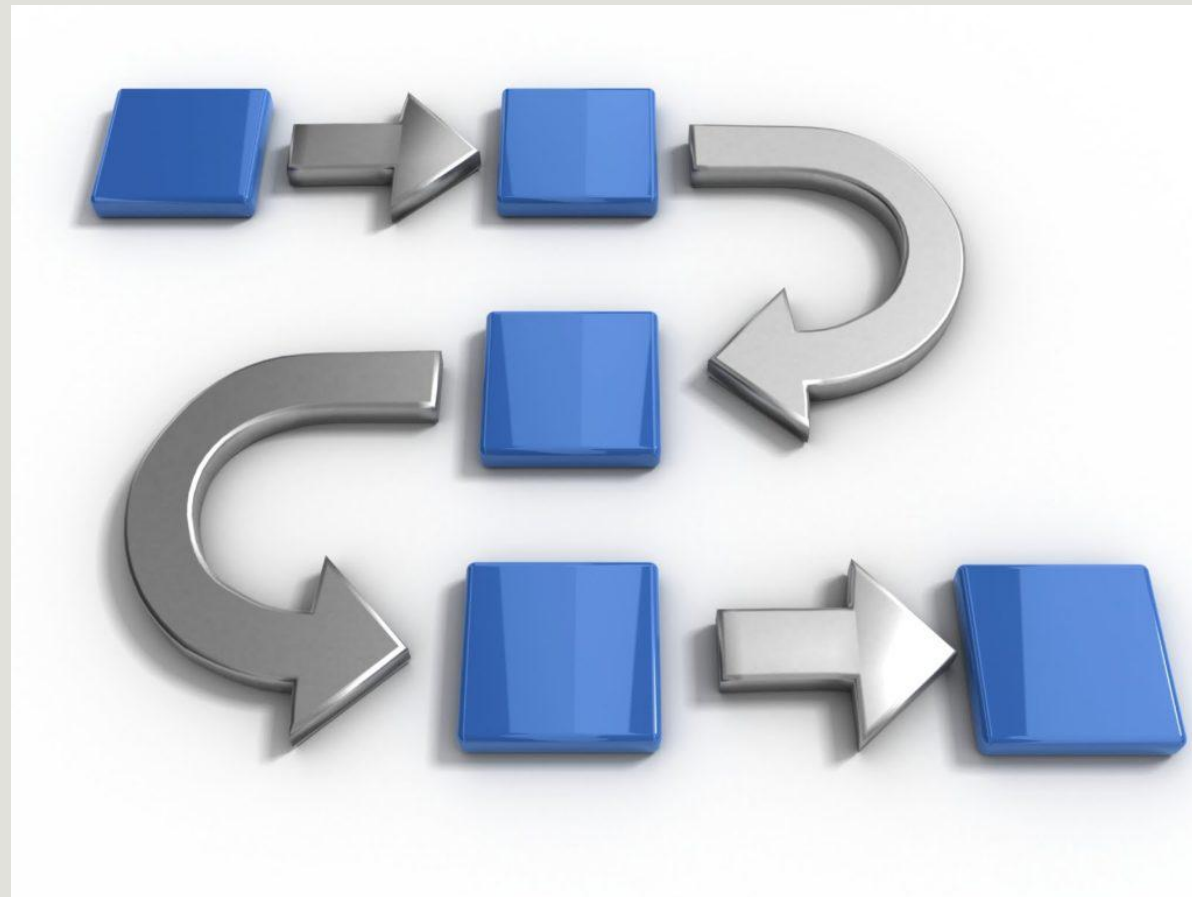
Персонал

те, кем это делается



Процесс

способ, которым это делается.



Проект

совокупность действий, необходимая для создания артефакта.

артефактами понимаются объектные модули, исходный код, документация, результаты тестов и измерений продуктивности.



Качество – приложения должны удовлетворять заранее определенному уровню качества.

Для достижения требуемого уровня качества применяются следующие методы:

- инспектирование (процесс проверки качества, ориентированный на команды разработчиков. Он применяется на всех этапах разработки);
- формальные методы (доказательство правильности – математическое или логическое);
- тестирование;
- методы управления проектом



Классификация типов программного обеспечения



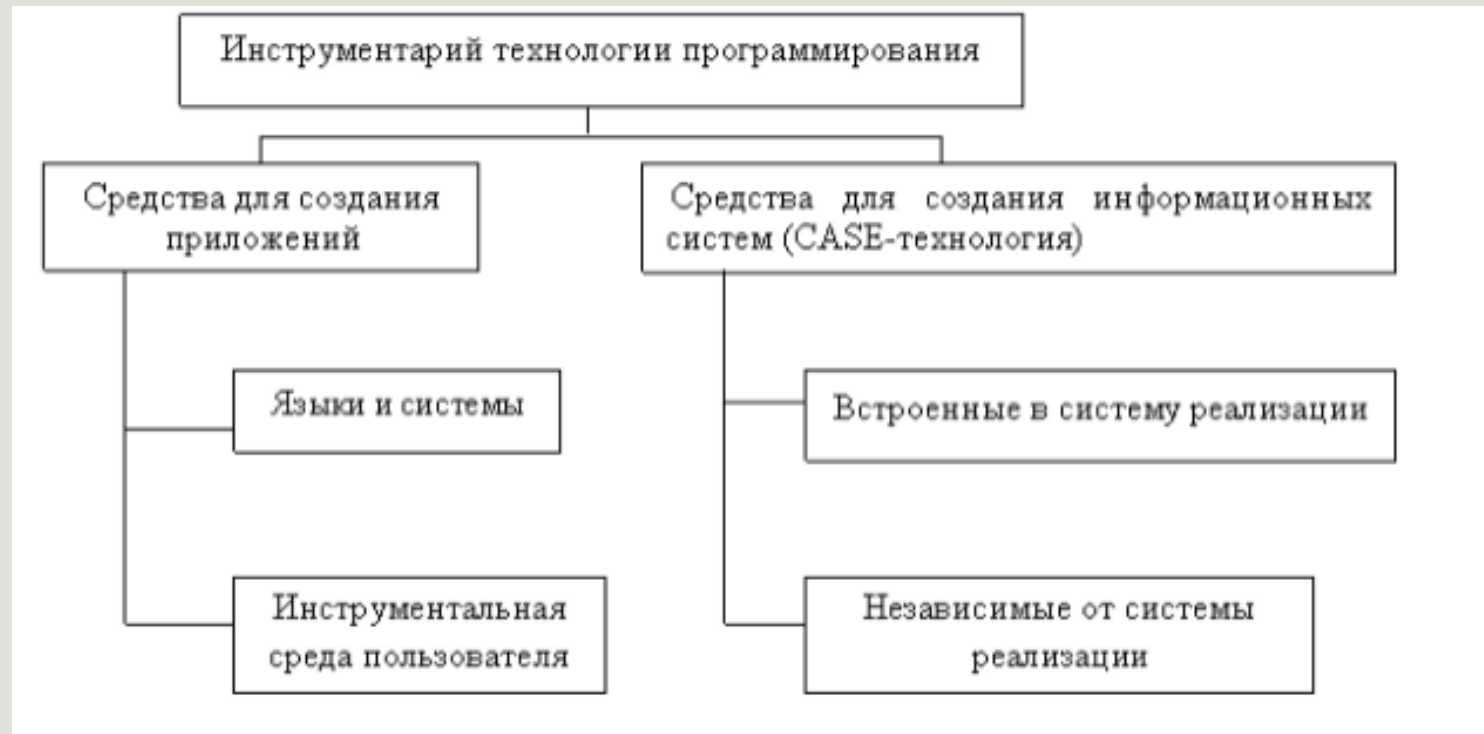
Технология разработки ПО должна охватывать разнообразные типы программ

- **Автономное:** устанавливаемое на одиночный компьютер; не связанное с другим программным и аппаратным обеспечением; пример - текстовый редактор.
- **Встроенное:** часть уникального приложения с привлечением аппаратного обеспечения; современные телевизоры, фотоаппараты, смартфоны, планшеты и уж тем более компьютера, являются неизменными носителями различного встроенного программного обеспечения. пример - автомобильный контроллер, биос.
- **Реального времени:** должны выполнять функции в течение малого интервала времени, обычно нескольких микросекунд; пример - программное обеспечение медицинского оборудования.
- **Сетевое:** состоит из частей, взаимодействующих через сеть; пример - основанная на вебтехнологии видеоигра

Инструментарий технологии программирования

Инструментарий технологии программирования

совокупность программ и программных комплексов, обеспечивающих технологию разработки, отладки и внедрения создаваемых программных продуктов.



CASE (*computer-aided software engineering*)

набор инструментов и методов программной инженерии для проектирования программного обеспечения, который помогает обеспечить высокое качество программ, отсутствие ошибок и простоту в обслуживании программных продуктов. Также под CASE понимают совокупность методов и средств проектирования информационных систем с использованием CASE-инструментов.



Средства для создания приложений

локальные средства, обеспечивающие выполнение отдельных видов работ по созданию программ, делятся на:

- языки и системы программирования; -
- инструментальная среда пользователя.

Язык программирования – формализованный язык для описания алгоритма решения задачи на компьютере.

Они делятся на классы:

- машинные языки – языки программирования, воспринимаемые аппаратной частью компьютера (машинные коды);
- машинно-ориентированные языки – языки программирования, которые отражают структуру конкретного типа компьютера (ассемблеры);
- алгоритмические языки – не зависящие от архитектуры компьютера языки программирования для отражения структуры алгоритма (Паскаль, бейсик, Фортран и др.);
- процедурно–ориентированные языки – языки программирования, где имеется возможность описания программы как совокупности процедур (подпрограмм).
- проблемно–ориентированные языки – предназначены для решения задач определенного класса (Lisp);

Системы программирования включают:

компилятор (транслятор);

- интегрированную среду разработки программ (не всегда); -

- отладчик;

- средства оптимизации кода программ;

- набор библиотек; -

- редактор связей;

- сервисные средства (утилиты) (для работы с библиотеками, текстовыми и двоичными файлами);

- справочные системы;

- систему поддержки и управления продуктами программного комплекса.

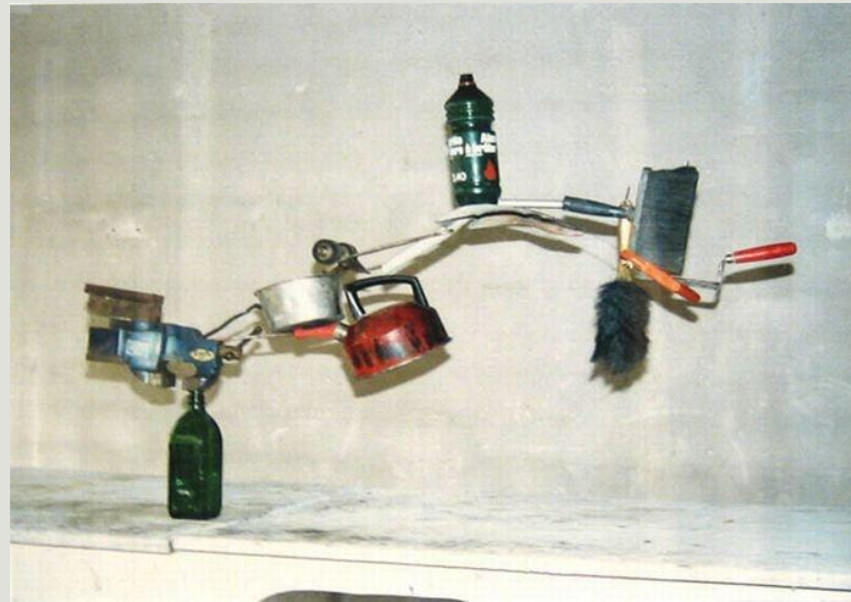
Инструментальная среда пользователя

это специальные средства, встроенные в пакеты прикладных программ, такие, как: библиотека функций, процедур, объектов и методов обработки; макрокоманды; клавишные макросы; языковые макросы; конструкторы экранных форм и объектов; генераторы приложений; языки запросов высокого уровня; конструкторы меню и др.

Сложная система. Признаки сложной системы.

Сложная система

система, состоящая из множества взаимодействующих составляющих (подсистем), вследствие чего сложная система приобретает новые свойства, которые отсутствуют на подсистемном уровне и не могут быть сведены к свойствам подсистемного уровня.



Признаками сложной системы являются:

- 1 Наличие общей задачи
- 2 Большое количество взаимодействующих компонентов
- 3 Возможность **декомпозиции системы**, (т.е. ее разбиение на взаимодействующие подсистемы, решающие автономные функциональные задачи).
- 4 Иерархическая архитектура системы и иерархия критериев качества
- 5 Сложность поведения системы, связанная со случайным характером внешних воздействий и большим количеством обратных связей внутри нее.
- 6 Устойчивость системы по отношению к внешним воздействиям. Наличие самоорганизации и самоадаптации при различных возмущениях.
- 7 Высокая надежность системы в целом при абсолютной надежности ее компонентов.

Разработка сложной системы

Роль декомпозиции

Роль абстракции

Роль иерархии