## **1.1 Что такое инженерия и назначение инженера**

**Инженер** – это специалист, осуществляющий инженерную деятельность. В свою очередь, **инженерная деятельность** – область технической деятельности, включающая в себя ряд специализированных областей и дисциплин. Эта деятельность направлена на практическое приложение и применение научных, экономических, социальных и практических знаний с целью обращения природных ресурсов на пользу человека. Конечной целью инженерной деятельности являются **изобретение, разработка, создание, внедрение, обслуживание и/или улучшение техники, материалов или процессов**.

**Проектирование** как особый вид инженерной деятельности формируется в начале 20-го столетия и связано с первоначальной деятельностью чертёжников, необходимостью особого (точного и понятного) графического изображения замысла инженера для передачи исполнителям на производстве.

**Проектирование следует отличать от конструирования.** Замысел проекта появляется, когда возникает социальный (в широком смысле) заказ, то есть потребность, вызванная конкуренцией, модернизацией, социальным развитием. Продукт проектировочной деятельности выражается в особой знаковой форме – в виде текста, чертежей, графиков, расчётов, моделей в памяти ЭВМ и, как правило, технико–экономического обоснования.

## **1.2 Особенности системотехнической деятельности**

Главной задачей инженера-системотехника является интеграция инженерной деятельности по отраслям и видам. Для выполнения организационной задачи требуется группа особых специалистов – координаторов. К ним относятся главный конструктор, руководитель темы, главный специалист проекта, руководитель научно-тематического отдела.

## **1.3 Социальная значимость результата инженерной деятельности**

## B инженерии сформировалось методология жизненного цикла изделия[[1]](#footnote-1) как совокупности всех существенных этапов «жизни» продукции. Включает в себя фазы формирования концепции, дизайнерской задумки, конструкторской проработки, технологической подготовки производства, изготовления, эксплуатации, обслуживания, утилизации и т.п.



## **1.4 Сущность и логика становления программной инженерии**

***Проект ПО* – совокупность спецификаций ПО (включающих модели и проектную документацию), обеспечивающих создание ПО в конкретной программно-технической среде**).

ПО можно разбить на два класса: «малое» и «большое».

*«****Малое» (простое****) программное обеспечение* имеет следующие характеристики:

● решает **одну несложную, четко поставленную задачу**;

**● размер исходного кода** не превышает **нескольких сотен строк**;

● **скорость** работы программного обеспечения и необходимые ему **ресурсы не играют большой роли**;

● **ущерб** от неправильной работы не имеет большого значения;

● **модернизация** программного обеспечения, дополнение его возможностей требуется редко;

**● как правило**, разрабатывается одним программистом или небольшой группой (**5 или менее человек**);

● **подробная документация** не требуется, ее может заменить исходный **код**, который **доступен**.

***«Большое» (сложное****) программное обеспечение* имеет **2-3 или более** характеристик из следующего перечня:

● решает **совокупность** взаимосвязанных задач;

● использование приносит **значимую выгоду**;

● **удобство** его использования играет **важную роль**;

● **обязательно** наличие **полной и понятной документации**;

● **низкая скорость** работы приводит к **потерям**;

● сбои, **неправильная работа**, наносит **ощутимый ущерб**;

● программы в составе ПО во время работы **взаимодействует** с другими **программами** и **программно-аппаратными комплексами**;

● работает на **разных** платформах;

● требуется **развитие**, **исправление ошибок**, добавление новых возможностей;

● группа разработчиков состоит из **более 5 человек**.

***системный подход*** – это методология исследования объекта любой природы как системы.  
***система***– это совокупность взаимосвязанных частей, работающих совместно для достижения некоторого результата.

## **1.5 Предпосылки возникновения программной инженерии**

Аналитики процессов в индустрии производства программного обеспечения сформулировали основные причины неудач при разработке ПО. Это:

● нечеткая и неполная формулировка требований;

● недостаточное вовлечение пользователей в работу над проектом;

● отсутствие необходимых ресурсов;

● неудовлетворительное планирование и отсутствие грамотного управления проектом;

● частое изменение требований и спецификаций;

● новизна и несовершенство используемой технологии;

● недостаточная поддержка со стороны высшего руководства;

● недостаточно высокая квалификация разработчиков, отсутствие необходимого опыта.

Отметим **особенности современных проектов ПО:**

● сложность – неотъемлемая характеристика создаваемого ПО;

● отсутствие полных аналогов и высокая доля вновь разрабатываемого ПО;

● наличие унаследованного ПО и необходимость его интеграции с разрабатываемым ПО;

● территориально распределенная и неоднородная среда функционирования;

● большое количество участников проектирования, разобщенность и разнородность отдельных групп разработчиков по уровню квалификации и опыту.

Разработка ПО имеет следующие **специфические** особенности:

● **неформальный** характер требований к ПО и формализованный основной объект разработки – программы;

● **творческий характер** разработки;

● **дуализм ПО**, которое, с одной стороны, является статическим объектом – совокупностью текстов, с другой стороны, – динамическим, поскольку при эксплуатации порождаются процессы обработки данных;

● при своем **использовании** (эксплуатации) ПО не **расходуется и не изнашивается, но морально устаревает**;

● «**неощутимость**», «воздушность», «квазинематериальность» ПО, что подталкивает к **безответственному переделыванию**, поскольку легко стереть и переписать, чего не сделаешь при проектировании зданий и аппаратуры.

***Инженерия ПО*** (**software engineering**) – совокупность инженерных методов и средств создания ПО.

**Фундаментальная идея программной инженерии: проектирование ПО является формальным процессом, который можно изучать и совершенствовать.**

Основными целями программной инженерии являются:

● Системы должны создаваться в короткие сроки и соответствовать требованиям заказчика на момент внедрения.

● Качество ПО должно быть высоким.

● Разработка ПО должна быть осуществлена в рамках выделенного бюджета.

● Системы должны работать на оборудовании заказчика, а также взаимодействовать с имеющимся ПО.

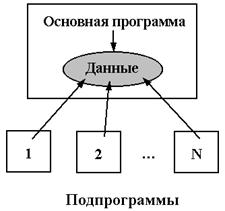
● Системы должны быть легко сопровождаемыми и масштабируемыми.

## **1.6 Этапы развития программной инженерии**

* 70-е и 80-е годы XX века– систематизация и стандартизация процессов создания ПО (структурный подход);
* 90-е годы- начало 21-го века – переход к сборочному, индустриальному способу создания ПО (объектно-ориентированный подход);
* с середины 90-х годов до настоящего времени – развитие компонентного подхода и сетевых технологий, создание CASE-технологий проектирования ПО

***Первый этап*** *– «стихийное» программирование*(от появления первых вычислительных машин до середины 60-х годов XX в). Первые программы имели простейшую структуру. Они состояли из собственно программы на машинном языке и обрабатываемых ею данных.



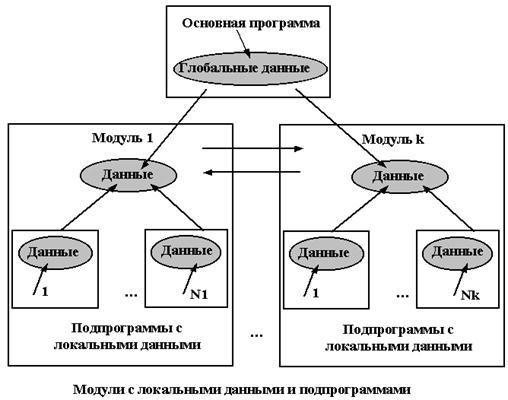


***Второй этап*** *– структурный подход к программированию* (60-70-е годы XX в.). В основе структурного подхода лежит *декомпозиция* (разбиение на части) сложных систем с целью последующей реализации в виде отдельных небольших (**до 40-50 операторов**) подпрограмм.

**Структурное программирование — методология разработки программного обеспечения, в основе которой лежит представление программы в виде иерархической структуры блоков.**

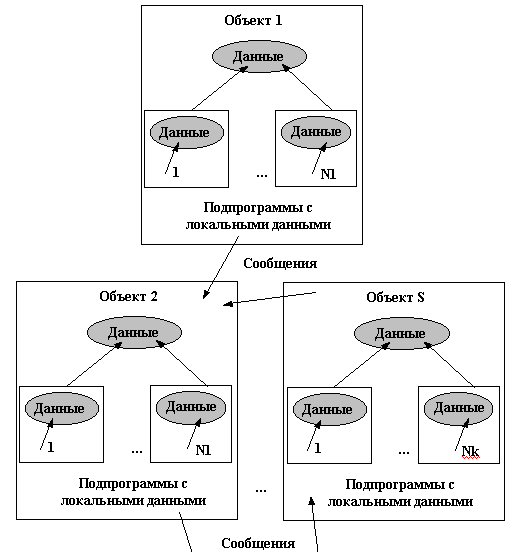
**Модульное программирование**. Главный принцип модульного программирования состоял в выделении таких фрагментов и оформлении их в виде модулей. **Каждый модуль снабжался описанием, в котором устанавливались правила его использования** – **интерфейс модуля**.

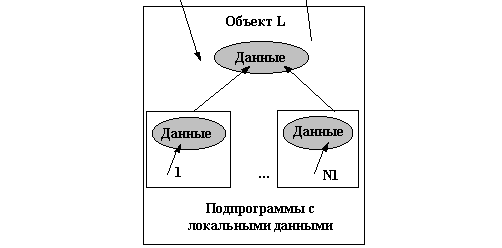
**Узким местом модульного программирования** является то, что ошибка в интерфейсе при вызове подпрограммы выявляется только при выполнении программы (из-за раздельной компиляции модулей обнаружить эти ошибки раньше невозможно).



Третий этап:

Объектно-ориентированное программирование определяется **как технология создания сложного программного обеспечения, основанная на представлении программы в виде совокупности *объектов***, каждый из которых является **экземпляром** определенного типа (*класса*), а классы образуют иерархию с *наследованием* свойств. Взаимодействие программных объектов в такой системе осуществляется путем передачи *сообщений* (рис. 1.6).





**Современная технология программирования** — **компонентный подход,** который предполагает построение программного обеспечения из отдельных компонентов — **физически отдельно существующих частей программного обеспечения, которые взаимодействуют между собой через стандартизованные двоичные интерфейсы**.

Четвертый этап:

**Компонент представляет собой готовый программный продукт**, который можно использовать как отдельно, так и совместно с подобными элементами в рамках решаемой задачи. В рамках этого подхода программное обеспечение строится из отдельных компонентов, физически отдельно существующих программных частей, которые распространяются в двоичном виде (в отличие от классов), взаимодействуют между собой посредством стандартизируемых интерфейсов и могут быть используемы в различных языках программирования

### ***1.6.2 Становление CASE-технологий***

Под термином CASE-средства понимаются программные средства, поддерживающие процессы:

* + создания и сопровождения ИС, включая анализ и формулировку требований,
  + проектирование прикладного ПО (приложений) и баз данных,
  + генерацию кода,
  + тестирование,
  + документирование,
  + обеспечение качества,
  + конфигурационное управление и
  + управление проектом, а также другие процессы.

**CASE-средства вместе с системным ПО и техническими средствами образуют полную среду разработки ИС.**

### ***1.6.3 Стандартизация и глобализация в сфере программной инженерии***

Развитие систем и средств вычислительной техники, телекоммуникационных систем и быстрое расширение сфер их применения привели к необходимости объединения конкретных вычислительных устройств и реализованных на их основе ИС в единые информационно-вычислительные системы и среды для формирования единого информационного пространства (Unified Information Area — UIA). Формирование такого пространства стало насущной необходимостью для решения многих важнейших экономических и социальных задач в ходе становления и развития информационного общества.

Такое пространство можно определить, как совокупность баз данных, хранилищ знаний, систем управления ими, информационно-коммуникационных систем и сетей**, методологий и технологий их разработки**, ведения и использования на основе единых принципов и общих правил, обеспечивающих информационное взаимодействие для удовлетворения потребностей пользователей.

1. [↑](#footnote-ref-1)