

Программная инженерия: назначение, основные принципы и понятия

Предпосылки и история

- Первый кризис программирования
- Повторное использование кода
 - Модульное программирование
- Рост сложности программ
 - Структурное программирование
- Модификация программ
 - Объектно-ориентированное программирование
- Продолжение кризиса программирования

Повторное использование кода

- Проблема
 - Дублирование фрагментов кода
- Модульное программирование
 - Выделение фрагментов в модули
 - Повторное использование модулей
 - Создание библиотек модулей

Рост сложности программ

- Проблема
 - Сложные программные комплексы
 - Объем кода, к-во связей, к-во разработчиков, к-во пользователей
 - Жизненный цикл: стадии внедрения и сопровождения
- Структурное программирование
 - «Правильное» проектирование и кодирование
 - Основные принципы:
 - Нисходящее проектирование
 - Применение специальных языков проектирования
 - Дисциплина проектирования и разработки:
 - планирование и документирование проекта
 - поддержка соответствие кода проектной документации
 - Структурное кодирование (линейный блок, If-then-else, цикл)

Модификация программ

- Проблема
 - изменения в проекте и программе без изменения ранее написанного кода
- Объектно-ориентированное программирование
 - Класс – модуль со свойствами, поведением, обязанностями
 - Парадигмы ООП:
 - Инкапсуляция и сокрытие деталей
 - Наследование
 - Полиморфизм

Некоторые итоги

- Главная цель программной инженерии - сокращение стоимости ПО
- Сформировались основные принципы и методы проектирования ПО:
 - Жизненный цикл ПО
 - Модульное программирование
 - Структурное проектирование и программирование
 - Объектно-ориентированное проектирование и программирование

Основные определения

- Программная инженерия – это
 - установление и использование обоснованных инженерных принципов (методов) для экономного получения ПО, которое надежно и работает на реальных машинах.
 - та форма инженерии, которая применяет принципы информатики (computer science) и математики для рентабельного решения проблем ПО.
 - применение систематического, дисциплинированного, измеряемого подхода к разработке, использованию и сопровождению ПО.
 - дисциплина, целью которой является создание качественного ПО, которое завершается вовремя, не превышает выделенных бюджетных средств и удовлетворяет выдвигаемым требованиям.

Разберемся в вопросах

- Что такое программное обеспечение (software)?
- Что такое программная инженерия?
- В чем разница между программной инженерией (software engineering) и информатикой (computer science)?
- В чем отличие программной инженерии от других инженерий?
- Из чего складывается стоимость ПО?

Программное обеспечение?

- Компьютерные программы и связанная с ними документация и данные (ISO/IEC 12207)
- Программные продукты могут разрабатываться для конкретного заказчика или для обобщенного рынка
- Программные продукты могут быть
 - Коробочными (generic products, shrink-wrapped software), т.е. разработанными для продажи многим различным заказчикам
 - Заказными (bespoke, custom), т.е. разработанными для одного покупателя по его спецификациям

Программная инженерия

Инженерная дисциплина, которая связана со всеми аспектами производства ПО от начальных стадий создания спецификации до поддержки системы после сдачи в эксплуатацию.

Две ключевые фразы в определении:

- **Инженерная дисциплина**
 - Ориентация на практический результат
 - Применение теорий, методов и способов для достижения результата
 - Лучшие практики (best practices)
 - При ограниченном ресурсе времени, бюджета, оборудования, людей
- **Все аспекты производства ПО**
 - Управление программными проектами
 - Разработка средств, методов и теорий

В чем отличия от информатики?

- Информатика (computer science) - теория и методы вычислительных и программных систем
- Программная инженерия - практические проблемы создания ПО
- Информатика – теоретический фундамент программной инженерии
 - Не всегда достаточный
 - Не единственный (финансы, управление проектом, работа с заказчиком, ...)

В чем отличие от других инженерий?

- Вопросы:
 - Почему так велика доля провальных проектов?
 - Можно ли применять опыт других инженерий?
- Фазы жизненного цикла любого продукта:
 - Проектирование, создание образца, испытание, производство, эксплуатация
- Программа – не материальный объект:
 - Фазы производства и изготовления образца отсутствуют
 - Стоимость программы – это стоимость проектирования
 - У коробочных продуктов «размазывается» по копиям
 - У заказных продуктов остается высокой.

В чем еще отличие от других инженерий?

- Программа – искусственный объект
 - Нет объективных законов контроля проекта
 - Тестирование – единственный способ проверки
- Программная инженерия – молодая дисциплина

Стоимость ПО

- Зависит от типа ПО, методологии разработки и метода оценки
- Типовое распределение:
 - 15% - спецификация
 - 25% - проектирование
 - 20% - разработка
 - 40% - интеграция и тестирование
- Коробочное ПО
 - Рост доли тестирования за счет спецификации
- Заказное ПО
 - Рост доли тестирования за счет проектирования и разработки

Программный процесс

- Жизненный цикл – непрерывный процесс с момента принятия решения о создании ПО до снятия его с эксплуатации.
- Процесс – совокупность действий и задач, имеющих целью достижение значимого результата.
- Основные процессы (этапы или фазы) ЖЦ:
 - Спецификация требований
 - Разработка проекта программы
 - Кодирование
 - Тестирование
 - Документирование

Программный процесс

- Дополнительные (нефункциональные) процессы:
 - создание инфраструктуры, управление конфигурацией, управление качеством, обучение, разрешение противоречий, ...
- Установление процесса:
 - Описание процесса
 - Обучение процессу
 - Введение метрик
 - Контроль выполнения
 - Усовершенствование

Модель программного процесса

- Модель программного процесса — это упрощенное описание программного процесса, представленное с некоторой точки зрения.
- Модели жизненного цикла:
 - Водопадная (каскадная) модель
 - Спиральная (циклическая) модель
 - Компонентная модель
 - Формальная модель
 - Комбинированные модели
- Модели организации работ:
 - Модель потока работ (workflow model)
 - Модель потоков данных (data flow model)
 - Ролевая модель

Методы программной инженерии

- Метод программной инженерии — это структурный подход к созданию ПО:
 - как высококачественного продукта
 - экономически эффективным способом.
- Наиболее известные методы:
 - Структурного анализа и проектирования (Том Де Марко, 1978),
 - Сущность-связь (Чен, 1976),
 - Объектно-ориентированного анализа и проектирования (Буч, 1994, Рамбо, 1991).

- Цель - создание и поэтапное преобразование моделей ПО
- Методы должны включать в себя следующие компоненты:
 - Описание моделей системы и нотация
 - Правила и ограничения
 - Рекомендации
 - Руководство по применению метода
- Нет идеальных методов, нет абсолютных методов

CASE системы

(Computer Aided System Engineering)

- Определение CASE

Использование компьютеров для поддержки процесса создания программ. Может включать инструментальные программные средства для проектирования ПО, формирования требований, кодирования, тестирования, документирования и других действий по созданию ПО

- Эволюция CASE

- Ассемблеры, компиляторы, интерпретаторы
- Трассировщики, символические отладчики
- Текстовые редакторы, системы анализа исходных текстов
- Средства для анализа требований и проектирования
- Среды визуального программирования
- Средства генерации исходных кодов
- Средства поддержки полного жизненного цикла ПО

- Классификация CASE средств:
 - По уровню применения:
 - Upper CASE -средства анализа требований
 - Middle CASE - средства проектирования
 - Low CASE - средства разработки приложений
 - Специализированные
 - Средства проектирования баз данных
 - Средства реинжиниринга (восстановления)
 - Вспомогательные
 - Планирования и управления проектом
 - Конфигурационного управления
 - Тестирования
- Интегрированные CASE
- Главное правило: сначала метод – потом CASE

Свойства хорошей программы

- Удовлетворять функциональным требованиям
- Нефункциональные требования:
 - Сопровождаемость (maintainability)
 - Возможность дальнейшего развития.
 - Надежность (dependability)
 - Отказоустойчивость, безопасность, защищенность
 - Эффективность (efficiency)
 - Память, процессорное время, каналы связи.
 - Удобство использования (usability)
 - Понятно пользователю

Основные трудности

Главная проблема: поиск универсального метода и процесса для создания ПО любого типа в любых условиях.

Основные трудности:

- Наследование ранее созданного ПО (legacy systems).
 - сопровождение – поддержка и развитие старого ПО.
- Разнородность программных систем.
 - распределенные сети, разнородное оборудование, разные среды, различные ОС
- Сокращение времени на разработку.
 - сократить время разработки ПО без снижения его качества.

Трудности часто оказываются связанными между собой

Профессиональные и этические требования

- Развитие IT индустрии оказывает все большее воздействие на общество
 - Internet, телекоммуникации, IP телефония, компьютерные игры ...
- IT специалисты работают в правовом и социальном окружении, под действием международных, национальных и местных законодательств.

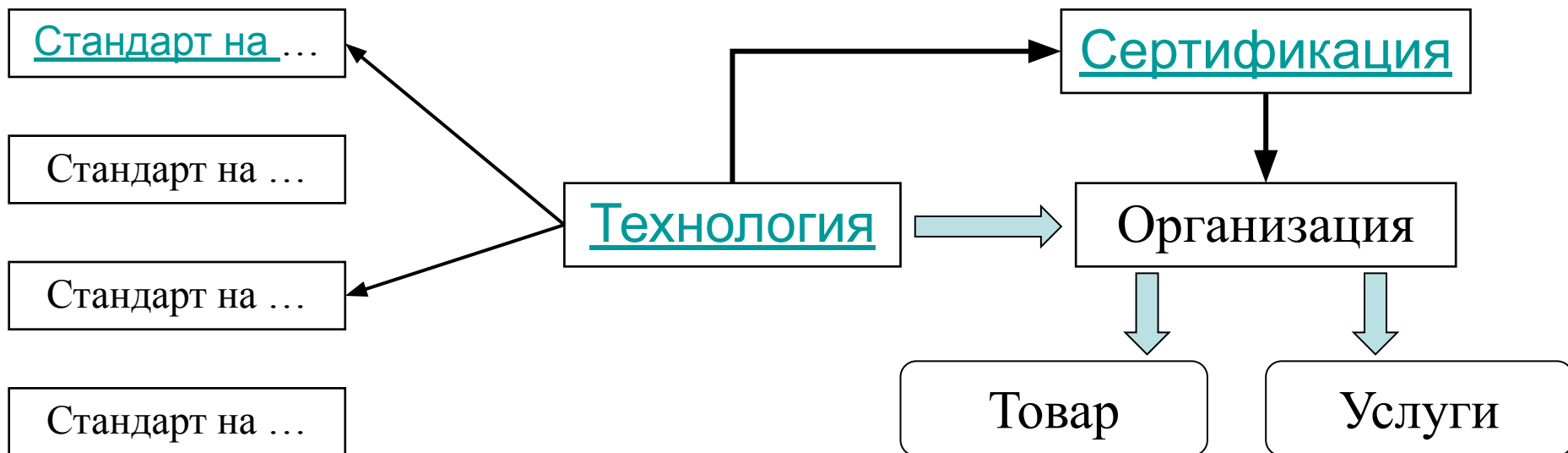
Профессиональные и этические требования

- Более тонкие профессиональные обязательства:
 - Конфиденциальность
 - Неразглашение сведений о своих работодателях или заказчиков независимо от того, подписывалось ли ими соответствующее соглашение.
 - Компетентность
 - не должен завышать свой уровень компетентности и браться за работу, не соответствующую этому уровню.
 - Защита интеллектуальной собственности
 - соблюдать законодательство при использовании чужой интеллектуальной собственности.
 - защищать интеллектуальную собственность работодателя и клиента.
 - Злоупотребление компьютером
 - от игр в компьютерные игрушки на рабочем месте до распространения вирусов и т.п.

Стандартизация и стандарты

- Что такое стандарты и сертификация на соответствие стандартам?
- Какие бывают стандарты?
- Кто разрабатывает стандарты программной инженерии?
- Основные стандарты программной инженерии

Стандарты и сертификация



Технология

- Знание (логия) мастерства (техно):
 - совокупность приёмов и способов производства;
 - научная дисциплина
 - сами операции производства
 - описание производственных процессов,
 - инструкции по их выполнению,
 - технологические правила, требования, карты, графики и др.

Что такое стандарт?

- От англ. standard - норма, образец, мерило:
 - нормативный документ, устанавливающий комплекс норм и правил;
 - типовой образец, эталон, модель,
- Стандарт может быть разработан на
 - материально-технические предметы
 - нормы, правила, требования
- Стандартизация распространяется на все сферы человеческой деятельности:
 - науку, технику, промышленное и с.-х. производство, строительство, здравоохранение, транспорт и т.д.

Что такое сертификация?

- Certification - "сделано верно“:
- Заявление о соответствии поставщика (изготовителя) (supplier's declaration)
 - письменная гарантия соответствия продукции заданным требованиям
- Сертификация соответствия
 - процедура, посредством которой третья сторона дает письменную гарантию, что продукция, процесс, услуга соответствуют заданным требованиям.

Какие бывают стандарты?

- Типы стандартов
 - Корпоративные
 - Отраслевые
 - Государственные
 - Международные
- Виды стандартов
 - Рекомендательные
 - Обязательные

Кто разрабатывает стандарты Software Engineering (SE)

Сокр.	Name	Название
ISO	International Organization for Standardization	Международная организация по стандартизации
ACM	Association for Computing Machinery	Ассоциация вычислительной техники
SEI	Software Engineering Institute	Институт программного инжиниринга
PMI	Project Management Institute	Институт управления проектами
IEEE	Institute of Electrical and Electronics Engineers	Институт инженеров по электротехнике и электронике
ANSI	American National Standards Institute	Американский национальный институт стандартов

Основные стандарты SE

- [ISO/IEC 12207](#) - Information Technology - Software Life Cycle Processes
- [SEI CMM](#) - Capability Maturity Model (for Software)
- [ISO/IEC 15504](#) - Software Process Assessment
- [PMBOK](#) - Project Management Body of Knowledge
- [SWEBOK](#) - Software Engineering Body of Knowledge
- [ACM/IEEE CC2001](#) - Computing Curricula 2001

IEEE SWEBOK

- IEEE Computer Society Software Engineering Body of Knowledge – Свод знаний по программной инженерии, 18 мая 2004 г.
- Области знаний программной инженерии:
 1. Software Requirements – требования к ПО
 2. Software Design – проектирование ПО
 3. Software Construction – конструирование ПО
 4. Software Testing – тестирование ПО
 5. Software Maintenance – сопровождение ПО
 6. Software Configuration Management – управление конфигурац.
 7. Software Engineering Management – управление IT проектом
 8. Software Engineering Process – процесс ПИ
 9. Software Engineering Tools and Methods – методы и инструменты
 10. Software Quality – качество ПО

ACM/IEEE Computing Curricula

- ACM/IEEE Computing Curricula 2001 – Академический образовательный стандарт в области компьютерных наук 2001
- Основные разделы компьютерных наук:
 - Computer science – Информатика (2001г);
 - Computer engineering – Компьютерная инженерия;
 - Software engineering – Программная инженерия (2004г.)
 - Information systems – Информационные системы.
- Сайты:
 - CC2001: <http://www.computer.org/education/cc2001>
 - CC2001. Информатика: <http://se.math.spbu.ru/cc2001>

ACM/IEEE Computing Curricula

- Области знаний Software engineering:
 - Computing Essentials - Основы применения ЭВМ
 - Mathematical & Engineering Fundamentals - Математические и инженерные основы
 - Professional Practice - Профессиональная практика
 - Software Modeling & Analysis - Моделирование и анализ ПО
 - Software Design - Проектирование ПО
 - Software V & V –Верификация и валидация ПО
 - Software Evolution - Эволюция ПО
 - Software Process - Процесс ПО
 - Software Quality - Качество ПО
 - Software Management -Управление проектом