

**Тема: «Жизненный цикл и этапы  
разработки программного обеспечения »**

**Жизненный цикл программного обеспечения (ПО)** – это непрерывный процесс, который начинается с момента принятия решения о необходимости его создания и заканчивается в момент его полного изъятия из эксплуатации.

Существует целый ряд стандартов, регламентирующих ЖЦ ПО, а в некоторых случаях и процессы разработки.

Среди наиболее известных стандартов можно выделить следующие:

***ГОСТ 34.601-90*** - распространяется на автоматизированные системы и устанавливает стадии и этапы их создания. Кроме того, в стандарте содержится описание содержания работ на каждом этапе. Стадии и этапы работы, закрепленные в стандарте, в большей степени соответствуют каскадной модели жизненного цикла.

***ISO/IEC 12207:1995*** - стандарт на процессы и организацию жизненного цикла. Распространяется на все виды заказного ПО. Стандарт не содержит описания фаз, стадий и этапов.

***Rational Unified Process (RUP)*** предлагает итеративную модель разработки, включающую четыре фазы: начало, исследование, построение и внедрение.

Каждая фаза может быть разбита на этапы (итерации), в результате которых выпускается версия для внутреннего или внешнего использования.

Прохождение через четыре основные фазы называется циклом разработки, каждый цикл завершается генерацией версии системы.

Если после этого работа над проектом не прекращается, то полученный продукт продолжает развиваться и снова минует те же фазы. Суть работы в рамках RUP - это создание и сопровождение моделей на базе UML.

***Microsoft Solution Framework (MSF)*** сходна с RUP, так же включает четыре фазы: анализ, проектирование, разработка, стабилизация, является итерационной, предполагает использование объектно-ориентированного моделирования.

MSF в сравнении с RUP в большей степени ориентирована на разработку бизнес-приложений.

*Extreme Programming (XP)*. Экстремальное программирование (самая новая среди рассматриваемых методологий) сформировалось в 1996 году. В основе методологии командная работа, эффективная коммуникация между заказчиком и исполнителем в течение всего проекта по разработке ИС, а разработка ведется с использованием последовательно дорабатываемых прототипов.

# Основные процессы жизненного цикла

В соответствии с базовым международным стандартом ISO/IEC 12207 все процессы ЖЦ ПО делятся на три группы:

## 1. Основные процессы:

- ✓ приобретение;
- ✓ поставка;
- ✓ разработка;
- ✓ эксплуатация;
- ✓ сопровождение.

## **2. Вспомогательные процессы:**

- ✓ документирование;
- ✓ управление конфигурацией;
- ✓ обеспечение качества;
- ✓ разрешение проблем;
- ✓ аудит;
- ✓ аттестация;
- ✓ совместная оценка;
- ✓ верификация.

## **3. Организационные процессы:**

- ✓ создание инфраструктуры;
- ✓ управление;
- ✓ обучение;
- ✓ усовершенствование.

Для поддержки практического применения стандарта ISO/IEC 12207 разработан ряд технологических документов:

**Руководство для ISO/IEC 12207** (ISO/IEC TR 15271:1998 Information technology -Guide for ISO/IEC 12207)

**Руководство по применению ISO/IEC 12207** к управлению проектами (ISO/IECTR 16326:1999 Software engineering - Guide for the application of ISO/IEC 12207 to project management).(заказчик) ИС)

## Этапы, характерные для решения большинства задач с помощью компьютера:

### 1) Постановка задачи:

- ✓ сбор информации;
- ✓ формулировка условий;
- ✓ определение конечных целей решения задачи;
- ✓ определение формы выдачи результатов;
- ✓ описание данных.

Т.е. в ходе выполнения этого этапа должны быть получены четкие ответы на следующие вопросы:

- Что должна делать программа?
- В чем состоят реальные проблемы, разрешению которых она должна способствовать?
- Что представляют собой входные данные?
- Какими должны быть выходные данные?
- Какими ресурсами располагает проектировщик?



## **2) Этап формализации:**

- ✓ анализ и исследование задачи (модели);
- ✓ анализ существующих аналогов;
- ✓ анализ технических и программных средств;
- ✓ разработка математической модели;
- ✓ разработка структур данных.

**Моделирование** – создание и исследование модели, замещающей оригинал.

**Модель** - это новый объект, который отражает некоторые существенные стороны изучаемого объекта.

Различают модели:

✓ **детерминированные** (в системах отсутствуют случайные воздействия) и **стохастические** (в системах присутствуют вероятностные воздействия);

✓ **предметные** (материальные) и **знаковые** (информационные);

✓ **статические** (описывающие систему в определенный момент времени) и **динамические** (рассматривающие поведение системы во времени).

В свою очередь, динамические модели подразделяют на **дискретные**, (в которых все события происходят по интервалам времени), и **непрерывные** (все события происходят непрерывно во времени).

Информационная модель (в общем случае) – это описание объекта моделирования.

**Формализация** – это замена реального объекта или процесса его формальным описанием, т.е. информационной моделью.

Формы информационных моделей:

- ✓ вербальные (словесные, описательные),
- ✓ графические,
- ✓ табличные,
- ✓ математические,
- ✓ формально-логические.

## Уровни информационных моделей:

- ✓ **концептуальная модель** – обеспечивает интегрированное представление о предметной области (технологические карты, план производства); имеет слабо-формализованный характер
- ✓ **логическая** – формализуется из предыдущей (к.м.) путем выделения конкретной части, ее детализации и формализации
- ✓ **математическая модель** – л.м., формализующая на языке математики взаимосвязи в выделенной предметной области.

### **3) Разработка алгоритма:**

- ✓ выбор метода решения задачи;
- ✓ выбор формы записи алгоритма;
- ✓ проектирование алгоритма.

### **4) Программирование:**

- ✓ выбор языка программирования;
- ✓ уточнение способов организации данных;
- ✓ запись алгоритма на выбранном языке.

### **5) Тестирование и отладка программы:**

- ✓ синтаксическая отладка;
- ✓ отладка семантики и логической структуры;
- ✓ тестовые расчеты и анализ результатов тестирования;
- ✓ совершенствование программы.

Когда программа закончена (готова работоспособная *альфа-версия*), она поступает на тестирование.

***Тестирование*** - проверка правильности работы программы в целом, либо ее составных частей.

***Отладка*** - это процесс поиска и устранения ошибок (синтаксических и логических) в программе, производимый по результатам ее выполнения на компьютере.

**6) Анализ результатов решения задачи** и уточнение в случае необходимости математической модели с повторным выполнением этапов 2-5.

**7) Сопровождение программы:** ее доработка для решения конкретных задач, а также составление технической документации к решенной задаче, к математической модели, к алгоритму, к программе, к набору тестов, к использованию.

***Сопровождение программы*** - документация и инструкция по эксплуатации программы.

