

image not found or type unknown



Классический жизненный цикл

Период создания и использования информационных систем, охватывающий совокупность состояний и этапов, которые система проходит в своем развитии от момента принятия решения о создании системы до момента прекращения функционирования системы, называется жизненным циклом ИС.

Для каждого этапа жизненного цикла определяются состав и последовательность выполняемых работ, получаемые результаты, методы и средства, необходимые для выполнения работ, роли и ответственность участников и т.д.

Такое формальное описание жизненного цикла ИС позволяет спланировать и организовать процесс коллективной разработки и обеспечить управление этим процессом.

Жизненный цикл ИС можно представить как ряд событий, происходящих с системой в процессе ее создания и использования.

Модель жизненного цикла отражает различные состояния системы, начиная с момента возникновения необходимости в данной ИС и заканчивая моментом ее полного выхода из употребления. Модель жизненного цикла - структура, содержащая процессы, действия и задачи, которые осуществляются в ходе разработки, функционирования и сопровождения программного продукта в течение всей жизни системы, от определения требований до завершения ее использования.

Жизненный цикл заканчивается, как правило, не в результате физического износа информационной системы, а в результате морального устаревания. Моральный износ, моральное устаревание- прекращение удовлетворения требований к информационной системе. При этом возможные модификации информационной системы экономически невыгодны или невозможны, что влечет за собой необходимость разработки новой информационной системы. Для информационных технологий является вполне естественным то, что они устаревают и заменяются новыми.

Суть содержания жизненного цикла-разработки ИС в различных подходах одинакова и сводится к выполнению следующих стадий:

1. Планирование и анализ требований (предпроектная стадия)

- Исследование и анализ существующей ИС
- формирование требований к новой ИС
- изучение объекта автоматизации
- выбор и разработка варианта концепции системы
- создание и утверждение технико-экономического обоснования
- разработка и утверждение технического задания на проектирование информационной системы

2. Проектирование (техническое проектирование, логическое проектирование) выбор проектных решений по всем аспектам разработки информационной системы

- описание всех компонентов информационной системы
- оформление и утверждение технического проекта
- выбор и разработка математических методов и алгоритмов программ
- корректировка структур баз данных
- создание документации на поставку и установку программных продуктов
- Выбор комплекса технических средств ИС
- разработка технорабочего проекта ИС

3. Реализация (рабочее проектирование, физическое проектирование, программирование) получение и установка технических средств

- разработка, тестирование и доводка программ
- получение и установка программных средств
- наполнение баз данных

- разработка инструкций по эксплуатации программного обеспечения, технических средств, должностных инструкций для персонала

- оформление рабочего проекта

4. Внедрение (тестирование, опытная эксплуатация) ввод в опытную эксплуатацию технических средств

- ввод в опытную эксплуатацию программных средств

- обучение персонала

- поэтапное внедрение ИС в эксплуатацию по подразделениям экономического объекта

- оформление акта о приемо-сдаточных испытаниях ИС

5. Эксплуатация ИС (сопровождение, модернизация) повседневная эксплуатация ИС

- сбор статистики о функционировании ИС

- исправление ошибок и недоработок

- оформление требований к модернизации ИС и ее выполнение.

Часто второй и третий этапы объединяют в одну стадию, называемую техно-рабочим проектированием или системным синтезом.

На каждом этапе жизненного цикла формируется набор документов и технических решений, они являются исходными для последующих решений.

В настоящее время известны и используются следующие модели жизненного цикла:

- Каскадная модель (рис1). предусматривает последовательное выполнение всех этапов проекта в строго фиксированном порядке. Переход на следующий этап означает полное завершение работ на предыдущем этапе.



(Рис1.)

Поэтапная модель с промежуточным контролем(рис.2) Разработка ИС ведется итерациями с циклами обратной связи между этапами. Межэтапные корректировки позволяют учитывать реально существующее взаимовлияние результатов разработки на различных этапах, но время жизни каждого из этапов растягивается на весь период разработки.



(Рис.2)

Спиральная модель(рис.3) На каждом витке спирали выполняется создание очередной версии продукта, уточняются требования проекта, определяется его качество, и планируются работы следующего витка. Происходит последовательное углубление и конкретизация деталей проекта ИС. Особое внимание уделяется начальным этапам разработки - анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством



рис.3

На практике наибольшее распространение получили две основные модели жизненного цикла:

- каскадная модель (характерна для периода 1970-1985 гг.);
- спиральная модель (характерна для периода после 1986 г.).

Существует целый ряд стандартов, регламентирующих жизненный цикл 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.
- Custom Development Method (методика Oracle) по разработке прикладных информационных систем - технологический материал, детализированный до уровня заготовок проектных документов, рассчитанных на использование в проектах с применением Oracle. Применяется CDM для классической модели ЖЦ

(предусмотрены все работы/задачи и этапы), а также для технологий «быстрой разработки» (Fast Track), рекомендуемых в случае малых проектов.

- 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 - цикл разработки, каждый цикл завершается генерацией версии системы. Если работа над проектом не прекращается, то полученный продукт снова минует те же фазы. Суть работы в рамках RUP - это создание и сопровождение моделей на базе UML.

- 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.