

Информационные системы

Тема 2

Жизненный цикл ИС. Основы применения инструментальных средств ИТ . Создание автоматизированных информационных систем.

2.1 Жизненный цикл ИС.

2.2 Основы применения
инструментальных средств ИТ .

2.3 Создание автоматизированных
информационных систем.

2.1 Жизненный цикл ИС

Необходимость проектирования ИС может обуславливаться разработкой и внедрением информационных технологий в организации (построение новой информационной системы) либо при модернизации существующих информационных процессов, либо при реорганизации деятельности предприятия (проведении бизнес-реинжиниринга). Потребности проектирования ИС указывают:

1. для достижения каких целей необходимо разработать систему;
2. к какому моменту времени целесообразно осуществить разработку;
3. какие затраты необходимо осуществить для проектирования системы.

Проектирование ИС является трудоемким, длительным и динамическим процессом. Технологии проектирования, применяемые в современных условиях, предполагают поэтапную разработку системы. Этапы по общности целей могут объединяться в стадии. Совокупность стадий и этапов, которые проходит ИС в своем развитии от момента принятия решения о создании системы до момента прекращения функционирования системы, называется **жизненным циклом ИС**.

2.1 Жизненный цикл ИС

- 1) **Планирование и анализ требований** (предпроектная стадия) — системный анализ. Проводится исследование и анализ существующей информационной системы, определяются требования к создаваемой ИС, формируются технико-экономическое обоснование (ТЭО) и техническое задание (ТЗ) на разработку ИС.

Системный анализ. Основными целями этапа являются: 1) формулировка потребностей в новой ИС (определение всех недостатков существующей ИС); 2) выбор направления и определение экономической обоснованности проектирования ИС.

Системный анализ ИС начинается с описания и анализа функционирования рассматриваемого объекта в соответствии с требованиями (целями), которые предъявляются к нему. В результате этого этапа выявляются недостатки существующей ИС, на основе которых формулируется потребность в совершенствовании системы управления этим объектом, и ставится задача определения экономически обоснованной необходимости автоматизации определенных функций управления (создается технико-экономическое обоснование проекта ИС). После определения этой потребности возникает проблема выбора направлений совершенствования объекта на основе выбора программно-технических средств. Результаты оформляются и виде технического задания на проект, в котором отражаются технические условия и требования к ИС, а также ограничения по ресурсам проектирования. Требования к ИС определяются в терминах функций, реализуемых системой.

2.1 Жизненный цикл ИС

2) **Проектирование (техническое и логическое проектирование).** В соответствии с требованиями формируются состав автоматизируемых функций (функциональная архитектура) и состав обеспечивающих подсистем (системная архитектура), проводится оформление технической проектом ИС;

Этап проектировании предполагает:

1) проектирование функциональной архитектуры ИС, которая отражает структуру функциональных подсистем и связей между ними; является наиболее ответственным и важным этапом с точки зрения качества всей последующей разработки ИС

2.1 Жизненный цикл ИС

2) проектирование системной архитектуры ИС (состав обеспечивающих подсистем); Построение системной архитектуры на основе функциональной предполагает определение элементов и модулей информационного, технического, программного обеспечения и других обеспечивающих подсистем, связей по информации и управлению между выделенными элементами и разработку технологии обработки информации.

3) **Реализация** (рабочее и физическое проектирование, кодирование). Разработка и настройка программ, формулировка рабочих инструкций для персонала, создание информационного обеспечения, включая формирование и наполнение баз данных, оформление рабочего проекта.

2.1 Жизненный цикл ИС

Важной особенностью жизненного цикла ИС является его повторяемость (цикличность) «системный анализ» - «разработка сопровождение — системный анализ». Это соответствует представлению об ИС, как о развивающейся, динамической системе. При первом выполнении стадии «Разработка» создается проект ПС, а при последующих реализациях данной стадии осуществляется модификация проекта для поддержания его в актуальном состоянии.

Существуют различные модели жизненного цикла. Среди известных можно выделить следующие:

1. **Каскадная модель** (до 70-х годов) - последовательный переход на следующий этап только после полного завершения предыдущего;

Достоинство - планирование времени осуществления всех этапов проекта, упорядочении хода конструирования.

Недостатки каскадной модели:

- ◆ модель недостаточно гибкая - реальные проекты часто требуют отклонения от стандартной последовательности шагов;
- ◆ цикл основан на точной формулировке исходных требований к ПО (реально в начале проекта требования заказчика определены лишь частично);
- ◆ результаты проекта доступны заказчику только в конце работы.

2.1 Жизненный цикл ИС

Итерационная модель (70-80-е годы) — с итерационными возвратами на предыдущие этапы после выполнения очередного этапа.

Построение комплексных ИС подразумевает согласование проектных решений, получаемых при реализации отдельных задач. Подход к проектированию «снизу вверх» предполагает необходимость таких итерационных возвратов, когда проектные решения по отдельным задачам объединяются и общие системные решения, и при этом возникает потребность в пересмотре ранее сформулированных требований.

Недостаток: Вследствие большого числа итераций возникают рассогласования и несоответствия в выполненных проектных решениях и документации.

2.1 Жизненный цикл ИС

3. **Спиральная модель** (80- 90-е годы) прототипная модель, предполагающая постепенное расширение ПО.

Спиральная модель определяет четыре действия, представляемые четырьмя квадрантами спирали:

- ◆ планирование - определение целей, вариантов и ограничений;
- ◆ анализ риска - анализ вариантов и распознавание (выбор) риска;
- ◆ конструирование — разработка продукта следующего уровня;
- ◆ оценивание - оценка заказчиком текущих результатов конструирования.

С каждой итерацией по спирали (продвижением от центра к периферии) строятся все более полные версии ПО.

Спиральная модель жизненного цикла ИС реально отображает разработку программного обеспечения; позволяет явно учитывать риск на каждом витке эволюции разработки; включает шаг системного подхода в итерационную структуру разработки; использует моделирование для уменьшения риска и совершенствования программного изделия.

2.1 Жизненный цикл ИС

Недостатками спиральной модели являются:

- ◆ новизна (отсутствует достаточная статистика эффективности модели):
- ◆ повышенные требования к заказчику;
- ◆ трудности контроля и управления временем разработки.

В основе спиральной модели жизненного цикла лежит применение **прототипной технологии** или **RAD-технологии** (**rapid application development — технологии быстрой разработки приложений**).

Основная идея этой технологии заключается в том, что ИС разрабатывается путем расширения программных прототипов, повторяя путь от детализации требований к детализации программного кода.

При прототипной технологии сокращается число итераций, возникает меньше ошибок и несоответствий, которые необходимо исправлять на последующих итерациях, а само проектирование ИС осуществляется более быстрыми темпами, упрощается создание проектной документации. Для более точного соответствия проектной документации разработанной ИС все большее значение придается использованию CASE-технологий.

2.1 Жизненный цикл ИС

RAD-технология обеспечивает экстремально короткий цикл разработки ИС. При полностью определенных требованиях и ограниченной проектной области RAD-технология позволяет создать полностью функциональную систему за очень короткое время (60-90 дней).

Выделяют следующие этапы разработки ИС с использованием RAD-технологии:

- 1) бизнес-моделирование. Моделируется информационный поток между бизнес-функциями. Определяются ответы на вопросы: Какая информация руководит бизнес-процессом? Какая информация генерируется? Кто генерирует ее? Где информация применяется? Кто обрабатывает информацию?
- 2) моделирование данных. Информационный поток отображается в набор объектов данных, которые требуются для поддержки деятельности организации. Определяются характеристики (свойства, атрибуты) каждого объекта, отношения между объектами;

2.1 Жизненный цикл ИС

- 3) моделирование обработки. Определяются преобразования объектов данных, обеспечивающие реализацию бизнес-функций. Создаются описания обработки для добавления, модификации, удаления или нахождения (исправления) объектов данных;
- 4) генерация приложения. Предполагается использование методов, ориентированных на языки программирования 4-го поколения. Вместо создания ПО с помощью языков программирования 3-го поколения, RAD-процесс работает с повторно используемыми программными компонентами или создает повторно используемые компоненты. Для обеспечения конструирования используются утилиты автоматизации (CASE-средства);
- 5) тестирование и объединение. Поскольку применяются повторно используемые компоненты, многие программные элементы уже протестированы, что сокращает время тестирования (хотя все новые элементы должны быть протестированы).

2.1 Жизненный цикл ИС

Применение RAD имеет и свои недостатки, и ограничения:

- ◆ большие проекты в RAD требуют существенных людских ресурсов (необходимо создать достаточное количество групп);
- ◆ RAD применима только для приложений, которые можно разделять на отдельные модули и в которых производительность не является критической величиной;
- ◆ RAD неприменима в условиях **ВЫСОКИХ** технических рисков.

2.2 Основы применения инструментальных средств ИТ

Переход на промышленную технологию производства программ, стремление к сокращению сроков, трудовых и материальных затрат на производство и эксплуатацию программ, обеспечение гарантированного уровня качества ИС обусловили бурно развивающееся направление — программотехнику, связанное с технологией создания программных продуктов.

Инструментарий технологии программирования — программные продукты поддержки технологии программирования. В рамках этих направлений сформировались следующие группы:

- ◆ средства для создания приложений;
- ◆ CASE-технологии (Computer-Aided Software Engineering), предназначенные для автоматизации процессом разработки и реализации информационных систем.

2.2 Основы применения инструментальных средств ИТ

Средства для создания приложений включают языки и системы программирования, а также инструментальную среду разработчика.

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

Средства для создания приложений - совокупность языков и систем программирования, а также различные программные комплексы для отладки и поддержки разрабатываемых программных продуктов.

Языки программирования разделяют на следующие классы (но синтаксису конструкций языка):

- ◆ машинные языки - языки программирования, воспринимаемые аппаратной частью компьютера (машинные коды);
- ◆ машинно-ориентированные языки - языки программирования, которые отражают структуру конкретного типа компьютера (ассемблеры);

2.2 Основы применения инструментальных средств ИТ

- ◆ алгоритмические языки, не зависящие от архитектуры компьютера языки программирования для отражения структуры алгоритма (Паскаль, Фортран, Бейсик и др.);
- ◆ процедурно-ориентированные языки - языки программирования, где имеется возможность описания программы как совокупности процедур, подпрограмм:
- ◆ проблемно-ориентированные языки - языки программирования, предназначенные для решения задач определенного класса (ЛИСП, РПГ, Симула и др.);
- ◆ интегрированные системы программирования. Другой классификацией языков программирования является их деление на языки, предназначенные для реализации основ структурного программирования, и объектно-ориентированные языки, поддерживающие понятие объектов, их свойств и методов обработки.

Программа, написанная на языке программирования, проходит этап трансляции, когда происходит преобразование исходного кода программы и объектный код, который далее пригоден к обработке редактором связей.

Редактор связей - специальная подпрограмма, обеспечивающая построение загрузочного модуля, пригодного к выполнению.

2.2 Основы применения инструментальных средств ИТ

Трансляция может выполняться с использованием средств компиляторов или интерпретаторов.

Компиляторы транслируют всю программу, но без ее выполнения.

Интерпретаторы, в отличие от компиляторов, выполняют пооператорную обработку и выполнение программы.

Необходимым средством для профессионального разработчика являются специальные программы, предназначенные для трассировки и анализа выполнения других программ, отладчики.

2.2 Основы применения инструментальных средств ИТ

Современная система программирования состоит из следующих компонентов:

- ◆ компилятор;
- ◆ интегрированная среда разработчика программ;
- ◆ отладчик;
- ◆ средства оптимизации кода программ;
- ◆ набор библиотек (возможно, с исходными текстами программ);
- ◆ редактор связей;
- ◆ сервисные средства (утилиты) для работы с библиотеками, текстовыми и двоичными файлами;
- ◆ справочные системы;
- ◆ документатор исходного кода программы;
- ◆ систему поддержки и управления проектом программного комплекса.

CASE-технология — программный комплекс, автоматизирующий весь технологический процесс анализа, проектирования, разработки и сопровождения сложных программных систем. Основное преимущество CASE-технологии — возможность коллективной работы над проектом за счет поддержки работы разработчиков в локальной сети, экспорта-импорта любых фрагментов проекта организационно-го управления проектом.

2.3 Создание автоматизированных информационных систем

Создание автоматизированных информационных систем регламентируется комплексом стандартов и руководящих документов. Можно выделить следующие стадии и этапы создания АИС (табл.).

Таблица. Стадии и этапы создания АИС

Стадии	Этапы работ
1. Формирование требований к ИС	1.1. Обследование объекта и обоснование необходимости создания ИС. 1.2. Формирование требований пользователя к ИС. 1.3. Оформление отчета о выполненной работе и заявки на разработку ИС (тактико-технического задания)
2. Разработка концепции ИС	2.1. Изучение объекта. 2.2. Проведение необходимых научно исследовательских работ. 2.3. Разработка вариантов концепции ИС и выбор варианта концепции ИС, удовлетворяющего требованиям пользователя. 2.4. Оформление отчета о выполненной работе
3. Техническое задание	3.1. Разработка и утверждение технического задания на создание ИС (см. ниже)
4. Эскизный проект	4.1. Разработка предварительных проектных решений по системе и ее частям. (см. ниже) 4.2. Разработка документации на ИС и ее части
5. Технический проект	5.1. Разработка проектных решений по системе и ее частям. (см. ниже) 5.2. Разработка документации на ИС и ее части.

2.3 Создание автоматизированных информационных систем

	<p>5.3. Разработка и оформление документации на поставку изделий для комплектования ИС и (или) технических требований (технических заданий) на их разработку.</p> <p>5.4. Разработка заданий на проектирование в смежных частях проекта объекта автоматизации</p>
6. Рабочая документация	<p>6.1. Разработка рабочей документации на систему и ее части.</p> <p>6.2. Разработка или адаптация программ (см. ниже)</p>
7. Ввод в действие	<p>7.1. Подготовка объекта автоматизации к вводу ИС в действие.</p> <p>7.2. Подготовка персонала.</p> <p>7.3. Комплектация ИС поставляемыми изделиями (программными и техническими средствами, программно-техническими комплексами, информационными изделиями).</p> <p>7.4. Строительно-монтажные работы.</p> <p>7.5. Пусконаладочные работы.</p> <p>7.6. Проведение предварительных испытаний.</p> <p>7.7. Проведение опытной эксплуатации.</p> <p>7.8. Проведение приемочных испытаний</p>

2.3 Создание автоматизированных информационных систем

Одним из центральных элементов всего процесса создания АИС является разработка **технического задания**, структура которого, согласно ГОСТ 34.602-89, содержит следующие разделы:

1. общие сведения;
2. назначение и цели создания (развития) системы;
3. характеристика объектов автоматизации;
4. требования к системе:
 - 4.1. требования к системе в целом - отражают концептуальные параметры и характеристики создаваемой системы, среди которых указываются требования к структуре и функционированию системы, к надежности и безопасности, к численности и квалификации персонала и т. д.

2.3 Создание автоматизированных информационных систем

4.2. требования к функциям (задачам), выполняемым системой - содержат перечень функций, задач или их комплексов; временной регламент каждой функции, задачи или комплекса задач; требования к качеству реализации каждой функции; к форме представления выходной информации; характеристики необходимой точности и времени выполнения, требования одновременности выполнения группы функций; достоверности выдачи результатов.

4.3. требования к видам обеспечения:

- к составу, структуре и способам организации данных в системе (информационно-логическая схема);
- к информационному обмену между компонентами системы;
- к информационной совместимости со смежными системами;
- по использованию общероссийских и других классификаторов, унифицированных документов;
- по применению систем управления базами данных;
- к структуре процесса сбора, обработки, передачи данных в системе и представлению данных;
- к защите данных от разрушений при авариях и сбоях в электропитании системы;
- к контролю, хранению, обновлению и восстановлению данных;
- к процедуре придания юридической силы документам, продуцируемым техническими средствами ПС.

2.3 Создание автоматизированных информационных систем

5. состав и содержание работ по созданию системы;
6. порядок контроля и приемки системы;
7. требования к составу и содержанию работ ПО подготовке объекта автоматизации к вводу системы в действие;
8. требования к документированию;
9. источники разработки.

На основе установленных в техническом задании основных требований и технических решений на последующих этапах конкретизируются и непосредственно разрабатываются компоненты и элементы системы.

В частности, на этапе 4.1 «Разработка предварительных проектных решений по системе и ее частям» определяются:

- ◆ функции ИС;
- ◆ функции подсистем;
- ◆ концепция информационной базы и ее укрупненная структура;
- ◆ функции системы управления базой данных;
- ◆ состав вычислительной системы;
- ◆ функции и параметры основных программных средств.

2.3 Создание автоматизированных информационных систем

На этапе 5.1 «Разработка проектных решений по системе и ее частям» осуществляется разработка общих решений по системе и ее частям:

- ◆ по функционально-алгоритмической структуре системы;
- ◆ по функциям персонала и организационной структуре;
- ◆ по структуре технических средств;
- ◆ по алгоритмам решения задач и применяемым языкам;
- ◆ по организации и ведению информационной базы (структура базы данных);
- ◆ по системе классификации и кодирования информации (словарно-классификационная база);
- ◆ по программному обеспечению.

Разработка и документация программного обеспечения в процессе создания или комплектования автоматизированных систем (п. 6.2) регламентируются комплексом стандартов, объединенных в группу «Единая система программной документации (ЕСПД)».