Тема реферата: Проектирование Информационных Систем

Введение

Основные особенности современных проектов ИС. Этапы создания ИС: формирование требований, концептуальное проектирование, спецификация приложений, разработка моделей, интеграция и тестирование информационной системы. Методы программной инженерии в проектировании ИС. Проектирование (от латинского projectus, что означает “брошенный вперед”) – это процесс составления описания, необходимого для создания в заданных условиях еще не существующего объекта по первичному описанию этого объекта путем его детализации, дополнения, расчетов и оптимизации. Описание объекта может быть задано по-разному: в виде текста, алгоритма, программы, чертежа, таблицы или, что чаще всего, комбинировано в традиционно бумажном или электронном виде.

Таким образом, главной особенностью проектирования является работа с еще не существующим объектом. В этом отличие проектирования от моделирования, где объект не может не существовать.

1. Характеристики современных проектов. Определение проектирования. Объект проектирования

Проектирование можно рассматривать с одной стороны, как заключительную фазу исследований, а с другой как начальную фазу производства.

Проектирование ИС охватывает три основные области:

\* проектирование объектов данных, которые будут реализованы в базе данных;

\* проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным;

\* учет конкретной среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры (файл-сервер или клиент-сервер), параллельной обработки, распределенной обработки данных и т.п.

Проектирование информационных систем всегда начинается с определения цели проекта. В общем виде цель проекта можно определить, как решение ряда взаимосвязанных задач, включающих в себя обеспечение на момент запуска системы и в течение всего времени ее эксплуатации:

\* требуемой функциональности системы и уровня ее адаптивности к изменяющимся условиям функционирования;

\* требуемой пропускной способности системы;

\* требуемого времени реакции системы на запрос;

\* безотказной работы системы;

\* необходимого уровня безопасности;

\* простоты эксплуатации и поддержки системы.

2. Программные средства поддержки жизненного цикла ПО

Современные методологии и реализующие их технологии поставляются в электронном виде вместе с CASE-средствами и включают библиотеки процессов, шаблонов, методов, моделей и других компонент, предназначенных для построения ПО того класса систем, на который ориентирована методология. Электронные методологии включают также средства, которые должны обеспечивать их адаптацию для конкретных пользователей и развитие методологии по результатам выполнения конкретных проектов.

Процесс адаптации заключается в удалении ненужных процессов, действий ЖЦ и других компонентов методологии, в изменении неподходящих или в добавлении собственных процессов и действий, а также методов, моделей, стандартов и руководств. Настройка методологии может осуществляться также по следующим аспектам: этапы и операции ЖЦ, участники проекта, используемые модели ЖЦ, поддерживаемые концепции и др.

Электронные методологии и технологии (и поддерживающие их CASE-средства) составляют ядро комплекса согласованных инструментальных средств среды разработки ИС.

Методология DATARUN опирается на две модели или на два представления:

\* модель организации;

\* модель ИС.

Методология DATARUN базируется на системном подходе к описанию деятельности организации. Построение моделей начинается с описания процессов, из которых затем извлекаются первичные данные, которые описывают продукты или услуги организации, выполняемые операции (транзакции) и потребляемые ресурсы. Они становятся основой для проектирования архитектуры ИС. Архитектура ИС будет более стабильной, если она основана на первичных данных, тесно связанных с основными деловыми операциями, определяющими природу бизнеса, а не на традиционной функциональной модели.

3. Требования к информационным системам (ИС). Техническое задание. Разработка и утверждение технического задания на создание ИС

Техническое задание - это документ, определяющий цели, требования и основные исходные данные, необходимые для разработки автоматизированной системы управления.

При разработке технического задания необходимо решить следующие задачи:

* установить общую цель создания ИС, определить состав подсистем и функциональных задач;
* разработать и обосновать требования, предъявляемые к подсистемам;
* разработать и обосновать требования, предъявляемые к информационной базе, математическому и программному обеспечению, комплексу технических средств
* (включая средства связи и передачи данных);
* установить общие требования к проектируемой системе;
* определить перечень задач создания системы и исполнителей;
* определить этапы создания системы и сроки их выполнения;
* провести предварительный расчет затрат на создание системы и определить уровень экономической эффективности ее внедрения.

4. Определение и назначение CASE-технологии. Качества, необходимые предприятию для успешного внедрения CASE-средств

CASE-технология представляет собой методологию проектирования ИС, а также набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать эту модель на всех этапах разработки и сопровождения ИС и разрабатывать приложения в соответствии с информационными потребностями пользователей. Большинство существующих CASE-средств основано на методологиях структурного (в основном) или объектно-ориентированного анализа и проектирования, использующих спецификации в виде диаграмм или текстов для описания внешних требований, связей между моделями системы, динамики поведения системы и архитектуры программных средств

CASE-средства позволяют создавать не только продукт, практически готовый к применению, но и обеспечить “правильный” процесс его разработки. Основная цель технологии – отделить проектирование программного обеспечения от его кодирования, сборки, тестирования и максимально “скрыть” от будущих пользователей все детали разработки и функционирования ПО. При этом значительно повышается эффективность работы проектировщика: сокращается время разработки, уменьшается число программных ошибок, программные модули можно использовать при следующих разработках.

Большинство CASE-средств основано на парадигме “методология / метод / нотация / структура / средство”.Методология задает руководящие указания для оценки и выбора проекта разработки ПО, этапы и последовательность работ, правила применения тех или иных методов.

Метод – систематическая процедура или технология генерации описаний компонент ПО (например, описание потоков и структур данных).

Нотации предназначены для описания системы в целом, ее элементов, таких как графы, диаграммы, таблица, блок-схемы, алгоритмы, формальные языки и языки программирования.

Структуры являются средством для реализации структурного анализа и построения структуры конкретной системы.

Средства – технологические и программные инструменты для поддержки и усиления методов.

CASE-технологии обладают следующими основными достоинствами, которые позволяют широко использовать их при разработке информационных систем:

* ускоряют процесс коллективного проектирования и разработки;
* позволяют за короткий срок создать прототип заказанной системы с заданными свойствами;
* освобождают разработчика от рутинной работы, оставляя время для творчества;
* обеспечивают эффективность и качество разрабатываемого ПО за счет автоматизации контроля всего процесса разработки;
* поддерживают сопровождение и развитие системы на высоком уровне.

Для успешного внедрения CASE-средств организация должна обладать нижеследующими качествами.

Культура. Готовность к внедрению новых процессов и взаимоотношений между разработчиками и пользователями, ИТ/ИС-управленцами и пользователями.

Управление. Четкое руководство и организованность по отношению к наиболее важным этапам и процессам внедрения.

Технология. Понимание ограниченности существующих возможностей и способность принять новую технологию.

Если организация не обладает хотя бы одним из перечисленных качеств, то внедрение CASE-средств может закончиться неудачей независимо от степени тщательности

следования различным рекомендациям по внедрению.

В качестве примеров популярных CASE-средств укажем программные средства компании Computer Associates, IBM-Rational Software и Oracle:

* BPwin – моделирование бизнес-процессов;
* ERwin – моделирование баз данных и хранилищ данных;
* ERwin Examiner – проверка структуры СУБД и моделей, созданных в Erwin;
* ModelMart – среда для командной работы проектировщиков;
* Paradigm Plus – моделирование приложений и генерация объектного кода;
* Rational Rose – моделирование бизнес-процессов и компонентов приложений;
* Rational Suite AnalystStudio – пакет для аналитиков данных;
* Oracle Designer (входит в Oracle9i Developer Suite) – высокофункциональное средство проектирования программных систем и баз данных, реализующее технологию CASE и собственную методологию Oracle – CDM. Позволяет команде разработчиков полностью провести проект, начиная от анализа бизнес-процессов через моделирование к генерации кода и получению прототипа, а в дальнейшем и окончательного продукта. Сложное CASE-средство, его имеет смысл использовать при ориентации на линейку продуктов Oracle.

5. Управление проектом. Жизненный цикл проекта.

ЖЦ ПО ИС – это непрерывный процесс, начинающийся с момента принятия решения о необходимости создания ИС и заканчивающийся в момент полного ее изъятия из эксплуатации

Основным нормативным документом, регламентирующим ЖЦ ИС, является международный стандарт ISO/IEC 12207.

Структура ЖЦ ПО по стандарту ISO/IEC 12207 базируется на трех группах процессов:

1. Основные процессы ЖЦ

Среди основных процессов ЖЦ наибольшую важность имеют три: разработка, эксплуатация и сопровождение.

Разработка ИС включает в себя все работы по созданию информационного ПО и его компонентов в соответствии с заданными требованиями, включая оформление проектной и эксплуатационной документации, подготовку материалов, необходимых для тестирования разработанных программных продуктов, и разработку материалов, необходимых для организации обучения персонала и т.д.

Эксплуатация включает в себя работы по внедрению компонентов ИС в эксплуатацию, в том числе конфигурирование базы данных и рабочих мест пользователей, обеспечение пользователей эксплуатационной документацией, проведение обучения персонала и т.д., и непосредственно эксплуатацию, в том числе локализацию проблем и устранение причин их возникновения, модификацию ПО в рамках установленного регламента, подготовку предложений по совершенствованию, развитию и модернизации системы.

Сопровождение включает в себя техническую поддержку ИС. Основными предварительными действиями при подготовке к организации технического обслуживания ИС являются: выделение наиболее ответственных узлов системы и определение для них критичности простоя; определение задач технического обслуживания и их разделение на внутренние, решаемые силами обслуживающего подразделения, и внешние, решаемые специализированными сервисными организациями; подготовка плана организации технического обслуживания, в котором необходимо определить этапы исполняемых действий, сроки их исполнения, затраты на этапах, ответственность исполнителей.

2.Вспомогательные процессы ЖЦ

Управление конфигурацией является одним из вспомогательных процессов, поддерживающих основные процессы жизненного цикла ИС, прежде всего процессы разработки и сопровождения. Управление конфигурацией позволяет организовать, систематически учитывать и контролировать внесение изменений в различные компоненты ИС на всех стадиях ее ЖЦ.

3.Организационные процессы ЖЦ

Управление проектом связано с вопросами планирования и организации работ, создания коллективов разработчиков и контроля за сроками и качеством выполняемых работ. Техническое и организационное обеспечение проекта включает выбор методов и инструментальных средств для реализации проекта, определение методов описания промежуточных состояний разработки, разработку методов и средств испытаний созданного ПО, обучение персонала и т.п.

Проверка позволяет оценить соответствие параметров разработки с исходными требованиями.

Стадии (этапы) ЖЦ

1. Планирование и анализ требований (предпроектная стадия) – системный анализ. Исследование и анализ существующей ИС, определение требований к создаваемой ИС, оформление технико-экономического обоснования (ТЭО) и технического задания (ТЗ) на разработку ИС.

2. Проектирование (техническое проектирование, логическое проектирование). Разработка в соответствии со сформулированными требованиями состава автоматизируемых функций (функциональная архитектура) и состава обеспечивающих подсистем (системная архитектура), оформление технического проекта ИС.

1. Реализация (рабочее проектирование, физическое проектирование, программирование). Разработка и настройка программ, наполнение базы данных, создание рабочих инструкций для персонала, оформление рабочего проекта.

2. Внедрение (тестирование, опытная эксплуатация). Комплексная отладка подсистем, обучение персонала, поэтапное внедрение ИС в Эксплуатацию по подразделениям экономического объекта, оформление акта о приемосдаточных испытаниях ИС.

3. Эксплуатация ИС (сопровождение, модернизация). Сбор рекламаций и статистики о функционировании ИС, исправление ошибок и недоработок, оформление требований к модернизации ИС и ее выполнение (повторение стадий 2-5).

6. Классификация CASE-средств по уровням.

Классификация по уровням связана с областью действия CASE в пределах жизненного цикла ПО. Однако четкие критерии определения границ между уровнями не установлены, поэтому данная классификация имеет, вообще говоря, качественный характер.

Верхние (Upper) CASE часто называют средствами компьютерного планирования. Они призваны повышать эффективность деятельности руководителей фирмы и проекта путем сокращения затрат на определение политики фирмы и на создание общего плана проекта. Этот план включает цели и стратегии их достижения, основные действия в свете целей и задач фирмы, установление стандартов на различные виды взаимосвязей и т.д. Использование верхних CASE позволяет построить модель предметной области, отражающую всю существующую специфику. Она направлена на понимание общего и частного механизмов функционирования, имеющихся возможностей, ресурсов, целей проекта в соответствии с назначением фирмы. Эти средства позволяют проводить анализ различных сценариев (в том числе наилучших и наихудших), накапливая информацию для принятия оптимальных решений.

Средние (Middle) CASE считаются средствами поддержки этапов анализа требований и проектирования спецификаций и структуры ПО. Их использование существенно сокращает цикл разработки проекта; при этом важную роль играет возможность накопления и хранения знаний, обычно имеющихся только в голове разработчика-аналитика, что позволит использовать накопленные решения при создании других проектов. Основная выгода от использования среднего CASE состоит в значительном облегчении проектирования систем, проектирование превращается в итеративный процесс, включающий следующие действия:

* пользователь обсуждает с аналитиком требования к проектируемой системе;
* аналитик документирует эти требования, используя диаграммы и словари входных данных;
* пользователь проверяет эти диаграммы и словари, при необходимости модифицируя их;
* аналитик отвечает на эти модификации, изменяя соответствующие спецификации.

Кроме того, средние CASE обеспечивают возможности быстрого документирования требований и быстрого прототипирования.

Нижние (Lower) CASE являются средствами разработки ПО (при этом может использоваться до 30% спецификаций, созданных средствами среднего CASE). Они содержат системные словари и графические средства, исключающие необходимость разработки физических спецификаций. Имеются системные спецификации, которые непосредственно переводятся в программные коды разрабатываемой системы (при этом автоматически генерируется до 80-90% кодов). На эти средства возложены также функции тестирования, управления конфигурацией, формирования документации. Главными преимуществами нижних CASE являются: значительное уменьшение времени на разработку, облегчение модификаций, поддержка возможностей прототипирования (совместно со средними CASE).

7. Типы ИС. Особенности внедрения ИС.

Информационные системы можно классифицировать по целому ряду различных признаков

По типу хранимых данных ИС делятся на фактографические и документальные. Фактографические системы предназначены для хранения и обработки структурированных данных в виде чисел и текстов. В документальных системах информация представлена в виде документов, состоящих из наименований, описаний, рефератов и текстов.

Основываясь на степени автоматизации информационных процессов в системе управления фирмой, информационные системы делятся на ручные, автоматические и автоматизированные.

Ручные ИС характеризуются отсутствием современных технических средств переработки информации и выполнением всех операций человеком.

В автоматических ИС все операции по переработке информации выполняются без участия человека.

Автоматизированные ИС предполагают участие в процессе обработки информации и человека, и технических средств, причем главная роль в выполнении рутинных операций обработки данных отводится компьютеру.

В зависимости от характера обработки данных ИС делятся на информационно-поисковые и информационно-решающие

Информационно-поисковые системы производят ввод, систематизацию, хранение, выдачу информации по запросу пользователя без сложных преобразований данных.

Информационно-решающие системы осуществляют, кроме того, операции переработки информации по определенному алгоритму. По характеру использования выходной информации такие системы принято делить на управляющие и советующие принимаемые человеком решения.

Советующие ИС вырабатывают информацию, которая принимается человеком к сведению и учитывается при формировании управленческих решений, а не инициирует конкретные действия.

В зависимости от сферы применения различают следующие классы ИС.

Информационные системы организационного управления – предназначены для автоматизации функций управленческого персонала как промышленных предприятий, так и непромышленных объектов (гостиниц, банков, магазинов и пр.).

ИС управления технологическими процессами (ТП) – служат для автоматизации функций производственного персонала по контролю и управлению производственными операциями.

ИС автоматизированного проектирования (САПР) – предназначены для автоматизации функций инженеров-проектировщиков, конструкторов, архитекторов, дизайнеров при создании новой техники или технологии.

Интегрированные (корпоративные) ИС – используются для автоматизации всех функций фирмы и охватывают весь цикл работ от планирования деятельности до сбыта продукции. Они включают в себя ряд модулей (подсистем), работающих в едином информационном пространстве и выполняющих функции поддержки соответствующих направлений деятельности. Типовые задачи, решаемые модулями корпоративной системы, приведены в таблице 1.

8. Компоненты интегрированного CASE-пакета.

Интегрированный CASE-пакет содержит четыре основные компоненты:

1) Средства централизованного хранения всей информации о проектируемом ПО в течении всего ЖЦ (репозитарий) zвляются основой CASE-пакета. Соответствущая БД должна иметь возможность поддерживать большую систему описаний и характеристик и предусматривать надежные меры по защите от ошибок и потерь информации.

Репозитарий должен обеспечивать:

* –инкрементный режим при вводе описаний объектов;
* –распространение действия нового или скорректированного описания на информационное пространство всего проекта;
* –синхронизацию поступления информации от различных пользователей;
* –хранение версий проекта и его отдельных компонент;
* –сборку любой запрошенной версии;
* –контроль информации на корректность, полноту и состоятельность.

2) Средства ввода предназначены для ввода данных в репозитарий, а также для организации взаимодействия с CASE-пакетом. Эти средства должны поддерживать различные методологии и использоватьсяна всем ЖЦ разными категориями разработчиков: аналитиками, проектировщиками, инженерами, администраторами и т. д.

3) Средства анализа, проектирования и разработки предназначены для того, чтобы обеспечить планирование и анализ различных описаний, а также их преобразования в процессе разработки.

4) Средства вывода служат для документирования, управления проектами кодовой генерации.

Все перечисленные компоненты в совокупности должны:

* –поддерживать графические модели;
* –контролировать ошибки;
* –организовывать и поддерживать репозитарий;
* –поддерживать процесс проектирования и разработки.

9. Определение автоматизированной ИС. Методы построения АИС.

Автоматизированная информационная система (АИС) — совокупность программно-аппаратных средств, предназначенных для автоматизации деятельности, связанной с хранением, передачей и обработкой информации.

В состав АИС входят:

* -информационные ресурсы, представленные в виде БД(БЗ), хранящих данные об объектах, связь между которыми задается определенными правилами
* -формальная логико-математическая система, реализованная в виде программных модулей (ввод, обработка, вывод)
* -интерфейс
* -персонал
* -комплекс технических средств

Три метода разработки АИС: оригинальный, типовой, автоматизированный.

Метод оригинального проектирования охватывает все виды работ для различных объектов, выполняемых по специальным проектам. Методики на всех этапах работ создаются для конкретного объекта по необходимости. Недостатки: высокая трудоемкость, большие сроки проектирования, плохие модернизируемость и сопровождаемость.

Метод типового проектирования предполагает разбиение системы на отдельные модули (элементы, подсистемы, объекты) и разработку для каждого из них законченного проекта.

Метод автоматизированного проектированияпредполагает автоматизацию основных этапов создания АИС, начиная от выбора состава задач и заканчивая автоматическим получением проектной документации. Используют САПР, CASE-технологии.

10. Жизненный цикл (ЖЦ) ИС. Модели ЖЦ.

ЖЦ ПО ИС – это непрерывный процесс, начинающийся с момента принятия решения о необходимости создания ИС и заканчивающийся в момент полного ее изъятия из эксплуатации. Иначе ЖЦ ПО ИС можно представить как ряд событий, происходящих с системой в процессе ее создания и использования.

Модели жизненного цикла ПО ИС

Стандарт ISO/IEC 12207 не предлагает конкретную модель ЖЦ и методы разработки ИС (под моделью ЖЦ понимается структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач, выполняемых на протяжении ЖЦ.

Модель ЖЦ зависит от специфики ИС и специфики условий, в которых последняя создается и функционирует). Его регламенты являются общими для любых моделей ЖЦ, методологий и технологий разработки. Стандарт ISO/IEC 12207 лишь описывает структуры процессов ЖЦ ИС, но не конкретизирует в деталях, как реализовать или выполнить действия и задачи, включенные в эти процессы.

Среди известных моделей ЖЦ можно выделить следующие:

• Каскадная модель (до 70-х гг.) предусматривает последовательное выполнение всех этапов проекта в строго фиксированном порядке. Переход на следующий этап означает полное завершение работ на предыдущем этапе.

• Итерационная модель (поэтапная модель с промежуточным контролем) (70-е – 80-е гг.). Разработка ИС ведется итерациями с циклами обратной связи между этапами. Межэтапные корректировки позволяют учитывать реально существующее взаимовлияние результатов разработки на различных этапах; время жизни каждого из этапов растягивается на весь период разработки.

• Спиральная модель (80-е – 90-е гг.). На каждом витке спирали выполняется создание очередной версии продукта, уточняются требования проекта, определяется его качество и планируются работы следующего витка. Особое внимание уделяется начальным этапам разработки – анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством создания прототипов (макетирования).

Заключение

Самым важным этом является проектирование информационной системы, поскольку именно от этого этапа зависит ее существования то, с чего, собственно, должна начинаться её жизнь. Например, для того, чтобы спроектировать сеть, нужно понять, какие задачи она будет решать, какими будут основные потоки трафика, как физически будут расположены пользователи и ресурсы, нужно ли задание приоритетов видов трафика, как будут решаться вопросы защиты информации внутри сети, как сеть будет подключена к Интернет, как решить задачи управления правами доступа пользователей.

Необходимо понять процесс создания ИС, что позволит гарантировать достижение целей разработки и соблюдение различных ограничений, например, бюджетных, временных и пр., а также привело к широкому использованию в этой сфере методов и средств программной инженерии: структурного анализа, объектно-ориентированного моделирования, CASE-систем.В работе рассмотрены цели создания ИС, задачи формирования требований к ИС, главная цель проектирования.