**ЛК1**

Жизненный цикл ПО

Жизненный цикл ПО определяется как период времени, который начинается с момента принятия решения о необходимости создания ПО и заканчивается моментом полного изъятия из эксплуатации. Выделяют следующие этапы жизненного цикла (ЖЦ) ПО:

1. Формирование требований к системе – планирование
2. Проектирование - документация
3. Реализация – написание кода
4. Тестирование
5. Ввод в действие – сдача
6. Эксплуатация и сопровождение

Для каждого из этого этапа выделяются следующие моменты:

1. Состав и последовательность выполняемых работ
2. Получаемый результат
3. Методы и средства, необходимые для выполнения работ
4. Роль и ответственность участников

На каждом этапе создаются специфичные модели. Формирование модели происходит рабочими группами команды и проекта.

Что мы подразумеваем, когда говорим о составе разработчиков:

1. ПМ – руководитель проекта
2. Девелоперы – бэк, фронт
3. Проектировщики – разраб архитектуры, спец по анализу предметной области, спец по анализу человеческого фактора, программист пользовательского интерфейса

Что подразумевается под девелоперами – прогеры, тестировщики, менеджеры по маркетингу, технические писатели

Есть ряд стандартов ПО:

1. Гост 34.601-90
2. CDM – методика Oracle
3. MSF – Microsoft solution framework
4. XP

Стандарт ISO-207, почитать про него

Модель жизненного цикла

Модель – структура, определяющая последовательность выполнения и взаимосвязи процессов, действий, задач, выполняемых на протяжении всего жизненного цикла

Модель ЖЦ зависит от специфики задач

Градации классической модели ЖЦ:

1. Каскадная модель
2. Поэтапная модель с промежуточным контролем
3. Спиральная модель

Как выглядит классическая модель – водопадная. Результаты первого блока влияют на процесс последующего блока.

Каскадная модель проектирования:

1. Однотактный подход – на каждом этапе формируется законченный набор документации. Прошел этап – дока
2. Прогнозирующая методология – выполняемая в логической последовательности этапов работ, которые позволяют планировать сроки всех работ и затраты. Это когда речь идет о времени и деньгах

Поэтапная модель с промежуточным контролем – похожа на классическую, только у нас есть момент промежуточного контроля. Возвращаемся к предыдущему. Последний – к первому

Итерационная модель – начинается с этапа планирования. После этого двигаемся по часовой стрелке, формируем требования. Переходим к проектированию и исполнению. После этого к тестированию и развертыванию. Оценка работы

Спиральная модель – каждый виток спирали соответствует созданию работоспособного фрагмента или версии системы. Преимущества - позволяет уточнить требования, цели и т.п. Минусы:

1. Высокие требования к заказчику. Изначально заказчик дает четкие требования к тому, что нужно разработать
2. Трудность контроля времени разработки и управления. Проблемы перехода на следующий виток

Существует огромное кол-во методологий разработки ПО:

1. Agile module – основные идеи: люди и взаимодействие важнее процессов и инструментов. Работающий продукт важнее исчерпывающей документации. Сотрудничество с заказчиком важнее согласования условий с заказчиком. Готовность к изменениям важнее первоначального плана
2. Scrum – набор принципов, на которых строится проект. Итерации в скраме – спринты.
3. Канбан – позволяет видеть конкретно все этапы

Подходы к выбору методологий:

Существует 4 принципа:

1. Большая по размеру методология нужна тогда, когда в проекте занято большее число разработчиков
2. Большая корректность методологий (видимая со стороны) нужна в тех случаях, когда скрытые ошибки в программе продукта могут повлечь за собой ущерб
3. Незначительное увеличение размеров или плотности методологий ведет к существенному увеличению стоимости продуктов
4. Наиболее эффективная форма коммуникации - непосредственное взаимодействие

**ЛК2**

Система – совокупность элементов, есть общая глобальная цель

Система – комплекс взаимодействующих элементов (основатель теории систем – Людвиг фон Берталанфи)

«Ограниченное, взаимно связанное множество, отражающее объективное существование конкретных отдельных взаимосвязанных совокупностей объектов и не содержащее объектов специфических ограничений, присущих частным системам»\

Свойства системы:

Ограниченность – система отделена от окружающей среды границами

Целостность – ее свойство целого принципиально не сводится к сумме свойств составляющих элементов

Структурность – поведение системы обусловлено не только особенностями отдельных элементов, сколько свойствами ее структуры

Взаимозависимость со средой – система формирует и проявляет свойства в процессе взаимодействия с средой

Иерархичность – соподчиненность элементов в системе

Множественность описаний – по причине сложности познания система требует множественности ее описаний

Отличительные особенности сложных систем – многочисленные и разные по типу связи между отдельно существующими элементами системы и наличие у системы функции (назначения), которой нет у составляющих ее частей

Организацией системы назовем связи (взаимодействия) между элементами сложной системы, которые могут характеризоваться определенным порядком, внутренними свойствами, направленностью на выполнение функций

Информационная система – взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели

Современные крупные проекты ИС характеризуются следующими особенностями:

Сложность описания

Наличие совокупности тесно взаимодействующих компонентов (подсистем)\

Ограниченная возможность использования типовых решений

Необходимость интеграции

Функционирование в неоднородной среде на нескольких аппаратных платформах

Разобщенность и разнородность отдельных групп разработчиков

Существенная временная протяженность проекта

Модель – это формальное описание особенностей системы, которые существенны для целей ее исследования

Пусть имеется некоторая контрольная система

Лишь в единичных случаях мы имеем возможность провести с самой этой системой все интересующие нас исследования

В большинстве же ситуаций по разным причинам (сложность, громоздкость, недоступность и т.д.) мы вынуждены рассматривать не саму систему, а ее модель

Различают три основных моделей:

Вербальные (словесные, описательные)

Натуральные (макетирование, физическое моделирование, масштабированные модели, модели части свойств и др)

Знаковые

Глобус - и то, и то

Знаковые модели:

Математическая модель – это описание протекания процессов (в том числе функционирования, движения), описание состояния, изменения системы на языке алгоритмических действий с математическими формулами и логических переходов

Уровни абстракции модели:

Модель без наполнения информацией до уровня соответствия единичной реальной системе называется общей (теоретической, абстрактной, системной)

Единую модель для всей сложной системы принято называть макромоделью

Формирование общего представления о системе можно разбить на стадии:

Стадии 1-5 описывают общее, цельное «изучение системы»

Стадии 6-9 образуют группу «формирование углубленных представлений о системе»

Стадия 10 - «Моделирование системы»

Стадии 11-13 образуют группу «сопровождение системы»

1. Выявление главных функций (свойств, целей, предназначений) системы. Формирование (выбор) основных предметных понятий, используемых в системе
2. Выявление основных частей (модулей) в системе и их функций. Понимание единства этих частей в рамках системы
3. Выявление основных процессов в системе, их роли, условий осуществления. Выявление стадийности, скачков, смен состояний и т.п. в функционировании системы; в системах с управлением – выделение основных управляющих факторов
4. Выявление основных этапом элементов «не-системы», с которыми связана изучаемая система. Выявление этих связей
5. Выявление неопределенностей и случайностей в ситуации их определяющего влияния на систему и выбор способа их математической формализации
6. Выявление разветвленной структуры, иерархии, формирование представлений о системе как о совокупности модулей, связанных входами-выходами
7. Выявление всех элементов и связей, важных для целей рассмотрения. Их отнесение к структуре иерархии в системе. Ранжирование элементов и связей по их значимости
8. Учет изменений и неопределенности в системе, входов и постоянных параметров
9. Исследование функций и процессов в системе с целью управления ими. Введение управления и процедур принятия решения. Рассмотрение управляющих воздействий как систем управления
10. Моделирование системы – описание системы, пригодное для предсказания ее поведения и вывода неочевидных свойств
11. Накопление опыта работы с системой и ее моделью. Уточнение сведений о системе, доводка и совершенствование моделей
12. Оценка предельных возможностей системы. Исследование отказов, выходов из строя, отклонений от нормы
13. Расширение функций (свойств) системы, изменение требований к ней, новый круг задач, новые условия работы. Включение системы элементов в систему более высокого уровня

Информационные потоки – неотъемлемая часть любой системы

Информационный поток – это совокупность циркулирующих в системе, между системой и внешней средой сообщений, необходимых для управления, анализа и контроля операций

Носитель информации – это любое материальное средство, фиксирующее информацию

Однородные информационные потоки характеризуются единым видом носителя, единой функциональной принадлежности

Информационные потоки (по периодичности):

Регулярные – соответствующие регламентированной во времени передаче данных

Оперативные – обеспечивающие связь по требованию

Виды информационных потоков характеризуются такими показателями:

Источник возникновения

Направление движения потока

Скорость передачи и приема

Интенсивность потока и др

Формирование ИС невозможно без исследования потоков в разрезе определенных показателей

Структура ИС

Подсистема – это часть системы, выделенная по какому-либо признаку

Общую структуру ИС можно рассматривать как совокупность подсистем независимо от сферы применения

Структура ИС = совокупность обеспечивающих подсистем

Типы обеспечивающих подсистем:

Информационное обеспечение

Техническое обеспечение

Математическое и программное обеспечение

Организационное обеспечение

Правовое обеспечение

Информационное обеспечение – совокупность единой системы классификации и кодирования информации, унифицированных систем документации, схем информационных потоков, циркулирующих в организации, а также методология построения баз данных

Назначение: своевременное формирование и выдача достоверной информации для принятия управленческих решений

Цели: устранение недостатков унифицированной системы документации

Техническое обеспечение – комплекс технических средств, предназначенных для работы ИС, а также соответствующая документация на эти средства и технологические процессы

Математическое и программное обеспечение – совокупность математических методов, моделей, алгоритмом и программ для реализации целей и задач ИС

Программное обеспечение:

Общественные продукты

Специальные программные продукты

Техническая документация

Организационное обеспечение – совокупность методов и средств, регламентирующих взаимодействие работников с техническими средствами и между собой в процессе разработки и эксплуатации информационное системы

Правовое обеспечение – совокупность правовых норм, определяюматещих создание, юридический статус и функционирование информационных систем, регламентирующих порядок получения, преобразования и использования информации

СОСТАВ ПРАВОВОГО ОБЕСПЕЧЕНИЯ:

Законы, указы, постановления гос. органов власти, приказы, инструкции и другие нормативные документы министерств, ведомств, организаций, местных органов власти

**ЛК 3**

Классификация ИС:

По типу хранимых данных

По степени автоматизации информационных процессов

По сфере применения

По характеру данных

По уровню управления

Фотка

По типу данных:

Под документальным – документы, структурированная информация и методы ее обработки

Фактографические – не структурируемая, необработанная инфа

По степени автоматизации:

Ручные – абсолютно полностью управляет человек

Автоматизированные – скрипты, но их должен кто-то запустить (50/50)

Автоматическое – полностью бесчеловечное управление

По сфере применения:

Интегрированные – часть какого-то глобального комплекса

Организационного управления – служат для манипуляции другими подсистемами, системы управления предприятиями (бухгалтерия, управление кадрами)

Управления ТП – информационные систему управления производственными процессами, система, управляющая сборками автомобилями

САПР – системы автоматического проектирования, которые позволяют что-то создавать

По характеру обработки:

Информационно-поисковые – это системы, которые где-то что-то взяли и куда-то что-то передали

Информационно-решающие – системы, которые вникают в суть данных, могут не отвечать за транспорт, они должны сказать некий вердикт.

Делятся на:

Управляющие – вмешиваются в процесс и могут корректировать

Советующие – простое уведомление

По уровню управление:

Стратегические – системы, которые предназначены для того, чтобы не было привязки ко времени, имеется доступ к большому кол-ву ресурсов у системы

Функциональные – обычные системы, которые решают узкую задачу, четко заданная задача – система ее решает

Операционные – интегрированные системы, обработка транзакций. Конкретного рода операции

Интегрированные (корпоративные) ИС – КИС:

Используются для автоматизации всех функций организации и охватывают весь цикл работ от планирования деятельности до сбыта продукции

Функциональное назначение модулей корпоративной ИС:

Фотка

Подсистемы могут включать:

Подсистему маркетинга

Производственные подсистемы

Финансовые и учетные подсистемы

Подсистему кадров (человеческие ресурсы)

Прочие подсистемы

ОСНОВЫ ПРОЕКТИРОВАНИЯ ИС

Основными документами, содержащими требования на разработку информационной системы, являются календарный план выполнения работ и техническое задание.

Первый из них регламентирует состав, сроки и финансирование работ

А второй – основные требования к системе

Техническое задание (ТЗ) является основным документом, определяющим требования и порядок создания (развития или модернизации) системы

В ТЗ указываются процессы контроля, контрольные точки, процессы приема-сдачи, описание документации, описание дальнейшего процесса модернизации. ТЗ очень важный документ для исполнителей

Вариант ТЗ:

Общие сведения о системе

Назначение и цели создания (развития) ИС

Характеристика объектов автоматизации

Требования к ИС в целом, к функциям и обеспечению

Состав и содержание работ по созданию системы

Порядок контроля и приемки системы

Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

Требования к документированию

Источники разработки

Проектирование – процесс перехода от одной модели в виде первичного описания системы (ТЗ) к ее описанию в виде набора стандартных документов, достаточных для создания систем (проектной документации)

Когда мы переходим к разработке проектной документации, есть несколько подходов к этому процессу. Все подходы использую определенную группу признаков

Основные принципы проектирования

1. Принцип декомпозиции («разделяй и властвуй») – любые сложные проблемы можно разбить на задачи
2. Принцип иерархического упорядочивания – есть некоторый уровень взгляда на систему, наследование
3. Принцип концептуальной общности – единый подход для всех этапов
4. Принцип абстрагирования – на каждом этапе системы нужно понимать, что важно
5. Принцип формализации – некое методические описание, все по полочкам, по шагам описано
6. Принцип унификации – стараться максимально использовать общепринятые формации
7. Принцип логической независимости – в данном аспекте он относится к синтезу информационной системы, мы продумываем бизнес логики без конкретной реализации
8. Принцип многомодельности – мы можем под разными ракурсами взглянуть на проблему
9. Принцип непротиворечивости (согласованности) – те же модели должны согласовываться между собой
10. Принцип информационной закрытости (инкапсуляции) – для некоторых моделей нужно скрывать внутреннюю сердцевину
11. Принцип полиморфизма – должны проектировать так, чтобы внешний вид нашего объекта можно было легко менять, не затрагивая бэк. Разделять сущность и интерфейс

Модель как проекция «системы»

При моделировании систем реального мира и нетривиальных программных систем выбор проекции неочевиден

Проекция (точка зрения) определяет, на какие вопросы может ответить модель

Визуальное представление моделей:

Блок-схема

Граф переходов

Конечный автомат

Схема формирования информационной модели

Фотка

Концептуальная модель – то, что написали в ТЗ. Из чего состоит система, что может

Логическая модель – все эти объекты – как взаимодействую между собой, что из себя представляют

Физическая модель – реализация

Классификация моделей ИС

По строгости описания:

Неформальное – приближенное, эскизы

Формальное – дотошное описание

По степени физической реализации (логической независимости):

Логические – концепты с точки зрения теоретического алгоритма

Физические – конкретные типы данных, структуры

По степени отображения динамики происходящих процессов:

Статические – блок-схемы

Динамические – конечные автоматы

По отображаемому аспекту:

Функциональные – основной аспект, должны показать, какие функции есть у системы

Информационные – показывают, какие информационные потоки внутри или снаружи системы

Поведенческие – чаще всего показывает, как можно пользоваться системой

Компонентные – структурная схема: структура файлов, классов, библиотек

Смешанные – и те, и иные модели

Нотация – установленные способы отображения элементов системы, т.е. графы, таблицы, блок-схемы, формальные и естественные языки

Парадигма – исходная концептуальная схема (модель) постановки проблемы и ее решения

Мониторинг – комплексная система наблюдения, контроля, оценки и прогноза явлений, процессов, объектов и т.п.

**ЛК 4-5**

Подходы к проектированию ИС:

- Структурное моделирование

- Объектно-ориентированное проектирование

Структурный подход:

Сущность подхода к разработке модели состоит в расчленении анализируемой системы на части («черные ящики» и их иерархической организации)

Структурным анализом принято называть метод исследования статических характеристик системы путем выделения в ней подсистем и элементов различного уровня иерархии, определения отношений и связей между ними

Отсутствие вывода – не анализ

Преимущество работы с «черными ящиками»: нет необходимости знать, как они работают – достаточно иметь информацию об их входах и выходах, а также функциях, которые они выполняют

Структурный анализ:

Сфера применения: проектирование производственно-экономических и инженерно-технических систем:

- анализ информационных потоков на предприятии;

- ре-инжиниринг бизнес-процессов;

- компьютеризация деятельности предприятия;

- разработка систем автоматизированного проектирования;

- разработка баз данных;

- разработка программных приложений, реализующих управление информационными потоками (например, системы электронного документооборота).

В моделировании бизнес-процессов структурный подход базируется на 3 основных положениях:

1. Разбиение исследуемого процесса на функциональные блоки – подпроцессы;
2. Возможность детализации любых процессов путем иерархической декомпозиции;
3. Использование для описания процесса графических нотаций с возможностью текстового разъясняющего дополнения

Схема применения структурного подхода:

- Разработка функциональной модели ->

-> Разработка информационной модели (какие структуры данных поступают на вход, какие возвращаем, как они изменяются внутри системы) ->

-> Разработка поведенческих моделей (модели могут быть с точки зрения использования системы, так и с точки зрения самой системы: разные алгоритмы) ->

-> Разработка моделей компонентов и развертывания (модели, которые строят будущую структуру моделей)

История развития структурного подхода

- SA – Structured Analysis (1960-e – середина 1970-х)

-- системы автоматизированного проектирования

-- структурный анализ при создании алгоритмических языков

- SADT – Structural Analysis and Design Technique (1974)

-- методология структурного проектирования

- Программа ICAM – Integrated Computer-Aided Manufacturing (конец 1970-х)

-- интегрированная компьютеризация производства США

-- начало разработки методологии IDEF (ICAM Definition)

SADT -> ICAM DEFinition -> Integrated DEFinition -> IDEF0

1993 – NIST [National Institute of Standards and Technology]

IDEF – сокращение от Integration Definition Metodology (Объединение Методологических Понятий).

Семейство совместно используемых методов для процесса моделирования

Семейство IDEF:

* IDEF0 (Function Modeling) – метод используется для построения функциональной модели, описывает бизнес-функции и контекст поведения
* IDEF1 (Information Modeling) – для поддержки функций производственной среды
* IDEF2 (Simulation Model Design) – для моделирования меняющихся во времени функций, поведений системы
* IDEF 3 (Process Description Capture) – метод описания процессов
* Process Flow Description (PFD) – описание процессов
* Object State Transition Description (OSTD) – описание состояний
* IDEF4 (Object-Oriented Design) – был создан для поддержки ОО проектирования
* IDEF5 (Ontology Description Capture) – описание онтологий
* IDEF6 (Design Rational Capture Method)
* IDEF7 (Information System Auditing)
* IDEF8 (User Interface Modeling)
* IDEF9 (Business Constraint Discovery)
* IDEF10 (Implementation Architecture Modeling)
* IDEF11 (Information Artifact Modeling
* IDEF12 (Organization Modeling)
* IDEF13 (Three Schema Mapping Design)
* IDEF14 (Network Design)

IDEF0

Это методология функционального моделирования и графическая аннотация, предназначенная для формализации и описания бизнес процессов информационной системы

Главная особенность – акцент на соподчиненность объектов

В IDEF0 рассматриваются только логические отношение между объектами, а также не учитывается временная шкала

Основной структурой IDEF0 является диаграмма. На ней (них) обозначается некоторая модель предметной области или её части

В IDEF0 реализованы три базовых принципа моделирования бизнес-процессов:

- принцип функциональной декомпозиции

- принцип ограничения сложности

- принцип контекста

В соответствии с принципом функциональной декомпозиции сложная бизнес-функция может быть представлена в виде совокупности составляющих ее более простых функций, которые сами в свою очередь могут быть подвергнуты декомпозиции

Согласно принципу ограничения сложности, количество функциональных боков на одной диаграмме должно быть не менее двух (за исключением контекстной диаграммы) и не более шести

Таким образом обеспечивается разборчивость и удобочитаемость диаграмм IDEF0. Практика показывает, что соблюдение этого принципа в большинстве случаев приводит к тому, что процессы, представленные в виде модели IDEF0, становятся лучше структурированы, более понятны и легче поддаются анализу

Принцип контекста состоит в том, что моделирование бизнес-процесса начинается с построения контекстной диаграммы

На этой диаграмме отображается только один блок – главная бизнес-функция моделируемой системы

При определении главной бизнес-функции необходимо всегда иметь в виду цель моделирования и точку зрения на модель

Основной структурной единицей IDEF0-модели является диаграмма, представляющая собой графическое описание модели предметной области или ее части

Главными компонентами на каждой диаграмме выступают блоки

Синтаксис IDEF0

Диаграммы содержат блоки, стрелки, правила

- Блоки представляют функции, определяемые как деятельность, процесс, операция, действе или преобразование

- Стрелки представляют данные или материальные объекты, связанные с функциями

- Правила определяют, как следует применять компоненты

Блок отображается в виде некоторого прямоугольника и отражает описание некоторой работы, процесса, функции, задачи, которые выполняются в некотором контексте в течение некоторого времени

Блоки содержат:

- имя функции – должен быть глагол или глагольный оборот

- номер в правом нижнем углу. Это идентификационный номер блока. Обязательно справа снизу. Нумерация сквозная по всей модели – т.е. блоки не только в рамках одной диаграммы уникальны, но и в рамках диаграммы

Требований к размерам блока нет. Блоки не должны выходить за рамки диаграммы. Должны помещать читаемый текст. Он должен быть с прямыми углами и со сплошными линиями

Стрелки – описывают потоки данных или материальные объекты, которые связанны с функцией

Стрелки формируется из одного или более отрезков прямых и наконечника на одном конце.

Важно! Стрелки не представляют поток или последовательность событий, как в традиционных блок-схемах потоков или процессов. Они лишь показывают, какие данные или материальные объекты должны поступить на вход функции для того, чтобы эта функция могла выполняться

Есть 4 вида стрелок, на самом деле 5:

1 – вход функции – слева в блок

2 – управление функцией (объекты или потоки, которые рассказывают, как функция должна работать) – сверху в блок

3 – механизмы (кто выполняет действия) – снизу в блок

4 – выход функции – справа от блока

5 – вызов или запрос – снизу от блока

Вход – всегда слева в сторону блока. Обозначает некоторый материал или информацию, который используется функцией и/или преобразуется некоторой функцией для получения результата. Обрабатываемая информация

Управление – это некоторые условия, правила, стандарты, концепции, стратегии, законы выполнения функции, которые влияют на ее работу. Поступает сверху. Например, функция проверки орфографии. Обрабатывающая информация

!!! У каждого блока должна быть хоть одна стрелка управления !!!

Механизмы – некоторые ресурсы, с помощью которых выполняется функция. В качестве таких ресурсов могут быть: денежные средства, персонал, инструментарий

Выход – симметрия входа: результат выполнения функции (некоторая информация или материал). В большей степени применялась к промышленности. Потенциал функции

!!! Выход всегда должен быть, хотя бы один !!!

Вызов – специализированная дуга, которая указывает на некоторую другую предметную область. С ее помощью можно обозначить связь с другой предметной областью. Или же проблемная область

Что из себя представляет стрелка: сплошная линия, иметь наконечник, может быть искривлена, но желательно под прямым углом. Желательно делать линейно. Край скругленный, но прямой. Могут быть слияния и разветвления. Все стрелки соприкасаются с блоком. Стрелка не должна входить в угол блока

Существует 5 типов взаимодействия между блоками:

- выход-вход

- выход-управление

- обратная связь выход-управление

- обратный выход-вход

- выход-механизм (обратного не может быть, т.к. входящая дуга)

Подготовка к моделированию

На самом раннем этапе моделирования перед началом разработки модели необходимо определить ее направленность:

- контекст

- точка зрения

- цель

Контекст определяет объект модели как часть целого. Очерчивает границы модели с ее внешним окружением посредством описания внешних интерфейсов

Точка зрения специфицирует, что можно «увидеть» в контексте и под каким «узлом». Определяет позицию автора как наблюдателя системы или ее элемента и выбирается таким образом, чтобы получить максимально полезную информацию из разрабатываемой модели

Цель определяет назначение модели или обеспечиваемых ею взаимодействий. Воплощает причину, по которой модель создана

Каждая модель представляет только одну точку зрения. Одна модель преследует только одну цель с определенной точки зрения

Выбор цели осуществляется с учетом тех вопросов, на которые должна ответить модель

Выбор точки зрения – в соответствии с выбранной позицией ответа на поставленные вопросы

Модель может содержать четыре типа диаграмм:

- контекстную диаграмм (в каждой модели может быть только одна контекстная диаграмма)

- диаграммы декомпозиции

- диаграммы дерева узлов

- диаграммы только для экспозиции

Контекстная диаграмма

Диаграмма наиболее абстрактного уровня описания системы в целом, содержащей определение субъекта моделирования, цели и точки зрения на модель

(Составить список всех функций, попытаться объединить)

Любой блок может быть конечным уровнем абстракции, то нумерация блока цифровая, числовая

Если же непонятно, если неполно отображается то, что делает данная функция – добавляем A

Диаграммы IDEF0 имеют двойную нумерацию

Диаграммы имеют номера по узлу, который они раскрывают и сквозную нумерацию

Контекстная диаграмма всегда имеет номер A-0

Декомпозиция контекстной диаграммы – номер A0, остальные диаграммы декомпозиции – номера по соответствующему узлу

Название – обычно пишется название узла. Оно описывает контекст того, что происходит. Для A0 – название продукта

Туннельные дуги – означают, что данные, выраженные этими дугами не рассматриваются на соответствующем уровне детализации

Дуга, помещенная в туннель в месте присоединения к блоку, означает, что данные, выраженные ею, не будут рассматриваться на диаграмме-потомке

Дуга, помещаемая в туннель на свободном конце, означает, что данные, выраженные ею, не относятся к родительскому блоку и на ней не описываются

Соглашения по построению диаграмм

Блоки располагаются на диаграмме от левого верхнего угла листа до правого нижнего и нумеруются в том же порядке

При декомпозиции блока (особенно при коллективной работе) снаружи правого нижнего угла блока записывается С-номер диаграммы

Стремитесь максимально увеличить расстояние между блоками и поворотами дуг, а также между пересечениями последних

Объединение дуг, источники которых не выделены на диаграмме позволяют графически подчеркнуть единый источник исходных данных

При соединении блоков избегайте пересечения дуг

Минимизируйте число петель и поворотов каждой дуги. Это также упростит диаграмму и повысит ее читабельность

Технология проектирования ИС подразумевает сначала создание модели AS-IS, ее анализ и улучшение бизнес-процессов, то есть создание модели TO-BE, и только на основе модели TO-BE строится модель данных, прототип и затем окончательный вариант ИС

Итог:

- методология IDEF0 представляет собой четко формализованные подход к созданию функциональных моделей – структурных схем изучаемой системы

- схемы строятся по иерархическому принципу с необходимой степенью подробности

- совокупность схем (IDEF0 диаграмм) образует модель системы. Эта модель носит качественный, описательный, декларативный характер

**ЛК 6**

Пример построения IDEF0 и IDEF3

Дихотомия

Интерфейс-реализация

Класс-объект

Что-то по умл

Три механизма:

\*помеченные (тегированные) значения, ограничения, стереотипы

Стереотип – расширение к словарю языка, которое позволяет связать с моделью новый семантический смысл <<>>

Тегированное значение – расширение свойства элемента модели (пара: имя свойства и значение свойства, еоторая м.б. добавлена к любому стандартному элементу модели) {}

Ограничение – логическое утверждение относительно значений свойств элементов модели