

Оглавление

1. Понятие автоматизации. Основные требования к автоматизации. Цели автоматизации. Процессы автоматизации. Основные термины и определения.	2
2. Стандартизация систем. Виды ГОСТов регламентирующих понятие ИС, АС, АСУП и т.д.	2
3. Анализ объекта автоматизации. Цели анализа. Принципы анализа и синтеза систем.	3
4. Анализ объекта автоматизации. Понятие методологии анализа объекта автоматизации. Виды методологий.	3
5. Анализ объекта автоматизации. Основные особенности и назначения методологий SADT, UML, Aris, BPMN.	3
6. Стандартизация систем. Классификация ИС. Основные подходы к автоматизации. ГОСТы.	4
7. Стандартизация систем. Модели создания ИС. Платформы реализации ИС.	5
8. Стандартизация систем. Методологии разработки и внедрения ИС.	6
9. Стандартизация систем. ГОСТ Р МЭК 62264-1-2010 Интеграция систем управления предприятием.	7
10. Стандартизация систем. Постановка задачи автоматизации предприятия. Функциональная модель управления предприятием.	8
11. Стандартизация систем. ГОСТ Р МЭК 62264-1-2010 Контур управления. Функции управления.	9
12. Стандартизация систем. Переход от функциональной модели предприятия к модели потоков данных.	12
13. Нотация DFD.	13
14. Стандартизация систем. Переход от модели потоков данных процессов предприятия к объектной модели данных.	15
15. Стандартизация систем. ГОСТ Р МЭК 62264-1-2010 Категории информации.	16
16. Нотация UML (диаграмма классов).	16
17. Объектные модели ресурсов: персонал, оборудование, материал.	19
18. Объектные модели возможности производства, Объектные модели графика производства.	23
19. Объектные модели определения изделия.	26
20. Формирование требований к ИС.	28
21. Управление требованиями. Цикл формирования требований. Анализ и структурирование первичных требований заказчика.	30
22. Трехуровневая модель построения ИС.	33
23. Особенности формирования модели данных. Виды СУБД.	34
24. Модели описания информационной архитектуры. ER-модели, UML, IDEF1x.	36
25. Модульный принцип построения ИС.	37
26. Описание программной архитектуры. Шаблоны проектирования.	37
27. Интерфейс пользователя ИС. Виды интерфейсов.	37
28. Особенности проектирования пользовательских интерфейсов. Подход психологии восприятия. Подход инженерной психологии. Подход когнитивной психологии.	38
29. Особенности интеграции ИС. Программные интерфейсы.	38
30. Особенности ввода в эксплуатацию ИС. Сопровождение внедрения ИС.	39
31. Требования к инфраструктуре ИС.	40
32. Особенности выбора прототипа ИС.	42
33. Особенности оценки качества ПО. Подходы к организации оценки качества ИС.	42
34. Документационное сопровождение ИС. Документация передаваемая пользователю.	44

1. Понятие автоматизации. Основные требования к автоматизации. Цели автоматизации.

Процессы автоматизации. Основные термины и определения.

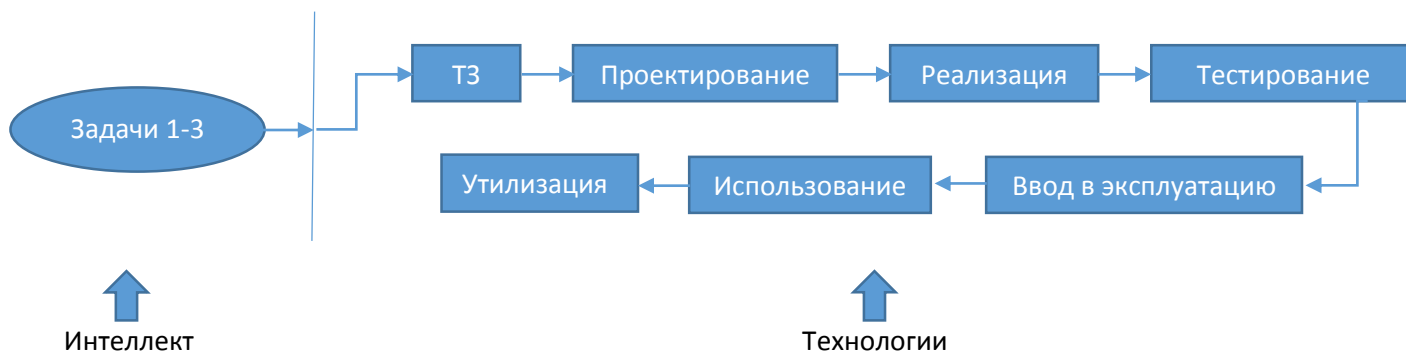
Автоматизация предприятия – введение (ввод в эксплуатацию) ИС.

Пример автоматизированной системы – станок с ЧПУ (раб. с данными, а не с инф.)

Проект внедрения (автоматизации) –

Основные (проблемы) задачи автоматизации:

- Отсутствие постановки задачи, менеджмента на предприятии. Нет комплексного подхода, четко формулирующего цели управления предприятием.
- Необходимость частичной или полной реорганизации структуры предприятия.
- Необходимость изменения технологии бизнеса.
- Сопротивление со стороны сотрудника. Решение проблемы ротации кадров.
- Необходимость создания группы внедрения (четкая постановка задачи; решение проблемы прибыли; тестирование систем; обучение пользователей).



(ЖЦИС, ГОСТ 12.207)

ИС:

- Производственные (складсистемы, автоматизированные станки, складирование)
- Управляющие предприятием (ERP-системы)

Задачи ИС:

- Создание инф. модели производства;
- Построение корпоративных информационно-управляющих систем;
- Интеграция производственных ИС;
- Разработка модели «как есть» и «как быть»;
- Проектирование производственной ИС.

Р МЭК 62264-N-20XX (6 частей)

65264-N-2010:

Часть I: «Модели и терминология»

Часть II: «Атрибуты объектных моделей»

Часть III: «Атрибуты модели управления производственными операциями»

Часть V: «Транзакции между корпоративными и производственными ИС»

2. Стандартизация систем. Виды ГОСТов регламентирующих понятие ИС, АС, АСУП и т.д.

- ГОСТы группы 34 (<http://www.swrit.ru/gost-34.html>)
- ГОСТ 34.003-90 (http://docs.nevacert.ru/files/gost/gost_34.003-1990.pdf) – «Автоматизированные системы. Термины и определения» (АС; ИАС; виды, а также функции и задачи АС, ее основные компоненты и др.)

- ГОСТ 24.103-84 (http://techwriters.ru/services-for-technical-writers/gosts-and-standards/gost/gost_detail.php?ELEMENT_ID=334) – «Автоматизированные системы управления. Общие положения»
- ГОСТ 24.104-85 (<http://pro-spo.ru/glossary/3105--24104-85->) – «Автоматизированные системы управления. Общие требования» (АСУ, АСУП, требования к АСУ)
- ГОСТ 12-1207 – Модели жизненного цикла ИС

3. Анализ объекта автоматизации. Цели анализа. Принципы анализа и синтеза систем.

Цель:

- сбор данных об объекте и осуществляемых им видов деятельности
- обследование пользования
- обследование внешней инф инфраструктуры
- оценка качества функций объекта и осуществ. им видов деятельности; на основе этой оценки:
- выявление проблем
- оценка целесообразности

Стратегии:

1) децентрализованная стратегия (функциональный подход) – последовательное проектирование функциональных подсистем. Выполняется относительно независимых функциональных подсистем. Минус – сложность обеспечения оптимальности реализации и функционирования разработанной ОС (ИС?)

2) централизованная стратегия (информационный подход) – анализ информационных потоков с целью организации единого информативного пространства с последовательным выделением на них функций и задач ИС

4. Анализ объекта автоматизации. Понятие методологии анализа объекта автоматизации. Виды методологий.

Цель:

- сбор данных об объекте и осуществляемых им видов деятельности
- обследование пользования
- обследование внешней инф инфраструктуры
- оценка качества функций объекта и осуществ. им видов деятельности; на основе этой оценки:
- выявление проблем
- оценка целесообразности

Стратегии:

1) децентрализованная стратегия (функциональный подход) – последовательное проектирование функциональных подсистем. Выполняется относительно независимых функциональных подсистем. Минус – сложность обеспечения оптимальности реализации и функционирования разработанной ОС (ИС?)

2) централизованная стратегия (информационный подход) – анализ информационных потоков с целью организации единого информативного пространства с последовательным выделением на них функций и задач ИС

Методология – комплекс из средств анализа, языка анализа и правил использования.

Виды: BPMN, UML, SADT, Aris

5. Анализ объекта автоматизации. Основные особенности и назначения методологий SADT, UML, Aris, BPMN.

Анализ объекта автоматизации проводится на основе требований предприятия.

Комплекс результатов полученных на основе анализа является исходными данными для ТЗ.

1. Business Process Model & Notation

Направлена на анализ и моделирование бизнес-процессов, рассматривает объект автоматизации с точки зрения операций, выполняемых субъектами объекта автоматизации.

Является стандартом ISO 2003 (19510)

Минус – отсутствие (формализованное) возможного описания модели данных необходимого или существующего в модели данных, а также формализованной организационной структуры

Примитивы:

- объекты потока управления: события, действия, логические операторы
- соединительные объекты: поток управления, поток сообщений, поток ассоциации¹
- артефакты: данные (результат действия/события), бумажные документы
- роли

2. UML

19505-1(2):2012

Применяется тогда, когда рассматриваем объект автоматизации с точки зрения существующей ИС

3. SADT

Объединяет набор методологий IDEF

IDEF.O – стандарт ISO в РФ – методология функционального моделирования ГОСТ Р 50.1.028

Подразумевает подход к моделированию с точки зрения функций и от общего к частному

Минусы – не показывает четкую структуру; не позволяет формализовать модель данных

IDEF.1X – методология построения реляционных моделей данных

IDEF.3 (вместе с IDEF.O) – формализует технологические процессы, т.е. алгоритмы выполнения функций

4. ARIS

Основное назначение – анализ и моделирование бизнес-процессов организации

Единственный комплекс программных продуктов, позволяющий осуществить анализ эффективности процессов.

Рассматривается объект автоматизации с точки зрения бизнес-процесса, данных

ARIS-Express – поддерживает BPMN

6. Стандартизация систем. Классификация ИС. Основные подходы к автоматизации. ГОСТы.

Классификация ИС:

С точки зрения создания:

- заказные (разрабатываются с нуля под конкретного заказчика)
- тиражируемые (базируются на прототипах)
 - простые(коробочные): узкая функциональность
 - среднего класса: объединяют несколько функций
 - высшего класса: серьезный мониторинг бизнес-процессов, происходит модернизация бизнес-процессов

По масштабу применения:

- локальные (в рамках одного рабочего места)
- местные (в пределах организации)
- территориальные (в пределах единой административной территории)
- отраслевые (рассматривается отрасль без привязки к территории)

По режиму использования:

- системы пакетной обработки (учебные, системы информационного обслуживания)
- запрос на ответные системы (билетные, библиотечные)
- диалоговые системы
- системы реального времени

По характеру представления информации:

- фактографические (работают с данными, представленными в виде модели данных)
- документальные (единица хранения – документ, а не содержание: системы электронного документооборота)

Основные подходы к автоматизации предприятий:

1) канонический – в основе идеология пром инженерной разработки, перенесенная на создание ИС и описанная в стандарте 34.601

Подход подразумевает разработку ИС с нуля индивидуально под каждого заказчика

34.601 выделяет стадии:

- анализ требований
- проектирование
- кодирование
- тестирование и отладка
- эксплуатация

Подход позволяет получить результат максимально адаптированный к требованиям заказчика, но ресурсоемкий; т.е. выполнение всех стадий требует много персонала и времени.

Реализ. – в сложных ИС, подразумевающих высокие риски

– на объектах, слабо подверженных изменениям

2) типовое проектирование – в основе принцип исп прототипа

Прототип – некий тиражируемый программный продукт, обладающий заявленными свойствами, реализующий заявленные функции, имеющий инструментарий разработки и методологию внедрения

Преимущества – сокращение издержек за счет отсутствия необходимости полноценного проектирования и тестирования элементов системы.

На этапе анализа необходимо выделить функциональные задачи и подобрать оптимальный прототип.

ГОСТ 34.601 Автоматизированные системы. Стадии создания

ГОСТ 12.207 Модели жизненного цикла -> 15.271 (?)

7. Стандартизация систем. Модели создания ИС. Платформы реализации ИС.

При создании ИС используется системный подход: двухаспектный анализ – макроанализ и микроанализ.

При макроанализе ее элементы исследуются как часть системы более высокого ранга, особое внимание уделяется информационным связям (устанавливается их кол-во и св-ва). Основная цель – выбор наиболее предпочтительных связей, наиболее эффективно реализовать цель системы.

При микроанализе изучается строение объекта и анализируются его составные элементы (с т.з. характеристик).

На стадии проектирования такой подход позволяет строить модель будущей ИС, основываясь на мат. подходе.

В ГОСТе 34.003 формулируются следующие виды структур (на основании которых описывается модель):

- Функциональная: элементы – функции, задачи – операции, связи – информационные связи
- Техническая – элементы – устройства, связи – физ. линии связи
- Организационная – элементы – коллективы людей, связи – инф. (или подчинения)
- Алгоритмическая – элементы – алгоритмы, связи – информация
- Программная – элементы – программные модули, связи – инф. и управляющие
- Информационная – элементы – формы и представление информации в системе, связи – преобразование информации в системе

Требования:

- Система должна обеспечивать целенаправленное функционирование;
- Сложные ИС включают в себя системы более низкого уровня;

- Сложные системы постепенно расширяются и обновляются;
- ИС всегда можно представить в виде двух объектов – управляющего объекта и объекта управления.

При проектировании ИС используются принципы:

- Развития (учет обновления и пополнения функций)
- Совместимости (взаимодействие с другими системами)
- Стандартизации и модификации
- Эффективности (достижение рационального соотношения между затратами и целевыми эффектами)

8. Стандартизация систем. Методологии разработки и внедрения ИС.

Основные подходы к автоматизации предприятия:

1. Канонический – технические, промышленные, инженерные разработки, перенесенные на создание ИС (стандарт 64-601). ИС разрабатывается с нуля, индивидуально каждому заказчику.

Стадии:

- Анализ требований
- Проектирование
- Кодирование
- Тестирование и отладка
- Эксплуатация

Позволяет получить результат, максимально адаптированный к требованиям заказчика. Используется на объектах, слабо подверженных изменениям и в сложных ИС с высокими рисками.

2. Типовое проектирование – использование прототипа.

Прототип – некоторый тиражируемый программный продукт, обладающий заявленными свойствами и реализующий заявленные функции, имеющий инструментарий разработки и методологию внедрения.

Сокращение издержек за счет отсутствия необходимости полноценного проектирования и тестирования элементов системы.

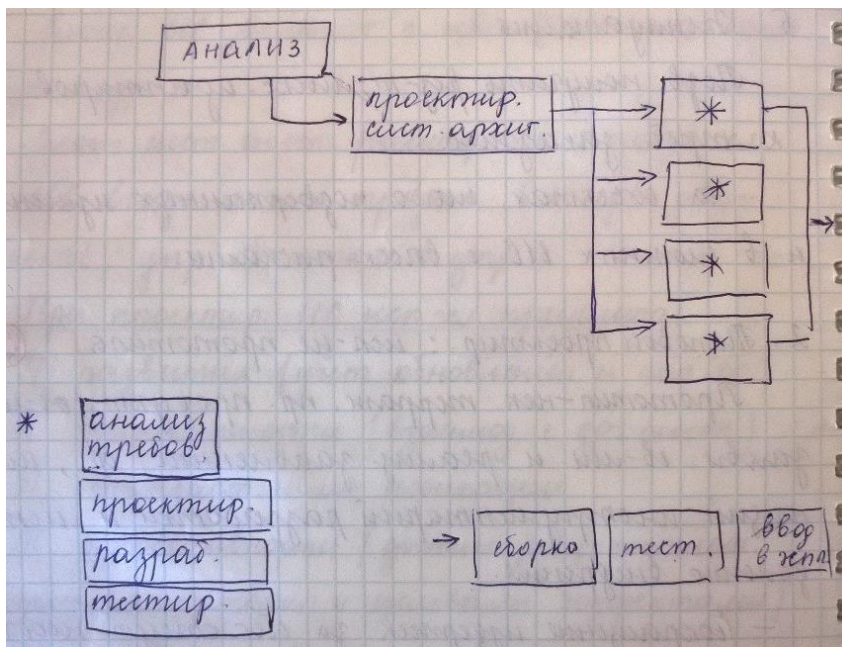
Первый этап – выделить функциональную задачу и подобрать прототип.

ГОСТ 12-207 – модели жизненного цикла ИС:

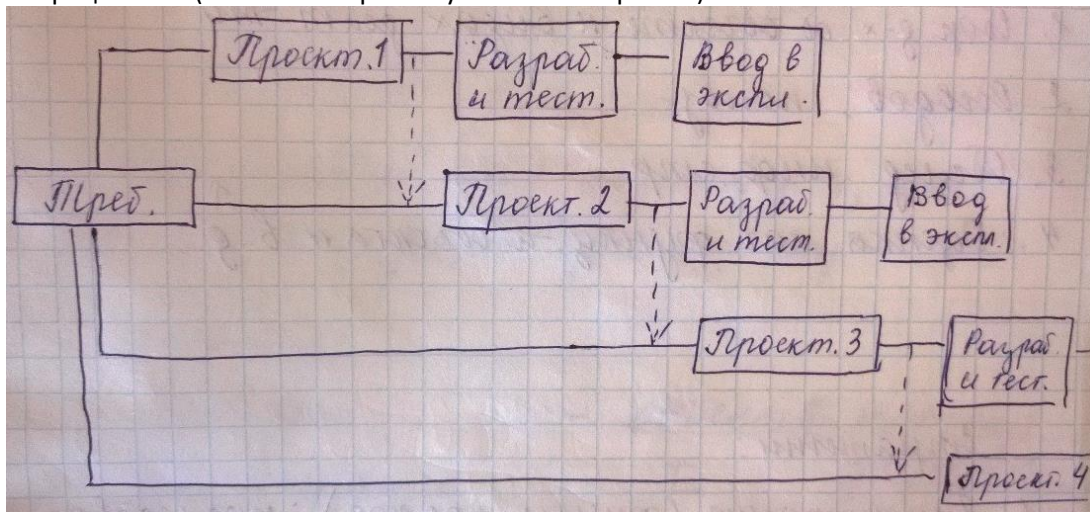
- Основные процессы
- Вспомогательные процессы
- Процессы документирования

Модели:

1. Каскадная



2. Итерационная (поэтапно с промежуточным контролем)



3. Спиральная
 - Подсистемные прототипы
 - Отраслевые прототипы

9. Стандартизация систем. ГОСТ Р МЭК 62264-1-2010 Интеграция систем управления предприятием.

Р МЭК 62264-N-20XX (6 частей)

65264-N-2010:

Часть I: «Модели и терминология»

Часть II: «Атрибуты объектных моделей»

Часть III: «Атрибуты модели управления производственными операциями»

Часть V: «Транзакции между корпоративными и производственными ИС»

ИС:

- Производственные (складсистемы, автоматизированные станки, складирование) – данные, алгоритмы
- Управляющие предприятием (ERP-системы) - информация

Задачи ИС:

- Создание инф. модели производства;
- Построение корпоративных информационно-управляющих систем;
- Интеграция производственных ИС;
- Разработка модели «как есть» и «как быть»;
- Проектирование производственной ИС.

Описание предметной области ---> функции (которые хотим автоматизировать; которые хотим автоматизировать и поэтапно автоматизируем; которые еще не автоматизированы) ---> формализация информационных потоков ---> категоризация информации ---> модели данных

4 уровень (кварталы, недели, месяцы)	Бизнес-логика	ERP-системы	Знания
3 уровень (сутки, смены, часы)	Управление производственными операциями	MES-системы	Потоки данных, информация
2 уровень	Периодическое/непрерывное/дискретное/рецептурное производство		Данные
1 уровень	Получение и обработка данных		Сигналы
0 уровень	Сам процесс		

10. Стандартизация систем. Постановка задачи автоматизации предприятия. Функциональная модель управления предприятием.

Задачи автоматизации:

Автоматизация - процесс в развитии машинного производства, при котором функции управления и контроля ранее выполняемые человеком передаются автоматически управляющим устройством.

В настоящее время направление автоматизации производства является создание таких высоко интенсивных технологических процессов, которые сводят до минимума участие людей. Поэтому перед полной автоматизацией ставятся следующие задачи:

1. Экономичное использование физического труда, материалов и энергии.
2. Возможность расширения производства без увеличения трудовых ресурсов
3. Полное исключение или существенное снижение отрицательного воздействия производственного процесса на человека.
4. Сокращение вреда от возникновения потребности в изделии до получения готовой продукции.
5. Получение очень высокой производительности труда и качества продукции.

Функциональная модель

Представляет функции предприятия, связанные с производственным процессом. Представляет функции предприятия, связанные с производственным процессом. Структура представленной модели отражает не внутреннюю организационную структуру гипотетической компании, а организационную структуру её функций. Используется в ИСА 95 для обозначения интерфейса “Предприятие – система управления”, но может быть использована для рассмотрения функциональных составляющих предприятия, т. е. предприятия в функциональном разрезе.

1. Обработка заказов

- а) Прием и подтверждение заказов
- б) прогнозирование сбыта
- в) отчетность по валовой прибыли
- г) размещение производственных заказов

2. Планирование производства

- а) составление графика производства
- б) определение долгосрочных потребностей в сырье

3. Управление производством

- а) контроль превращения исходных материалов в готовую продукцию в соответствии с графиком производства и стандартами производства
- б) определение потребностей в сырье
- с) выпуск отчетов по производительности и затратам

4. Учет материалов и энергозатрат

- а) управление запасами, перемещением и качеством материалов и энергии
- б) формирование заявок на приобретение материалов и энергии
- с) регистрация поступления закупленных материалов и энергии

5. Поставки

- а) Размещение заказов у поставщиков
- б) обработка входящих счетов для оплаты

6. Управление качеством

- а) Контроль и классификация материалов
- б) установление стандартов качества материалов
- с) проверка изделий

7. Управление запасами (управление запасами готовой продукции)

8. Расчет стоимости

9. Отгрузка товара

- а) организация транспортировки продукции
- б) подготовка сопроводительных документов на отгрузку

10. ТОиР

- а) Техобслуживание оборудования
- б) составление плана профилактического обслуживания

11. Маркетинг

формирование планов сбыта, планов маркетинга

12. НИР и технологии

Разработка новых изделий, определение технологических требований

11. Стандартизация систем. ГОСТ Р МЭК 62264-1-2010 Контур управления. Функции управления.

Функции

Обработка заказов (1.0)

- а) прием и подтверждение заказов потребителей;
- б) прогнозирование сбыта;
- с) обработка дефектных ведомостей и резервирование материалов;
- д) отчетность по валовой прибыли;
- е) размещение производственных заказов.

Как правило, не бывает прямого интерфейса между функциями обработки заказов и функциями управления производственным процессом.

Календарное планирование производства (2.0)

Функции календарного планирования производства связаны с функциями систем управления технологическими процессами через график производства, информацию о ходе производства и информацию о производственных мощностях. Обмен информацией осуществляется в рамках функций управления производством. Детальное календарное планирование работы производственного участка считается функцией управления.

- а) составление календарного графика производства;
- б) определение долгосрочных потребностей в сырье;
- с) формирование графика упаковки конечной продукции;
- д) определение наличной продукции для продажи.

Управление производством (3.0)

Базовые функции управления производством

- a) контроль превращения исходных материалов в готовую продукцию в соответствии с производственным графиком и стандартами производства;
- b) выполнение внутренних конструкторских разработок и обновление планов технологической подготовки производства;
- c) определение потребностей в сырье;
- d) выпуск отчетов по производительности и затратам;
- e) оценку ограничений производственных возможностей и уровня качества;
- f) самоконтроль и диагностику производственного и управляющего оборудования;
- g) разработку стандартов производства, технологических инструкций по типовым рабочим операциям, рецептур исходных материалов и правил работы со специальным технологическим оборудованием.

Контроль расхода материалов и энергии (4.0)

- h) управление запасами, перемещением и качеством материалов и энергии;
- i) формирование заявок на приобретение материалов и энергии на основе анализа краткосрочных и долгосрочных потребностей производства;
- j) расчет и документирование баланса запасов и определение потерь используемых материалов и энергии;
- k) получение поступающих материалов и энергии и запрашивание результатов входного контроля качества;
- l) регистрация поступления закупленных материалов и энергии.

Закупки (5.0)

- m) размещение заказов у поставщиков сырья, электроэнергии, запчастей, инструмента, оборудования и прочих требуемых материалов;
- n) контроль хода закупок и уведомление подателей заявок;
- o) обработка входящих счетов для оплаты после получения и контроля закупленных товаров;
- p) сбор и обработка отдельных заявок на исходные материалы, запчасти и т.п. для размещения заказов у поставщиков.

Обеспечение качества (6.0)

- q) контроль и классификация материалов;
- r) установление стандартов качества материалов;
- s) рассылка стандартов по технологическим и испытательным лабораториям соответственно потребностям технологических и маркетинговых служб, а также служб по работе с клиентами;
- t) сбор и накопление информации по качеству материалов;
- u) выпуск материалов для последующего использования (отгрузки либо дальнейшей переработки);
- v) сертификация изделий, произведенных в соответствии со стандартными требованиями к технологическому процессу;
- w) проверка характеристик изделий на соответствие требованиям заказчика и проведение статистического контроля продукции перед отгрузкой для обеспечения надлежащего уровня качества;
- x) фиксация отклонений характеристик материалов и изделий от стандартов и передача этой информации технологам для оценки и последующего совершенствования технологических процессов.

Управление производственными запасами (7.0)

- a) управление запасами готовой продукции;
- b) накопление запасов конкретного изделия в соответствии с указаниями по его продаже;
- c) инициирование упаковки конечной продукции согласно графику отгрузки;
- d) сообщение сведений о запасах службе планирования;
- e) отчетность о производственных потерях и текущем состоянии производства для обеспечения калькуляции себестоимости;
- f) организация погрузки и отправки продукции во взаимодействии с руководством, ответственным за поставки.

Калькуляция себестоимости продукции (8.0)

- a) проведение расчетов и формирование отчета по суммарной величине затрат на выпуск изделия;

- b) уведомление производственного отдела о результирующих затратах для надлежащего регулирования производства;
- c) установление целевых показателей затрат для производства;
- d) сбор информации о расходах сырья, труда, энергии, а также других расходах для отчетности;
- e) проведение расчетов и формирование отчета по суммарной величине производственных затрат, уведомление производственного отдела о результирующих затратах для надлежащего регулирования производства;
- f) установление целевых показателей затрат на приобретение и распределение материалов и энергии.

Организация отгрузки продукции (9.0)

- a) организация транспортирования продукции для отгрузки в соответствии с требованиями принятых заказов;
- b) проведение переговоров и размещение заказов в транспортных компаниях;
- c) приемка отправляемых грузов на производственную площадку и оформление разрешения на отгрузку;
- d) подготовка сопроводительных документов на отгрузку (транспортная накладная, таможенная очистка);
- e) подтверждение отгрузки и разрешение оформления общего счета;
- f) уведомление производственной бухгалтерии о затратах на отгрузку.

Управление техническим обслуживанием (10.0)

- g) обеспечение техобслуживания действующего оборудования;
- h) составление плана профилактического обслуживания;
- i) обеспечение текущего контроля оборудования в целях прогнозирования отказов, включая планирование процедур самоконтроля и технической диагностики;
- j) размещение заказов на приобретение материалов и запчастей;
- k) выпуск отчетов о затратах на техобслуживание и координация сторонних контрактных работ;
- l) обеспечение обратной связи со службой технологической подготовки производства по текущему состоянию обслуживаемого оборудования в части характеристик производительности и надежности.

Исследования, разработки и проектирование

- a) разработка новых изделий;
- b) определение технологических требований;
- c) определение требований к характеристикам изделий в плане технологии их изготовления.

Маркетинг и сбыт

- a) формирование планов сбыта;
- b) формирование планов маркетинга;
- c) определение требований заказчиков к производимым изделиям;
- d) определение технических условий и стандартов на изделия;
- e) взаимодействие с заказчиками.

Контур управления

- 1) Контур управления в экономических системах

Контур управления процессом в экономических системах можно представить с помощью цикла PDCA.

PDCA (Plan-Do-Check-Act) — цикл организационного управления, или Цикл Деминга-Шухарта. Согласно этому циклу, управление должно циклически проходить по следующим стадиям: Планирование (проектирование) – Реализация (работа) - Контроль (Анализ) – Корректировка (регулировка).

1.Планирование

установление целей и процессов, необходимых для достижения целей, планирование работ по достижению целей процесса и удовлетворения потребителя, планирование выделения и распределения необходимых ресурсов.

2.Выполнение

выполнение запланированных работ.

3.Проверка

сбор информации и контроль результата на основе ключевых показателей эффективности (KPI), получившегося в ходе выполнения процесса, выявление и анализ отклонений, установление причин отклонений.

4. Воздействие (управление, корректировка)

принятие мер по устранению причин отклонений от запланированного результата, изменения в планировании и распределении ресурсов.

Цикл PDCA получил широкое распространение в практической деятельности многих компаний.

2) Контур управления в технических системах.

Контур управления в технических системах – замкнутая совокупность элементов системы управления (связка исполнителя и управляющего устройства). В зависимости от логики функционирования могут находиться в разной последовательности. В контуре управления каждый элемент системы воздействует на последующий и получает, в свою очередь, воздействие от предыдущего. Воздействие осуществляется по информационным каналам. Система управления может содержать один или несколько контуров управления. Бывает прямой и обратный. Прямой: от органа управления исполнителю поступает команда, после этого обработка результата в зависимости от работы исполнителя. Обратный: сначала работа исполнителя, после – корректировка.

- 1) Исполнитель посылает устройству данные и управляющие команды, устройство возвращает результат работы.
- 2) Управляющее устройство корректирует свою работу в зависимости от результатов деятельности. (Исполнитель корректирует работу управляющего устройства в зависимости от результатов деятельности.)

Контур управления:

- Объекты контура
- Формализация потоков (Что передается)
- Порядок (При каких условиях)

Оперативное управление – полная автоматизация контура, либо быстрое прохождение по нему.

Контур может быть декомпозирован.

Декомпозиция — научный метод, использующий структуру задачи и позволяющий заменить решение одной большой задачи решением серии меньших задач, пусть и взаимосвязанных, но более простых.

12. Стандартизация систем. Переход от функциональной модели предприятия к модели потоков данных.

Одним из важнейших способов описания процесса являются диаграммы потоков данных (информации) DFD (Data Flow Diagram).

Часто нотацию DFD путают с простым описанием потоков информации между подразделениями. Это далеко не одно и то же. Почему нельзя рассматривать простое описание потоков между подразделениями организации как схему процесса? В каждом большом подразделении (например, отдел сбыта крупного предприятия) выполняются различные бизнес-процессы. Часто у этих процессов существуют различные внутренние и внешние клиенты. Именно поэтому схема потоков информации между подразделениями, описывает только потоки данных, пересекающие границы подразделений, но не содержит всей информации о внешних и внутренних изменениях потока информации. То есть диаграмма DFD содержит шаги модификаций/изменений информации от одного действия к другому. При этом, описание потоков информации между подразделениями является практически важным и широко используемым инструментом.

Пример описания процесса в DFD можно усложнить, используя понятие «хранилище данных». Под этим понимается любой носитель информации, например, бумажный документ, электронный файл, промышленная база данных на сервере организации и т.д. При построении модели процесса с использованием хранилищ данных, необходимо помнить, что данные (информация) не могут перемещаться между функциями процесса сами по себе. Их можно передавать только через определенных посредников — носителей информации или, что то же самое, хранилищ данных.

Для чего служат нотации DFD? В первую очередь они нужны для описания реально существующих в организации потоков данных. Описания могут создаваться как по процессному, так и по функциональному признаку. В первом

случае мы получаем модели бизнес-процессов в формате DFD, во втором — схему обмена данными между подразделениями. Созданные модели потоков Данных организации могут быть использованы при решении таких задач, как:

- определение существующих хранилищ данных (текстовые документы, файлы, Система управления базой данных — СУБД);
- определение и анализ данных, необходимых для выполнения каждой функции процесса;
- подготовка к созданию модели структуры данных организации, так называемая ERD-модель (IDEF1X);
- выделение основных и вспомогательных бизнес-процессов организации.

Следует отметить что нотация DFD может быть эффективно применена для описания потоков документов или потоков материальных ресурсов.




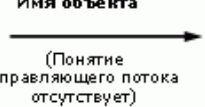
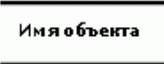

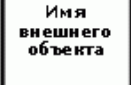
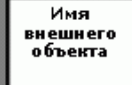
Более того, нотация DFD может быть несколько модернизирована таким образом, чтобы на одной диаграмме можно было бы показать как потоки данных, так и потоки материальных ресурсов.

На практике при создании моделей процессов часто бывает полезно использовать несколько способов описания. Сначала, например, мы создаем модель в нотации IDEF0, выявляем функции, входящие в процесс. Затем проводим декомпозицию процесса. При достижении некоторого уровня детализации (три-четыре) становится целесообразно сформировать для каждого детального процесса несколько схем в различных форматах: управление — IDEF0, а потоки данные и материалов — в DFD.

13. Нотация DFD

При построении DFD-схемы бизнес-процесса нужно помнить, что данная схема показывает потоки материальных и информационных потоков и ни в коем случае не говорит о временной последовательности работ, хотя в большинстве случаев временная последовательность работ и совпадает с направлением движения потоков в бизнес-процессе.

Существуют две нотации DFD:

Элемент	Описание	Нотация Йордана-Де Марко	Нотация Гейна-Сарсона
Функция	Работа.		
Поток данных	Объект, над которым выполняется работа. Может быть логическим или управляющим. (Управляющие потоки обозначаются пунктирной линией со стрелкой).		
Хранилище данных	Структура для хранения информационных объектов.		
Внешняя сущность	Внешний по отношению к системе объект, обменивающийся с ней потоками.		

Требования к оформлению функций:

1. Каждая функция должна иметь идентификатор;
2. Названия работы нужно формулировать согласно следующему формуле:
Название работы = Действие + Объект, над которым действие осуществляется

Требования к оформлению потока данных:

1. Название потока нужно формулировать согласно следующей формуле:
Название потока = Объект, представляющий поток + Статус объекта
2. Название должно быть по возможности кратким и состоять из 2-3 слов.

Построение DFD-модели

Построение DFD-модели базируется на принципе декомпозиции. DFD-модель включает в себя три документа, которые ссылаются друг на друга: Графические диаграммы, Миниспецификация, Словарь данных.

1. Контекстная диаграмма и детализация контекстной диаграммы

Первым шагом является построение контекстной диаграммы. Диаграмма имеет звездообразную топологию, в центре которой находится так называемый главный процесс, соединенный с приемниками и источниками информации, посредством которых с системой взаимодействуют пользователи и другие внешние системы. При этом контекстная диаграмма верхнего уровня содержит не единственный главный процесс, а набор подсистем, соединенных потоками данных. Контекстные диаграммы следующего уровня детализируют контекст и структуру подсистем. Для каждой подсистемы, присутствующей на контекстных диаграммах, выполняется ее детализация при помощи диаграммы DFD. Каждый процесс, в свою очередь, может быть детализирован при помощи отдельной диаграммы или миниспецификации, рис.1..

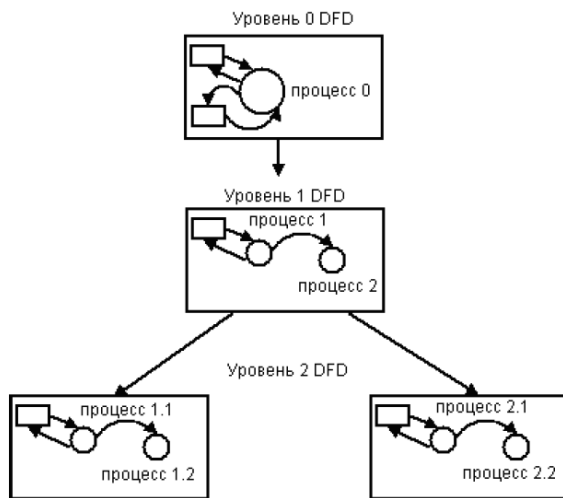


рисунок 1. Детализация моделей DFD

При детализации должны выполняться следующие правила:

правило балансировки — при детализации процесса дочерняя диаграмма в качестве внешних источников/приемников данных может иметь только те компоненты (подсистемы, процессы, внешние сущности, накопители данных), с которыми имеет информационную связь соответствующий процесс на родительской диаграмме;

правило нумерации — при детализации процессов должна поддерживаться их иерархическая нумерация.

правило семи — для того, чтобы диаграмма легко читалась, количество функций на диаграмме не должно быть больше семи.

Например, процессы, детализирующие процесс с номером 12, получают номера 12.1, 12.2, 12.3 и т. д.

2. Миниспецификация

Миниспецификация — документ, детально описывающий логику процесса. Она содержит номер процесса, списки входных и выходных данных, тело процесса — подробный алгоритм функции, преобразующий входные потоки данных в выходные.

Миниспецификация является конечной вершиной иерархии модели DFD. Решение о завершении детализации процесса и использовании миниспецификации принимается аналитиком исходя из следующих критериев:

- у процесса небольшое количество входных и выходных потоков данных (2-3 потока);
- процесс можно описать в виде последовательного алгоритма;
- процесс выполняет единственную логическую функцию преобразования входной информации в выходную;
- описать логику процесса можно в виде миниспецификации небольшого объема (не более 20-30 строк).

3. Словарь данных

В словаре данных определяется структура и содержание всех потоков данных и накопителей данных, которые присутствуют на диаграммах.

Для каждого потока в словаре хранятся: имя потока, тип, атрибуты.

4. Проверка DFD модели

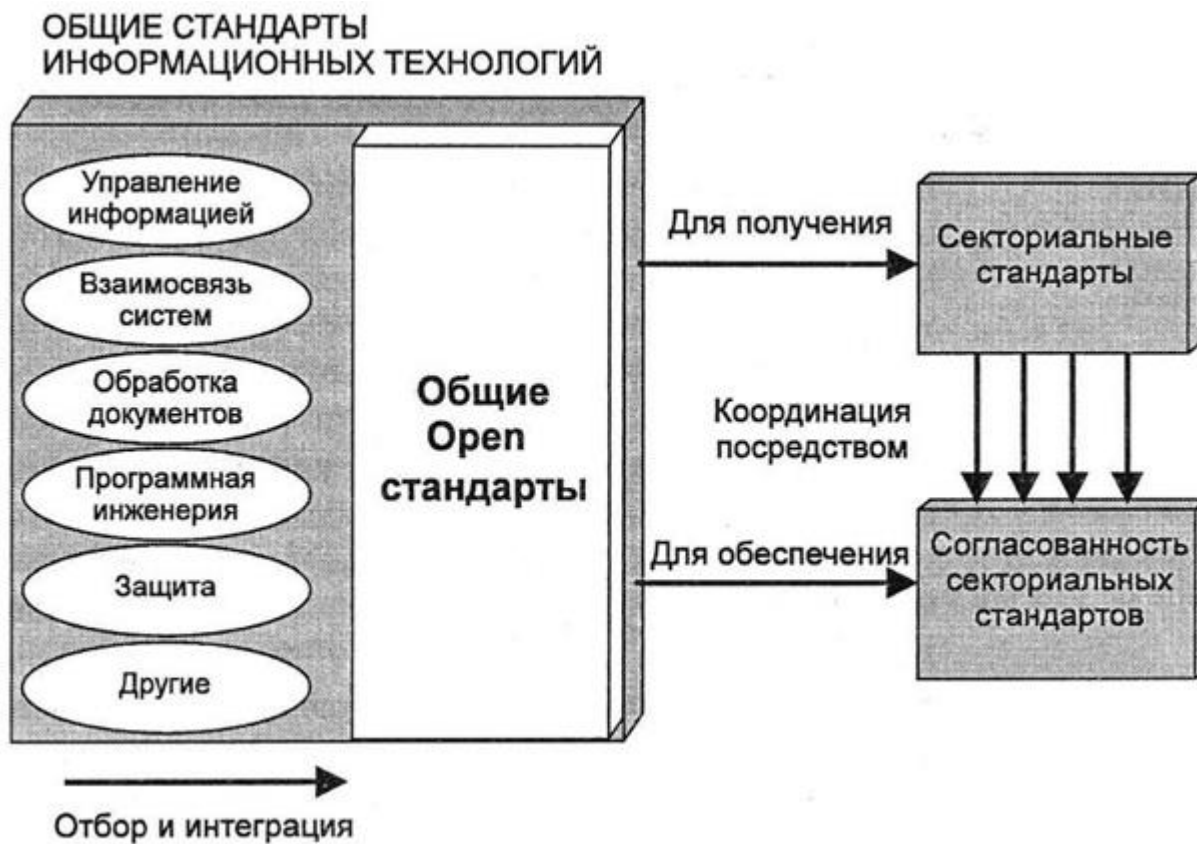
После построения законченной модели системы ее необходимо проверить на полноту и согласованность.

Модель считается **полной**, если все ее объекты (подсистемы, процессы, потоки данных) подробно описаны и детализованы.

Модель считается **согласованной**, если для всех потоков данных и накопителей данных выполняется правило сохранения информации: все поступающие куда-либо данные должны быть считаны, а все считываемые данные должны быть записаны.

14. Стандартизация систем. Переход от модели потоков данных процессов предприятия к объектной модели данных.

Общая схема стандартизации ИТ



Алгоритм:

1. Из DataFlow модели выбираются элементы, относящиеся к типу хранилища
2. Проверяется корректность их семантического назначения, достаточность этих функций для реализации системы
3. На основе потоков данных для каждой сущности хранения формируется набор атрибутов
4. Проверяется, что количество атрибутов достаточно для полного семантического описания сущности
5. Необходимо проверить атрибуты на связность между собой
6. Необходимо назначить ключевой атрибут (из сущностей / ввести новый)

15. Стандартизация систем. ГОСТ Р МЭК 62264-1-2010 Категории информации.

Информация на производственном предприятии, в соответствии с ИСА 95 классифицируется по трем основным областям:

1. информация о производственных мощностях для выпуска изделий

а) информация о производственных возможностях

Данные о доступности персонала, оборудования, исходных материалов для каждой производственной площадки, участка и элемента производства.

б) о техническом обслуживании

Сведения о текущем техническом состоянии оборудования.

в) о планировании использования производственных мощностей

Планирование мощностей – процесс определения производственной мощности, необходимой предприятию для реагирования на изменяющийся спрос на продукцию.

2. информация, требуемая технологическим процессам (информация определений изделий)

а) информация для календарного планирования (ведомость ресурсов)

Полный список ресурсов, требуемых для изготовления изделия (используемые материалы, персонал, оборудование, энергия, расходные материалы).

б) информация о материалах (ведомость материалов)

Список всех материалов, необходимых для изготовления изделия, с указанием их требуемого количества. В списке могут фигурировать исходное сырьё, промежуточные материалы, сборочные узлы, детали и расходные материалы, материалы, не относящиеся к процессу изготовления изделия (упаковка).

в) инструкции по изготовлению изделий

Это информация для настройки производственной операции на изготовление конкретного изделия. Это могут быть общие указания по организации рабочего места, описание основного способа производства, технологический маршрут или этапы сборки и т.д.

3. информация о фактическом ходе производства

а) информация о производстве (о предыстории производственного процесса)

Это все сохраненные сведения о ходе выпуска продукции. Эта информация может называться по-разному: журнал партий продукции, производственный журнал, книга хода производства.

б) о производственных запасах

Полные сведения о запасенных материалах, включая сведения о текущем состоянии материальных запасов. Как правило, все потребляемые и производимые материалы учитываются в составе информации о производственных запасах.

в) о календарном планировании

Информация, касающаяся выполнения запланированных производственных циклов (показатели производства, сведения о сегментах производственного процесса (длительность, объем потребляемых и производимых материалов)).

Сегмент производства – логическая группировка трудовых ресурсов, оборудования и материалов, требуемых для выполнения определенного шага производственного процесса.

Некоторые категории информации используют совместно системами управления производственными процессами и корпоративными системами: Информация, которая одновременно используется и в производственной системе предприятия и в системе управления.

16. Нотация UML (диаграмма классов).

Класс (class) — абстрактное описание множества однородных объектов, имеющих одинаковые атрибуты, операции и отношения с объектами других классов.

Графически класс в нотации языка UML изображается в виде прямоугольника, который дополнительно может быть разделен горизонтальными линиями на разделы или секции. В этих секциях могут указываться имя класса, атрибуты и операции класса.



Имя класса

Имя класса должно быть уникальным в пределах пакета, который может содержать одну или несколько диаграмм классов. Имя указывается в самой верхней секции прямоугольника, поэтому она часто называется секцией имени класса. В дополнение к общему правилу именования элементов языка UML, имя класса записывается по центру секции имени полужирным шрифтом и должно начинаться с заглавной буквы. Рекомендуется в качестве имен классов использовать существительные, записанные по практическим соображениям без пробелов. Необходимо помнить, что имена классов образуют словарь предметной области.

Класс может иметь или не иметь экземпляров или объектов. В зависимости от этого в языке UML различают конкретные и абстрактные классы.

Конкретный класс (concrete class) — класс, на основе которого могут быть непосредственно созданы экземпляры или объекты.

Абстрактный класс (abstract class) — класс, который не имеет экземпляров или объектов.

Для обозначения имени абстрактного класса используется наклонный шрифт (курсив). В языке UML принято общее соглашение о том, что любой текст, относящийся к абстрактному элементу, записывается курсивом. Это имеет принципиальное значение, поскольку является семантическим аспектом описания абстрактных элементов языка UML.

Атрибуты класса

Атрибут (attribute) — содержательная характеристика класса, описывающая множество значений, которые могут принимать отдельные объекты этого класса.

Каждому атрибуту класса соответствует отдельная строка текста, которая состоит из квантора видимости атрибута, имени атрибута, его кратности, типа значений атрибута и, возможно, его исходного значения. Общий формат записи отдельного атрибута класса следующий:

<квантор видимости> <имя атрибута> [кратность] :

<тип атрибута> = <исходное значение> {строка-свойство}.

Видимость в языке UML специфицируется с помощью квантора видимости (visibility), который может принимать одно из 4-х возможных значений и отображаться при помощи специальных символов.

- Символ " + " — public
- Символ " # " —protected
- Символ " - " — private
- Символ " ~ " - package

Имя атрибута представляет собой строку текста, которая используется в качестве идентификатора соответствующего атрибута и поэтому должна быть уникальной в пределах данного класса. Имя атрибута - единственный обязательный элемент синтаксического обозначения атрибута, должно начинаться со строчной (малой) буквы и не должно содержать пробелов.

Кратность (multiplicity) — спецификация области значений допустимой мощности, которой могут обладать соответствующие множества.

Если в качестве кратности указывается единственное число, то кратность атрибута принимается равной данному числу. Если же указывается единственный знак " * ", то это означает, что кратность атрибута может быть произвольным положительным целым числом или нулем. Если кратность атрибута не указана, то по умолчанию в языке UML принимается ее значение равное [1..1], т.е. в точности 1.

Знак " / " перед именем атрибута указывает на то, что данный атрибут является производным от некоторого другого атрибута этого же класса.

Производный атрибут (derived element) — атрибут класса, значение которого для отдельных объектов может быть вычислено посредством значений других атрибутов этого же объекта.

Операции класса

Операция (operation) - это сервис, предоставляемый каждым экземпляром или объектом класса по требованию своих клиентов, в качестве которых могут выступать другие объекты, в том числе и экземпляры данного класса.

Операции класса записываются в третьей сверху секции прямоугольника класса, которую часто называют секцией операций. При этом каждой операции класса соответствует отдельная строка, которая состоит из квантора видимости операции, имени операции, выражения типа возвращаемого операцией значения и, возможно, строка-свойство данной операции. Общий формат записи отдельной операции класса следующий:

<квантор видимости> <имя операции>(
 список параметров):

<выражение типа возвращаемого значения>
 {строка-свойство}

Имя операции представляет собой строку текста, которая используется в качестве идентификатора соответствующей операции и поэтому должна быть уникальной в пределах данного класса. Имя операции - единственный обязательный элемент синтаксического обозначения операции, должно начинаться со строчной (малой) буквы, и, как правило, записываться без пробелов.

Список параметров является перечнем разделенных запятой формальных параметров, каждый из которых, в свою очередь, может быть представлен в следующем виде:

<направление параметра> <имя параметра>:

<выражение типа> =
 <значение параметра по умолчанию>.







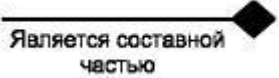
Параметр (parameter) — спецификация переменной операции, которая может быть изменена, передана или возвращена.

Операция с областью действия на весь класс показывается подчеркиванием имени и строки выражения типа. В этом случае под областью действия операции понимаются все объекты этого класса. В этом случае вся строка записи операции подчеркивается.

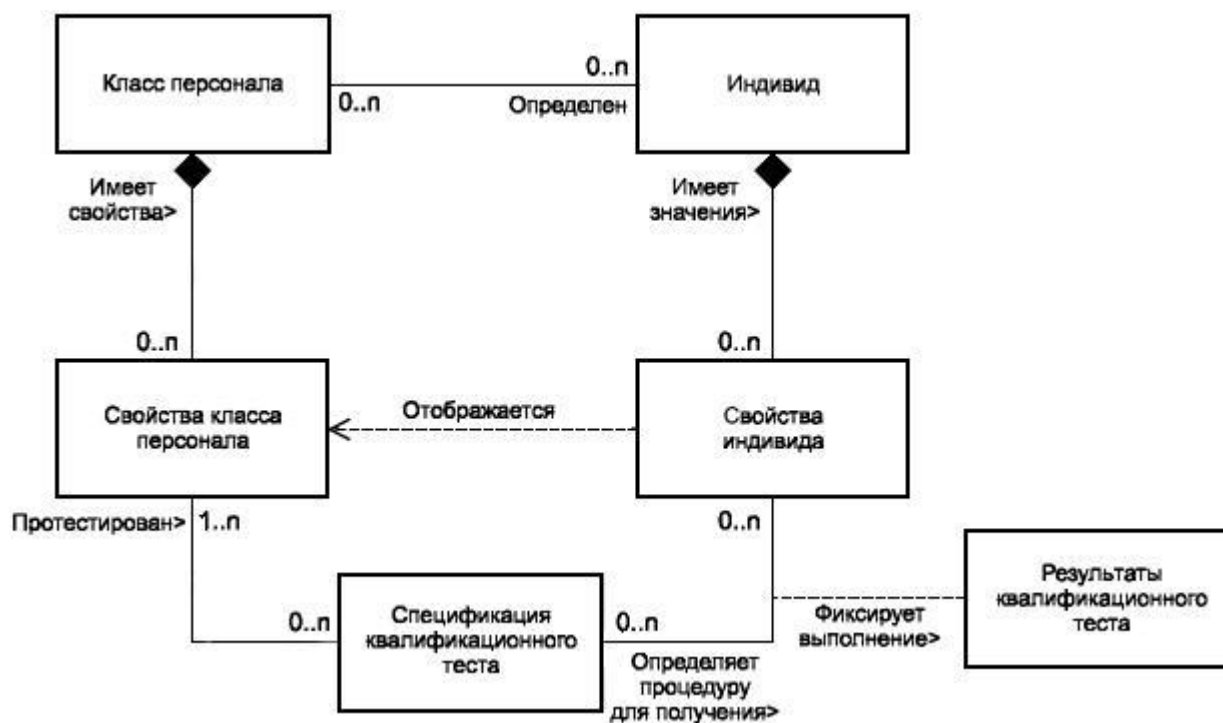
Строка-свойство служит для указания значений свойств, которые могут быть применены к данной операции. Строка-свойство может отсутствовать, если свойства не специфицированы.

17. Объектные модели ресурсов: персонал, оборудование, материал.

Объектные модели представляются с использованием нотации универсального языка моделирования (UML). Нотация языка UML.

Символ	Определение
	Определяет пакет, набор объектных моделей, модели состояний, классы использования и прочие UML-модели. В рамках настоящего стандарта пакет может быть использован для описания внешней модели (например, модели инструкции по изготовлению изделия) или для обращения к другой части модели
	Определяет класс объектов с одинаковыми типами атрибутов. Каждый объект имеет уникальный идентификатор или номер. Для классов не перечисляются операции или методы. Символ "-" перед именем атрибута указывает на то, что данный атрибут обычно необязателен при любом использовании класса
	Отображение связи между элементами некоторого класса и элементами другого или того же самого класса. Каждая связь имеет свой идентификатор и может характеризоваться ожидаемым числом или набором членов подкласса, и тогда 'n' - это промежуточный номер. Например, запись 0,n означает, что могут существовать 0 или более членов подкласса
	Обобщение (стрелка указывает на родительский объект), означающее, что элемент данного класса является специализированным типом указанного суперкласса (надкласса)
	Указывает на зависимость (тесную взаимосвязь) между данным элементом и элементом другого класса
	Агрегация (сборка из элементов) показывает, что элемент определенного класса состоит из элементов других классов
	Композиция, отображающая сильную форму агрегирования, при которой требуется, чтобы экземпляр детали включался одновременно не более чем в один агрегированный объект и чтобы этот объект целиком "отвечал" за компоновку его деталей

Модель персонала содержит информацию о конкретном персонале, категориях персонала и квалификации



Класс персонала – информационное представление группы индивидов с аналогичными характеристиками для целей планирования объемов и календарного планирования производства называется классом персонала.

Характеристики класса персонала – характеристики определенной категории персонала должны рассматриваться как характеристики класса персонала.

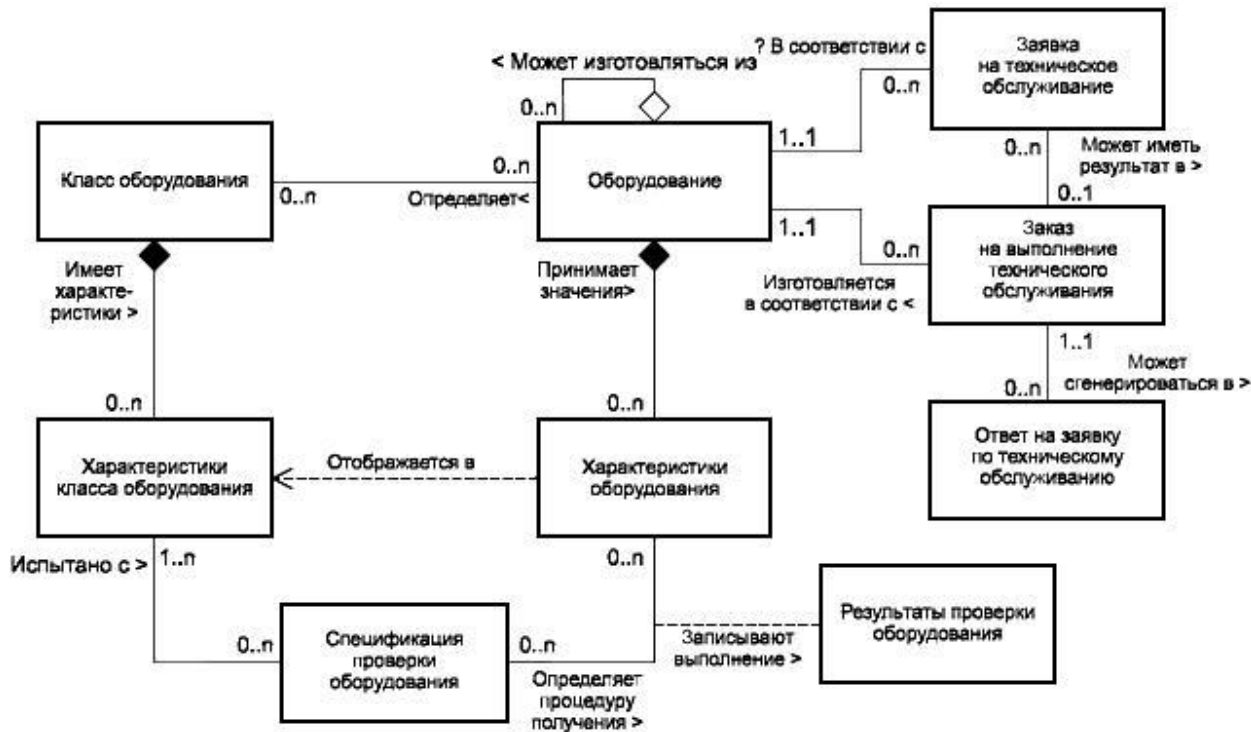
Индивид – конкретно идентифицируемый специалист должен быть представлен в модели как индивид.

Характеристики индивида – качества (параметры) индивида должны представляться списком его характеристик. Их текущие значения определяют степень полезности индивида применительно к данной характеристике.

Спецификация квалификационного теста – представление квалификационного теста должно иметь форму спецификации. Эта спецификация может ассоциироваться как с характеристиками класса персонала, так и с характеристиками индивида. Ее, как правило, используют тогда, когда требуется выполнение квалификационного теста для получения уверенности в том, что данный индивид имеет нужную профессиональную подготовку или практический опыт для выполнения конкретных операций.

Результат квалификационного теста – дату тестирования; результат прохождения тестирования; срок действия аттестации.

А. Модель оборудования содержит информацию о конкретных агрегатах, классах оборудования, тестах для проверки возможностей оборудования, а также сведения о его техническом обслуживании.



Класс оборудования – информационные представления групп оборудования с одинаковыми характеристиками в модели для целей объемного и календарного планирования должны быть использованы как классы оборудования.

Характеристики класса оборудования – свойства класса оборудования должны быть представлены списком его характеристик.

Оборудование – Оборудованием могут быть список производственных площадок, список участков производства, производственных агрегатов, поточных линий, ГПМ, рабочих ячеек или установок.

Характеристики оборудования – характеристики оборудования должны быть представлены списком. Оборудование может иметь нулевое или большее число характеристик, которые определяют текущие значения параметров для соответствующего класса оборудования. В список характеристик оборудования может быть включена единица их измерения.

Спецификация возможностей оборудования – производственные возможности должны быть представлены в модели как спецификация проверки возможностей оборудования, которая может ассоциироваться с какой-либо характеристикой оборудования. Эту спецификацию используют в тех случаях, когда необходимо установить, что конкретное оборудование обладает необходимыми производственными возможностями.

Результат тестирования возможностей оборудования – дату проверки; итог тестирования; конечный срок действия аттестации.

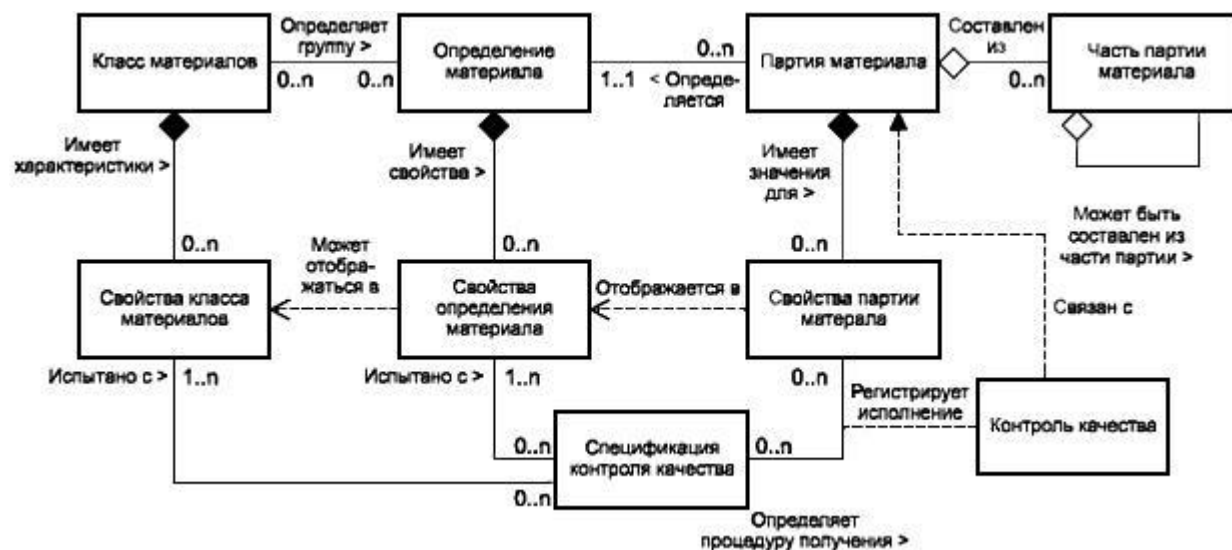
Информация о техническом обслуживании – в пространстве характеристик оборудования информация по управлению технологическим процессом пересекается с информацией по техобслуживанию. К этому пересечению относятся заявки на техническое обслуживание, ответы на заявки и заказы на выполнение работ применительно к конкретному оборудованию.

Запрос технического обслуживания - имя заявителя; дату и время подачи заявки; дату и время выполнения нужных работ; сведения об оборудовании, требующем техобслуживания; суть запроса; приоритет заявки.

Заказ на выполнение техобслуживания: указание конкретного индивида или нужного класса персонала; установленный приоритет заказа; состояние выполняемого заказа.

Ответ на заявку по техобслуживанию: дату и время ответа; имя индивида, ответившего на полученный заказ; текст ответа; результат обработки заказа на выполнение техобслуживания.

В. В модели материалов представляются списки используемых материалов, описания материалов и приводятся сведения, касающиеся определения классов материалов. Информация о материалах охватывает запасы сырья, готовых исходных материалов и полуфабрикатов. Текущие сведения о материалах содержатся в информации о партии или ее части. В целях упорядочения потоков материалов выделяются классы материалов.



Определение материала – модельное представление товаров и изделий с аналогичными характеристиками для целей календарного планирования и планирования объемов имеет форму определения материалов. Материалы могут идентифицироваться как сырье, полуфабрикаты или конечные продукты и могут характеризоваться какой-либо статусной информацией. Определения материалов могут быть связаны также с производственными заявками.

Свойство определения материалов.

Класс материалов – это модельное представление группы определений материалов, выделенное для целей календарного планирования производства и организации процесса обработки.

В характеристиках класса материалов часто приводят список их номинальных или стандартных значений. Характеристика материала необязательно должна соответствовать характеристике класса материалов.

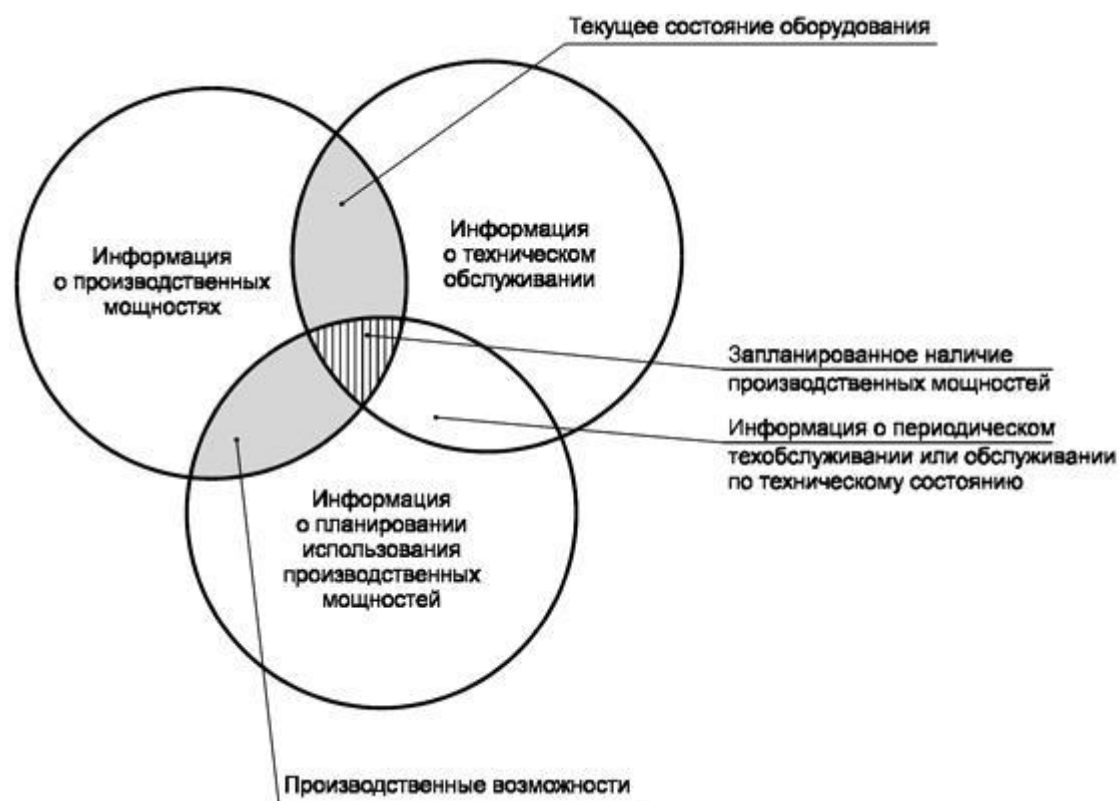
Партия материала - это представление точно определенного количества перечислимого или взвешиваемого материала. Партия характеризуется фактическим или доступным количеством материала, текущим состоянием и конкретными значениями свойств.

Партия материала может храниться отдельно идентифицируемыми частями. Каждая такая часть одной и той же партии материала должна представляться как частичная партия. Все частные партии относятся к одной и той же партии материала, и потому применительно к ним используются значения характеристик элементов партии. В частичной партии может быть только единственный продукт.

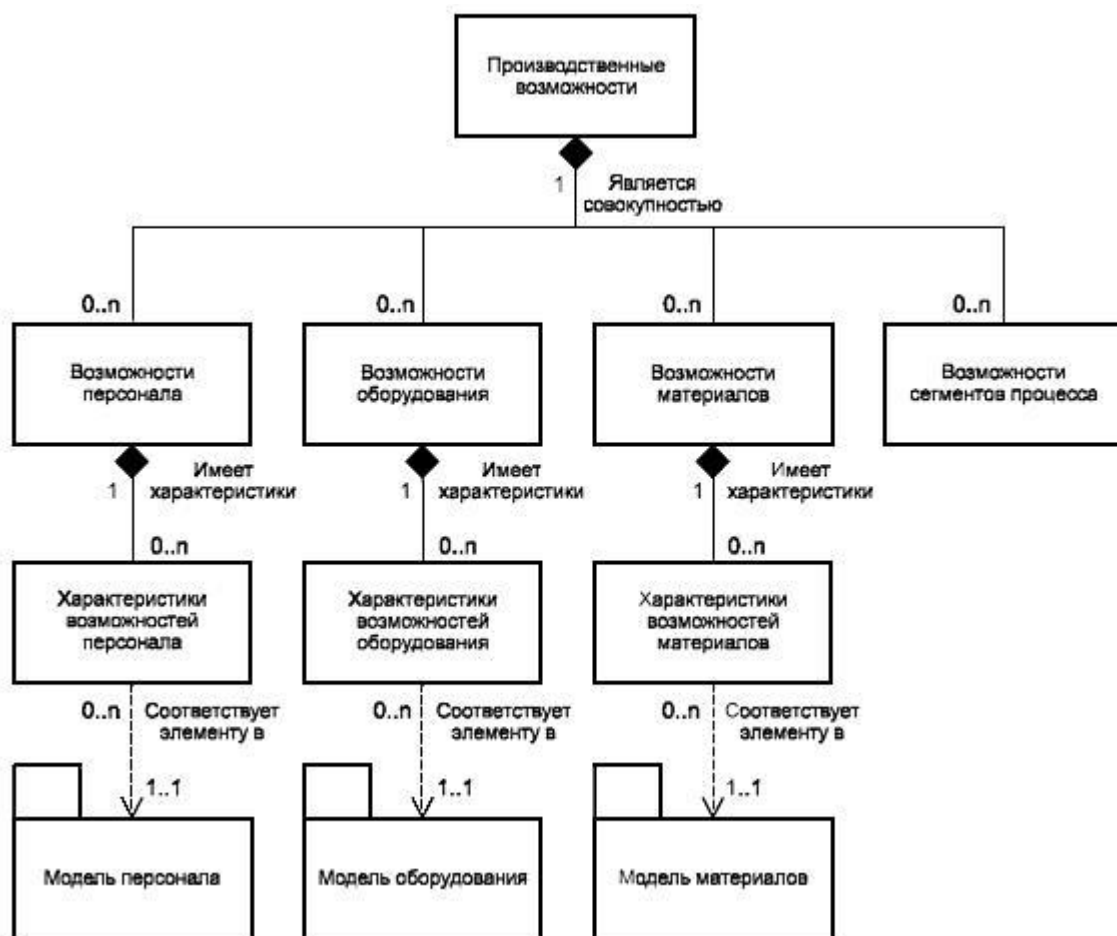
Модельное представление контроля качества должно иметь форму спецификации контроля качества. Эта спецификация может охватывать одно или несколько свойств определений материалов и, как правило, используется тогда, когда требуется проверка качества для получения уверенности в том, что данный материал имеет требуемое производством значение характеристики. Спецификация контроля качества может относиться к одной или нескольким характеристикам материала. Наличие спецификации контроля качества требуется не для всех характеристик.

Итоги проверки качества материалов должны быть представлены как результат выполнения процедуры контроля качества. Результаты контроля качества содержат регистрационные записи по конкретной партии материалов и имеют следующие особенности.

18. Объектные модели возможности производства, Объектные модели графика производства.



Информация о производственных возможностях - это совокупность сведений обо всех ресурсах производства на определенных интервалах времени. Она соответствует области перекрытия, показанной на рисунке, и включает информацию об оборудовании, материалах, персонале и сегментах процесса. Эта информация содержит описания имен, временных параметров, состояний и количественных величин, известных системе управления производством. Информация о производственных возможностях содержит также словарь для информации календарного планирования мощностей и технического обслуживания.



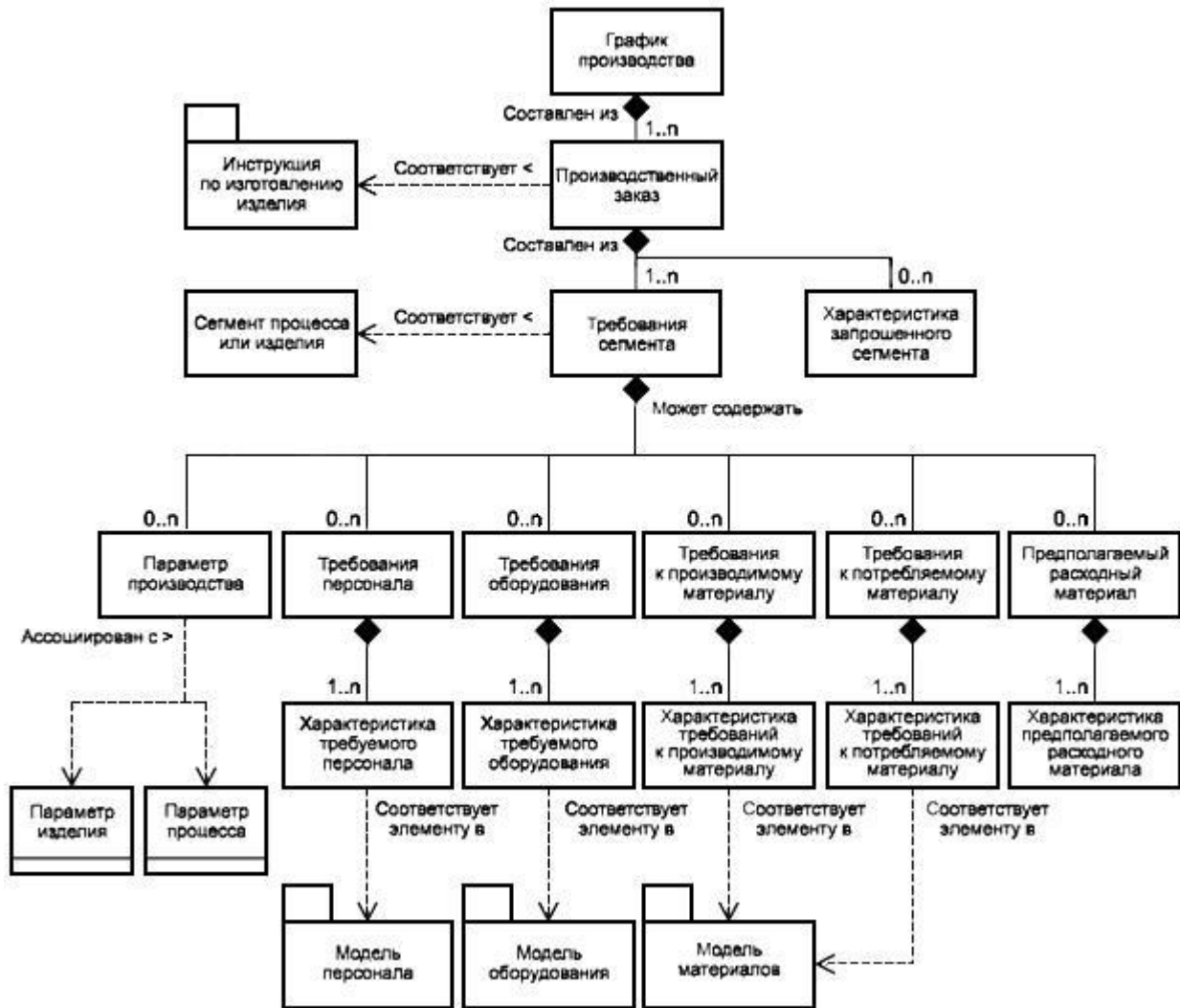
Возможности персонала – это модельное представление производственных возможностей отдельных индивидов и классов персонала, которые используются, доступны или недоступны в течение определенного времени. Возможности персонала могут содержать ссылки как на индивидов, так и на классы персонала. Возможности конкретного персонала должны быть представлены в характеристиках возможностей персонала, которые могут охватывать и сведения о количественных параметрах запрашиваемого ресурса.

Возможности оборудования должны быть представлены как производственные возможности единиц или классов оборудования, которые могут быть задействованы, доступны или недоступны в конкретные промежутки времени. Информация о возможностях конкретного оборудования должна использоваться в определении их характеристик. Эти характеристики могут включать сведения об объемах запрашиваемых ресурсов.

Возможности материалов должны быть представлены как производственные возможности использования материалов, которые могут быть задействованы, доступны или недоступны в конкретные промежутки времени. Возможности использования конкретных материалов рассматриваются применительно к их целостным и частичным партиям. Информация о материальных возможностях включает сведения о функциях контроля использования материалов и энергии и о функциях управления запасами. Постоянно доступными и используемыми материальными ресурсами являются запасы. Незавершенное производство представляет собой материальные возможности, находящиеся под контролем системы управления производством. Конкретные возможности использования материалов должны быть представлены соответствующим списком в характеристиках материальных возможностей. Эти характеристики могут включать сведения об объемах запрашиваемых ресурсов.

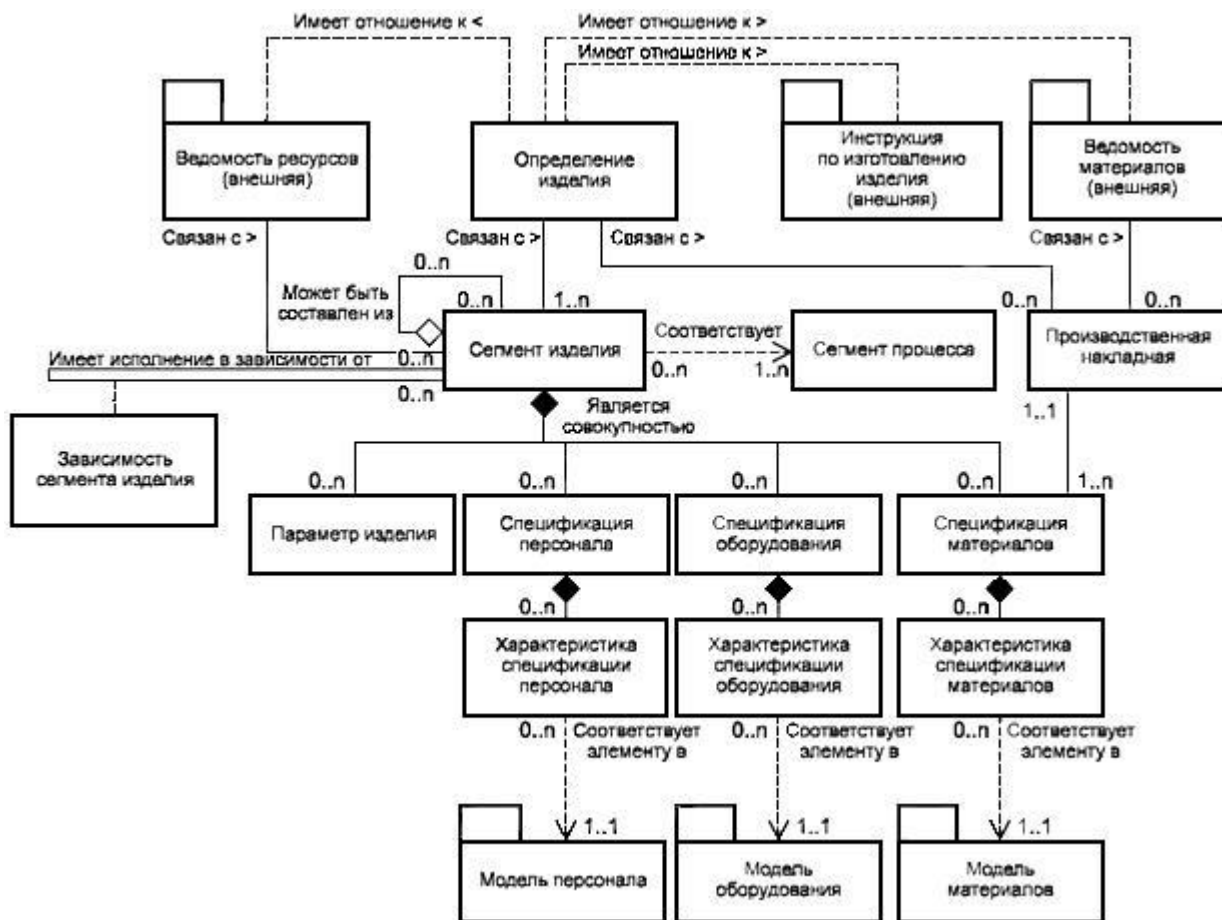
Возможности сегмента процесса должны быть представлены как логические группы ресурсов персонала, оборудования и материалов, которые могут быть задействованы, доступны или недоступны для данного сегмента в конкретные интервалы времени.

Модель производственного графика



19. Объектные модели определения изделия.

Информация определения изделия – это та информация, которая фигурирует одновременно в инструкциях по изготовлению изделий, ведомости материалов и ведомости ресурсов.



Определение изделия содержит листинг расширенной информации об изделии, участвующей в информационном обмене. Эта информация используется во множестве сегментов изделия. Определение изделия содержит ссылки на ведомость материалов, инструкцию по изготовлению изделия и ведомость ресурсов.

Совокупность сегментов изделия, ассоциируемая с каким-либо продуктом, определяет их необходимый набор и порядок следования в процессе изготовления изделия со степенью детализации, достаточной для объемного и календарного планирования производства. Соответствующая инструкция по изготовлению изделия предоставляет дополнительную информацию, требующуюся для реального производства.

Для описания специфических для данного изделия связей могут быть использованы различные зависимости сегмента изделия.

Параметр изделия – определение инструкции по изготовлению изделия выходит за рамки настоящего стандарта, но следует иметь в виду, что она может содержать соответствующее число параметров изделия для каждого сегмента изделия применительно к каждому рассматриваемому продукту. В этих параметрах представляются имена и типы значений, которые могут передаваться в систему управления для задания параметров конкретного изделия.

Спецификация персонала – идентификация возможностей персонала, обращение к ресурсам персонала и определение соответствия возможностей персонала требованиям должны осуществляться с использованием спецификации персонала. Спецификация персонала обычно задает класс персонала, но может определять и конкретного индивида. В спецификации указываются возможности конкретного персонала применительно к данному сегменту изделия.

Спецификация оборудования – идентификация возможностей оборудования, обращение к его ресурсам и определение соответствия возможностей оборудования требованиям должны осуществляться с использованием спецификации оборудования, которая может задавать класс оборудования или конкретную единицу оборудования. В спецификации указываются возможности конкретного оборудования применительно к данному сегменту изделия.

Спецификация материалов – идентификация материальных возможностей или установление их соответствия требованиям должны осуществляться с использованием спецификации материалов. Спецификация материалов задает конкретный материал либо класс материалов применительно к данному сегменту изделия.

20. Формирование требований к ИС.

В подразделе «Требования к системе в целом» указывают:

- требования к структуре и функционированию системы;
- требования к численности и квалификации персонала системы и режиму его работы;
- показатели назначения;
- требования к надежности;
- требования безопасности;
- требования к эргономике и технической эстетике;
- требования к транспортабельности для подвижных АС;
- требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы;
- требования к защите информации от несанкционированного доступа;
- требования по сохранности информации при авариях;
- требования к защите от влияния внешних воздействий;
- требования к патентной чистоте;
- требования по стандартизации и унификации;
- дополнительные требования.

Существенные виды требований:

1. В требованиях к структуре и функционированию системы приводят:

- 1) *перечень подсистем, их назначение и основные характеристики, требования к числу уровней иерархии и степени централизации системы;*
- 2) *требования к способам и средствам связи для информационного обмена между компонентами системы;*
- 3) *требования к режимам функционирования системы;*
- 4) *требования по диагностированию системы;*
- 5) *перспективы развития, модернизации системы.*

- *Перечень подсистем:* производство и предоставление продукции и услуг, ТО продукции, модернизация продукции.
- Подсистемы в процессе функционирования должны обмениваться информацией на основе открытых форматов обмена данными, используя для этого входящие в их состав модули информационного взаимодействия. В состав передаваемых данных входят: сведения о функционировании и использовании продукции, сведения о клиентах и поставщиках, технические данные оборудования
- Режимы функционирования: нормальный режим функционирования; аварийный режим функционирования.

Аварийный режим функционирования системы характеризуется отказом одного или нескольких компонент программного и (или) технического обеспечения.

В случае перехода системы в предаварийный режим необходимо:

- завершить работу всех приложений, с сохранением данных;
- по возможности, перевести нагрузку на альтернативные мощности.

После этого необходимо выполнить комплекс мероприятий по устранению причины перехода системы в аварийный режим.

- Должна предоставлять инструменты диагностирования основных процессов системы, трассировки и мониторинга состояния сетей.
При возникновении аварийных ситуаций, диагностические инструменты должны позволять сохранять данные, находить источник сбоя и предоставлять возможности для его устранения.
- *Должна реализовывать возможность дальнейшей модернизации как программного обеспечения, так комплекса технических средств.*
Также необходимо предусмотреть возможность увеличения производительности системы путем её масштабирования.

2. В требования к надежности включают:

- 1) состав и количественные значения показателей надежности для системы в целом или ее подсистем;
- 2) перечень аварийных ситуаций, по которым должны быть регламентированы требования к надежности, и значения соответствующих показателей;
- 3) требования к надежности технических средств и программного обеспечения;
- 4) требования к методам оценки и контроля показателей надежности на разных стадиях создания системы в соответствии с действующими нормативно-техническими документами.

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении следующих внештатных ситуаций:

- при сбоях в системе электроснабжения аппаратной части, приводящих к перезагрузке ОС, восстановление программы должно происходить после перезапуска ОС и запуска исполняемого файла системы;
- при ошибках в работе аппаратных средств восстановление функции системы возлагается на ОС;
- при ошибках, связанных с программным обеспечением, восстановление работоспособности возлагается на ОС.

3. В требования по безопасности включают требования по обеспечению безопасности при монтаже, наладке, эксплуатации, обслуживании и ремонте технических средств системы, по допустимым уровням освещенности, вибрационных и шумовых нагрузок.

4. В требования к эксплуатации, техническому обслуживанию, ремонту и хранению включают:

- 1) условия и регламент (режим) эксплуатации, которые должны обеспечивать использование технических средств (ТС) системы с заданными техническими показателями, в том числе виды и периодичность обслуживания ТС системы или допустимость работы без обслуживания;
- 2) предварительные требования к допустимым площадям для размещения персонала и ТС системы, к параметрам сетей энергоснабжения и т. п.;
- 3) требования по количеству, квалификации обслуживающего персонала и режимам его работы;
- 4) требования к составу, размещению и условиям хранения комплекта запасных изделий и приборов;
- 5) требования к регламенту обслуживания.

5. В требования к защите информации от несанкционированного доступа

Компоненты подсистемы защиты от НСД должны обеспечивать:

- идентификацию пользователя;
- проверку полномочий пользователя при работе с системой;
- разграничение доступа пользователей на уровне задач и информационных массивов.

6. В требованиях по сохранности информации приводят перечень событий: аварий, отказов технических средств, при которых должна быть обеспечена сохранность информации в системе. Должна быть предусмотрена возможность организации автоматического и (или) ручного резервного копирования данных системы средствами системного и базового программного обеспечения.

7. В требованиях по патентной чистоте указывают перечень стран, в отношении которых должна быть обеспечена патентная чистота системы и ее частей. Установка системы в целом, как и установка отдельных частей системы не должна предъявлять дополнительных требований к покупке лицензий на программное обеспечение сторонних производителей, кроме программного обеспечения

8. В требования к стандартизации и унификации включают: показатели, устанавливающие требуемую степень использования стандартных, унифицированных методов реализации функций (задач) системы, поставляемых программных средств, типовых математических методов и моделей, типовых проектных решений, унифицированных форм управленческих документов, установленных ГОСТ 6.10.1, общесоюзных классификаторов технико-экономической информации и классификаторов других категорий в соответствии с областью их применения, требования к использованию типовых автоматизированных рабочих мест, компонентов и комплексов.

21. Управление требованиями. Цикл формирования требований. Анализ и структурирование первичных требований заказчика.

Под **управлением требованиями** обычно понимается систематический подход к выявлению, документированию, планированию реализации требований и отслеживанию их изменений. Методика призвана регламентировать следующие процедуры работы с требованиями: выявление, документирование, верификация и утверждение требований, планирование реализации и отслеживание изменений требований.

Методика предусматривает две стадии: подготовку управления требованиями и непосредственно управление ими. Процесс подготовки управления требованиями представлен в таблице1. Определение этапов жизненного цикла системы, на которых будет организована работа с требованиями, зависит от выбора его модели. Далее в качестве примера мы будем ориентироваться на ГОСТ 34.601-90 «Автоматизированные системы. Стадии создания», в котором работа с требованиями в рамках процедур, регламентируемых методикой, осуществляется на следующих стадиях: формирование требований к автоматизированной системе, разработка ее концепции, техническое задание и сопровождение.

№	Входная/выходная информация	Шаг процесса	Участник	Бизнес-правила
01	Информация от заинтересованных лиц (заказчики, менеджер проекта)/этапы жизненного цикла системы	Определение этапов жизненного цикла системы, на котором будет организована работа с требованиями	Специалист по управлению требованиями	ГОСТ 34.601-90, ГОСТ 19.102-77, ГОСТ Р ИСО/МЭК 12207-99, РUP
02	Этапы жизненного цикла/шаблоны документов	Подготовка шаблонов документов для каждого этапа	Специалист по управлению требованиями	ГОСТ 34.602-89, ГОСТ 19.201-78, РД 50-34.698-90, РUP
03	Шаблоны документов/коды типов требований	Кодирование типов требований в документах	Специалист по управлению требованиями	ГОСТ 34.602-89, ГОСТ 19.201-78, РД 50-34.698-90, РUP
04	Коды типов требований/атрибуты типов требований	Определение атрибутов типов требований	Специалист по управлению требованиями	Типовые атрибуты согласно РUP
05	Коды типов требований/зависимости между типами требований	Задание зависимостей между типами требований	Специалист по управлению требованиями	Выполняется для каждого конкретного проекта. Шаг процесса можно пропустить
06	Зависимости между типами требований, типы требований с атрибутами, шаблоны документов/План управления требованиями	Разработка плана управления требованиями	Специалист по управлению требованиями	ГОСТ 34.601-90, ГОСТ 19.102-77, ГОСТ 2.503-90, РUP
07	Информация от заинтересованных лиц по процедурам работы с требованиями/процедура выявления требований, анализа, верификации и утверждения	Разработка процедур выявления, верификации, утверждения, изменения требований	Специалист по управлению требованиями	ГОСТ 34.602-89, ГОСТ 19.201-78, РД 50-34.698-90, ГОСТ 2.503-90, РUP
08	ГОСТ 34.602-89, ГОСТ 19.201-78/шаблон модели и его описание	Создание шаблона модели требований и его описание	Специалист по управлению требованиями	ГОСТ 34.602-89, ГОСТ 19.201-78

Подготовка шаблонов документов выполняется для каждого этапа жизненного цикла. Состав документов, оформляемых для выделенных этапов жизненного цикла, приведен в таблице 2. В документах, формируемых с использованием подготовленных шаблонов, будут фиксироваться различные типы требований.

№	Входная/ выходная информация	Шаг процесса	Участник	Бизнес-правила
01	Методы выявления требований, результаты обследования объекта автоматизации, план управления требованиями/список выявленных требований	Выявление требований и их структурирование	Системный аналитик	Процедуры выявления требований
02	Список выявленных требований/модель требований и ее описание	Моделирование требований	Системный аналитик	Шаблон модели требований и его описание
03	Список выявленных требований/документы с требованиями	Документирование требований	Системный аналитик, Специалист по требованиям	Шаблоны документов
04	Документы с требованиями, модель требований и ее описание/документы и модели с требованиями верифицированные и утвержденные	Верификация и утверждение требований	Эксперт предметной области, рецензент	Процедура верификации и утверждения требований
05	Документы и модели с требованиями верифицированные и утвержденные /базовая версия требований	Определение базовой версии требований	Архитектор	Процедура верификации и утверждения требований (раздел, посвященный определению базовой версии требований)
06	Базовая версия требований/измененные требования	Изменение требований	Все заинтересованные лица	Процедура управления изменениями

Далее выполняется кодирование типов требований в документах. Назначенные коды типов требований используются при определении их атрибутов и при задании зависимостей между ними. Для каждого типа требований должен быть разработан атрибутивный состав, позволяющий описать данное требование, а также дать возможность проектной группе оценить его с точки зрения ключевых аспектов разработки: объем работ, стабильность требования, влияние на архитектуру, приоритетность. Для большинства требований можно выделить общий блок атрибутов. При необходимости дополнительные атрибуты могут быть назначены индивидуально для каждого типа требования.

Следующим шагом процесса является разработка основных процедур управления требованиями: процедура выявления требований, процедура верификации требований, процедура изменения требований. При этом учитывается информация, поступившая от заинтересованных лиц по данным процедурам, и требования ГОСТ 34.602-89, ГОСТ 19. 201-78. В заключение создается шаблон модели требований и его описание. В качестве опорных документов используются ГОСТ 34.602-89, ГОСТ 19.201-78.

Цикл формирования требований. Анализ и структурирование первичных требований заказчика.

Проводится предварительный анализ целесообразности разработки ИС. формирование требований к ИС, корректно и точно отражающих цели и задачи организации-заказчика, оценка требуемых ресурсов, и возможных рисков.

Основная задача - оценка реального объема проекта, его целей и задач, а также получение определений сущностей и функций на высоком уровне. По завершении основной стадии обследования системы технические специалисты формируют вероятные технические подходы и приблизительно рассчитывают затраты на аппаратное обеспечение, закупаемое программное обеспечение и разработку нового программного обеспечения.

Результатом этапа является документ, где четко сформулировано, что получит заказчик, если согласится финансировать проект; когда он получит готовый продукт (график выполнения работ); сколько это будет стоить (для крупных проектов должен быть составлен график финансирования на разных этапах работ). В документе должны быть отражены не только затраты, но и выгода, например время окупаемости проекта, ожидаемый экономический эффект (если его удастся оценить).

В документе обязательно должны быть описаны:

- ограничения, риски, критические факторы, влияющие на успешность проекта, например время реакции системы на запрос является заданным ограничением, а не желательным фактором;

- совокупность условий, при которых предполагается эксплуатировать будущую систему: архитектура системы, аппаратные и программные ресурсы, предоставляемые системе, внешние условия ее функционирования, состав людей и работ, которые обеспечивают бесперебойное функционирование системы;
- сроки завершения отдельных этапов, форма сдачи работ, ресурсы, привлекаемые в процессе разработки проекта, меры по защите информации;
- описание выполняемых системой функций;
- будущие требования к системе в случае ее развития, например, возможность работы пользователя с системой с помощью Интернета и т.п.;
- интерфейсы и распределение функций между человеком и системой;
- требования к программным и информационным компонентам ПО, требования к СУБД;
- что не будет реализовано в рамках проекта.

Выполненная на данном этапе работа позволяет ответить на вопрос, стоит ли продолжать данный проект и какие требования заказчика могут быть удовлетворены при тех или иных условиях. Может оказаться, что проект продолжать не имеет смысла, например из-за того, что те или иные требования не могут быть удовлетворены по каким-то объективным причинам. Если принимается решение о продолжении проекта, то проводится анализ.

Анализ предполагает подробное исследование бизнес-процессов (функций, определенных ранее) и информации, необходимой для их выполнения (сущностей, их атрибутов и связей (отношений)). На этом этапе создается информационная модель, а на следующем за ним этапе проектирования - модель данных.

Вся информация о системе, собранная ранее, формализуется и уточняется во время анализа. Как правило, заказчик не сразу формирует требования к системе в целом, а формулирует требования к отдельным ее компонентам.

Аналитики собирают и фиксируют информацию в двух взаимосвязанных формах:

- функции - информация о событиях и процессах, которые происходят в бизнесе;
- сущности - информация о вещах, имеющих значение для организации и о которых что-то известно.

Двумя классическими результатами анализа являются:

- иерархия функций, которая разбивает процесс обработки на составные части (что делается и из чего это состоит);
- модель "сущность-связь" (ER-модель), которая описывает сущности, их атрибуты и связи (отношения) между ними.

При проектировании информационной системы важно учесть и дальнейшее развитие системы, например рост объемов обрабатываемых данных, увеличение интенсивности потока запросов, изменение требований надежности информационной системы.

Во время анализа выбор тех или иных СУБД, аппаратной платформы и операционной системы. Так же уточняется выбор средств разработки. Осуществляя выбор тех или иных средств, следует учитывать наличие высококвалифицированного персонала, который владеет выбранными средствами разработки, а также наличие администраторов выбранной СУБД. Уточняются также ограничения, риски, критические факторы. Более подробно описываются те возможности, которые не будут реализованы в системе.

Задача формирования требований к ИС является одной из наиболее ответственных, трудно формализуемых и наиболее дорогих и тяжелых для исправления в случае ошибки.

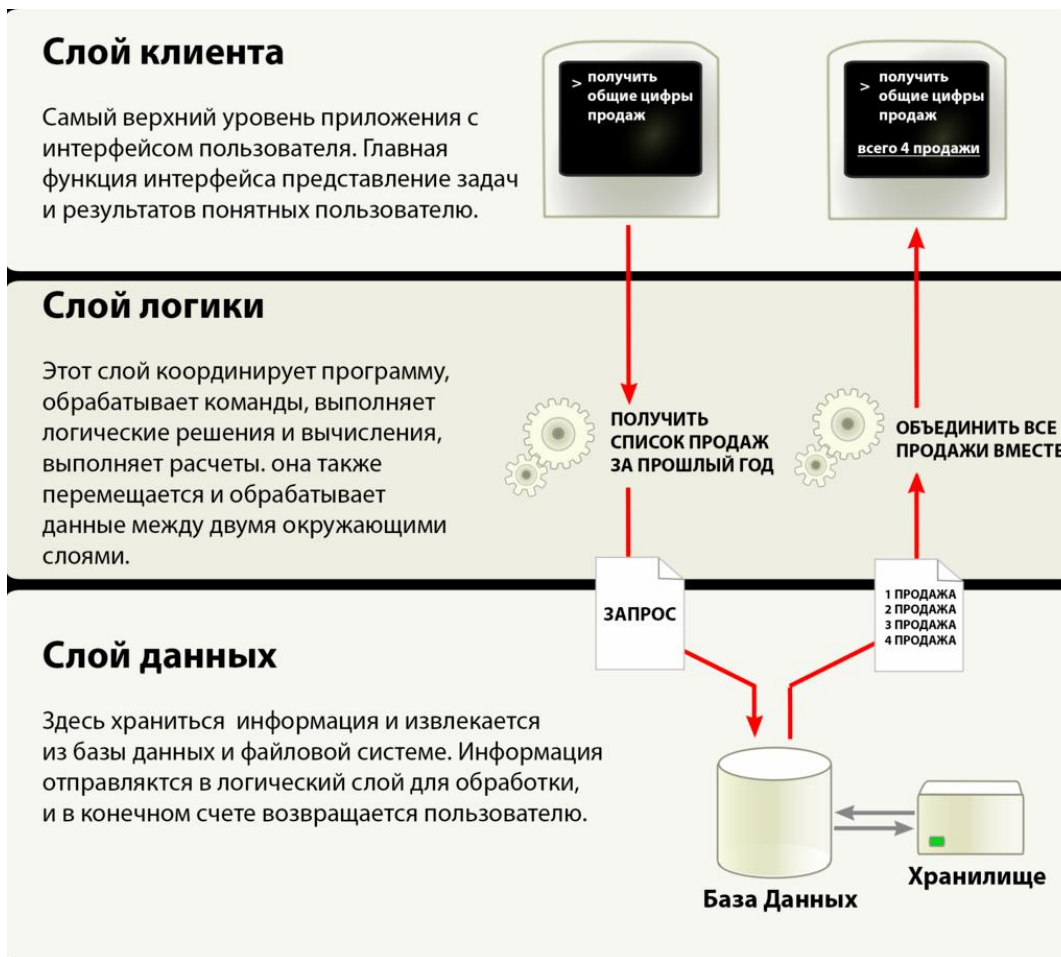
В качестве ресурсов рассматриваются:

- время на разработку ИС

- финансы
- персонал
- технические и программные средства
- другие ресурсы

22. Трехуровневая модель построение ИС.

- Уровень данных – серв. БД – целостность
- Уровень бизнес-логики – серв. приложение – модульность ис(возможность частичной или последовательной автоматизации, повторное использование кода), тестир. рез.
- Уровень представления(интерфейсный) – клиент – визуальное представление



23. Особенности формирования модели данных. Виды СУБД.

Разработка модели системы в общем случае означает объединение отдельных составляющих системы в единую модель. При этом каждая составляющая (подсистема) изолирована от других частей модели и решает собственные задачи. Количество связей между элементами системы (подсистемы), в том числе с внутренней и внешней средой, может быть значительным.

Структура системы – это совокупность связей между элементами, отражающими их взаимодействие. Структура определяет алгоритм выборки отдельных элементов данных, она же отражает и особенности организации и обработки информации, свойственные человеку в его повседневной деятельности.

Для создания структурированных и слабоструктурированных представлений предметной области и выполнения операций в терминах таких представлений служат разнообразные, реализованные в программных средствах, инструменты моделирования, которые называют моделями данных.

Модель данных – это не результат, а инструмент моделирования, то есть совокупность правил структурирования данных, допустимых операций над ними и видов ограничений целостности, которым они должны удовлетворять.

Информационная модель – это модель объекта, процесса или явления, в которой представлены информационные аспекты моделируемого объекта, процесса или явления.

Формирование и исследование моделей осуществляется с помощью различных методов моделирования. Процессы моделирования всё чаще осуществляются с использованием специальных компьютерных программных средств, позволяющих автоматизировать эту деятельность.

Система управления базами данных (СУБД) — совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

Основные функции СУБД:

- управление данными во внешней памяти (на дисках);
- управление данными в оперативной памяти с использованием дискового кэша;
- журнализация изменений, резервное копирование и восстановление базы данных после сбоев;
- поддержка языков БД (язык определения данных, язык манипулирования данными).

Виды СУБД:

- полнофункциональные СУБД;
- серверы БД;
- клиенты БД;
- средства разработки программ работы с БД.

Полнофункциональные СУБД (ПФСУБД) представляют собой традиционные СУБД, которые сначала появились для больших машин, затем для мини-машин и для ПЭВМ. Это самая многочисленная группа СУБД, обладающих большими возможностями.

ПФСУБД последнего поколения имеют развитый интерфейс, позволяющий с помощью команд меню выполнять основные действия с БД: создавать и модифицировать структуры таблиц, вводить данные, формировать запросы, разрабатывать отчеты, выводить их на печать и т. п. Во многих из них для

создания запросов можно пользоваться языком QBE (Query By Example). Многие ПФСУБД включают средства программирования для профессиональных разработчиков.

Серверы БД предназначены для организации центров обработки данных в сетях ЭВМ. Эта группа СУБД в настоящее время менее многочисленна, но их количество постепенно растет. Серверы БД реализуют функции управления базами данных, запрашиваемые другими (клиентскими) программами обычно с помощью операторов SQL. Примерами серверов БД являются следующие программы: NetWare SQL (Novell), SQL Server (Microsoft), InterBase (Borland) и др.

В роли **клиентских программ** для серверов БД в общем случае могут использоваться различные программы: ПФСУБД, электронные таблицы, текстовые процессоры, программы электронной почты и т. п. При этом элементы пары «клиент — сервер» могут принадлежать одному или разным производителям программного обеспечения. В случае, когда клиентская и серверная части выполнены одной фирмой, естественно ожидать, что распределение функций между ними выполнено рационально. В остальных случаях обычно преследуется цель обеспечения доступа к данным «любой ценой».

Средства разработки программ работы с БД могут использоваться для создания разновидностей следующих программ:

- клиентских программ;
- серверов БД и их отдельных компонентов;
- пользовательских приложений.

Программы первого и второго вида весьма малочисленны, так как предназначены, главным образом, для системных программистов. Пакетов третьего вида гораздо больше, но меньше, чем полнофункциональных СУБД.

По характеру использования СУБД делят на персональные и многопользовательские.

Персональные СУБД обычно обеспечивают возможность создания персональных БД и недорогих приложений, работающих с ними. Персональные СУБД или разработанные с их помощью приложения зачастую могут выступать в роли клиентской части многопользовательской СУБД. К персональным СУБД, например, относятся Visual FoxPro, Paradox, Clipper, dBase, Access и др.

Многопользовательские СУБД включают в себя сервер БД и клиентскую часть и, как правило, могут работать в неоднородной вычислительной среде (с разными типами ЭВМ и операционными системами). К многопользовательским СУБД относятся, например, СУБД Oracle и Informix.

По используемой модели данных СУБД (как и БД), разделяют на иерархические, сетевые, реляционные, объектно-ориентированные и другие типы. Некоторые СУБД могут одновременно поддерживать несколько моделей данных.

3 уровень	рабочие места серверные системы (обеспечивают безопасность хранения данных, целостность)
2 уровень	операторский уровень автоматный уровень
1 уровень	датчики, исп. мех.

24. Модели описания информационной архитектуры. ER-модели, UML, IDEF1x

ER-модель (entity-relationship model) - модель "Сущность-связь".

Базовые понятия

Сущность (объект)	 Сотрудник
Атрибут сущности (свойство, характеризующее объект)	 ФИО
Ключевой атрибут (атрибут, входящий в первичный ключ)	 Номер_сотр
Связь	 Работает

Первичный ключ - атрибут или группа атрибутов, однозначно идентифицирующих объект. Первичный ключ может состоять из нескольких атрибутов, тогда подчеркивается каждый из них.

Связи между объектами могут быть 3-х типов:

- *Один - к одному*. Этот тип связи означает, что каждому объекту первого вида соответствует не более одного объекта второго вида, и наоборот.
- *Один - ко многим*. Этот тип связи означает, что каждому объекту первого вида может соответствовать более одного объекта второго вида, но каждому объекту второго вида соответствует не более одного объекта первого вида.
- *Многие - ко многим*. Этот тип связи означает, что каждому объекту первого вида может соответствовать более одного объекта второго вида, и наоборот.

Ромб связи и прямоугольник объекта соединяются ненаправленными дугами в сторону "ко многим" и направленными в сторону "к одному".

UML (Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это — открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем.

IDEF1X является методом для разработки реляционных баз данных и использует условный синтаксис, специально разработанный для удобного построения концептуальной схемы. Концептуальной схемой наз. универсальное представление структуры данных в рамках коммерческого предприятия, независимое от конечной реализации базы данных и аппаратной платформы.

Использование метода IDEF1X наиболее целесообразно для построения логической структуры базы данных после того, как все информационные ресурсы исследованы и решение о внедрении реляционной базы данных, как части корпоративной информационной системы, было принято.

Сущность в IDEF1X описывает собой совокупность или набор экземпляров похожих по свойствам, но однозначно отличаемых друг от друга по одному или нескольким признакам. Каждый экземпляр является реализацией сущности. Связи в IDEF1X представляют собой ссылки, соединения и ассоциации между сущностями.

25. Модульный принцип построения ИС.

Модульный принцип построения подразумевает, что состав программных или технических средств любой АИС представлен в виде отдельных относительно автономных блоков. Подобное построение АИС обеспечивает возможность оперативной замены или изменения ее составных частей с целью совершенствования функционирования указанной системы.

26. Описание программной архитектуры. Шаблоны проектирования.

Шаблон – многократно применяемая архитектурная конструкция – pattern

- *проектирования

- *архитектуры

- *проектирования отдельных классов и объектов

- *интегрирования ИС

- + использование паттернов унифицирует взаимодействие отдельных разработчиков, т.к. индивидуальные проектные решения – одинаковые

- + возможность повторного применения решений и их согласованность

- + ускорение проектирования и разработки

- шаблонные решения заставляют разработчика усложнять систему

- следование шаблонам ограничивает поиск новых решений

- вероятность неправильного применения шаблонов

Pattern может быть реализован на любом языке программирования в любой среде.

27. Интерфейс пользователя ИС. Виды интерфейсов.

Интерфейс – набор правил для взаимодействия между программами и оборудованием, человеком. Совокупность методов для взаимодействия между элементами систем.

С т. зр. ИС – способ взаимодействия компонентов ИС.

Виды интерфейсов:

1. Физический интерфейс.

Физический (аппаратный) интерфейс — способ взаимодействия физических устройств (устройство, преобразующее сигналы и передающее их от одного компонента оборудования к другому).

Примеры: сетевой интерфейс - точку соединения между компьютером пользователя и частной или общественной сетью; сетевая карта компьютера; точка соединения двух сетей между собой. Компьютерная шина – в архитектуре компьютера подсистема, которая передаёт данные между функциональными блоками компьютера.

2. Программный интерфейс

Программный интерфейс — функциональность, которую некоторый программный компонент предоставляет другим программным компонентам. Имеют полную формализованную спецификацию. Обеспечивают возможность независимых модификаций ИС и возможность доступа по этому интерфейсу другой системы. Могут быть как внешними, так и внутренними. Примеры:

Интерфейс программирования приложений (API) - набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) для использования во внешних программных продуктах.

Используется программистами для написания всевозможных приложений.

Интерфейс объектно-ориентированного программирования - семантическая и синтаксическая конструкция в коде программы, используемая для специфицирования услуг, предоставляемых классом или компонентом.

3. Пользовательский интерфейс

Пользовательский интерфейс - совокупность средств, при помощи которых пользователь взаимодействует с различными программами и устройствами.

- Интерфейс командной строки: инструкции компьютеру даются путём ввода с клавиатуры текстовых строк (команд).
- Графический интерфейс пользователя: программные функции представляются графическими элементами экрана.
- Диалоговый интерфейс: например, Поиск
- Жестовый интерфейс: сенсорный экран, руль, джойстик и т. д.

28. Особенности проектирования пользовательских интерфейсов. Подход психологии восприятия. Подход инженерной психологии. Подход когнитивной психологии.

Интерфейс — набор правил/процедур для взаимодействия между программами, программами и оборудованием, программами и пользователем (между элементами ИС)

В основе создания пользовательских интерфейсов:

- 1) психология восприятия
- 2) инженерная психология
- 3) теория деятельности
- 4) когнитивная психология

Инженерная психология:

4 основных критерия качества

- скорость работы пользователя
- количество ошибок
- скорость обучения
- удовлетворение

Основная идея когнитивной психологии — человек рассматривается как ИС

29. Особенности интеграции ИС. Программные интерфейсы.

Проблемы, связанные с интеграцией

Интеграция требует значительных изменений в корпоративной политике. Бизнес приложения обычно направлены на определённую функциональную область, такую как управление взаимоотношениями с клиентами, финансы. А

для проведения успешной интеграции необходимо наладить взаимодействие не только между различными компьютерными системами, но также между производственными модулями и отделами. Таким образом, конкретный отдел больше не контролирует какое-то приложение, так как оно становится частью общей совокупности интегрированных приложений и служб.

Из-за своего масштаба, интеграция имеет значительные последствия для предприятия. Когда обработка наиболее важных функций предприятия добавляется в интегрированную систему, нормальное функционирование интегрированной системы становится для предприятия жизненно важным. Сбой или ошибка интегрированной системы может стоить предприятию значительную сумму.

Одним из ограничений разработки интегрированного решения является то, что разработчики недостаточно могут управлять приложениями-частями интегрированного решения. В большинстве случаев такие приложения - это унаследованные системы или поставляемые пакетами приложения, которые нельзя изменить, чтобы подключить к интегрированному решению.

Сложна не только разработка EAI (Интеграция приложений предприятия (Enterprise Application Integration)) решения, работа с таким решением и его поддержка могут стать даже более сложной задачей. Смесь технологий и распределённый характер EAI решений делают развёртывание, мониторинг и исправление неисправностей сложными задачами, которые требуют весьма разносторонней квалификации.

Способы интеграции корпоративных приложений:

- Интеграция "каждый с каждым". (Фиксация интерфейсов и форматов данных с двух сторон). Это традиционный подход к интеграции прикладных программ. Он состоит в создании специализированных *интерфейсов обмена данными* для каждой пары обменивающихся приложений. Такой подход хорош для небольшого числа приложений.
- Интеграция на уровне пользовательских интерфейсов.
- Интеграция на уровне данных - то есть несколько приложений могут обращаться в одну базу данных или в несколько баз данных, связанных репликациями. Поддержка данных в специальных хранилищах независимо от породившей их бизнес-логики. Доступ к хранилищам могут получать различные приложения. При этом подходе важное значение принимает наличие хорошо документированной (и редко изменяющейся) модели данных.

Программный интерфейс — функциональность, которую некоторый программный компонент предоставляет другим программным компонентам. Имеют полную формализованную спецификацию. Обеспечивают возможность независимых модификаций ИС и возможность доступа по этому интерфейсу другой системы. Могут быть как внешними, так и внутренними. Примеры:

Интерфейс программирования приложений (API) - набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) для использования во внешних программных продуктах. Используется программистами для написания всевозможных приложений.

Интерфейс объектно-ориентированного программирования - семантическая и синтаксическая конструкция в коде программы, используемая для специфицирования услуг, предоставляемых классом или компонентом.

30. Особенности ввода в эксплуатацию ИС. Сопровождение внедрения ИС.

Ввод КИС в эксплуатацию:

- Подготовка технических и системных программных средств;
- Установка программного обеспечения КИС на сервер и перенос настроек из Макета в БД;
- Настройка пользовательских рабочих мест;
- Разработка программных средств импорта данных, необходимых для старта Системы (если необходимо);
- Разработка и отладка программных интерфейсов взаимодействия КИС с другими программными системами в соответствии с требованиями технического задания на их разработку (если необходимо);

- Разработка программных конвертеров для переноса нормативно-справочной информации (НСИ) и начальных данных необходимых для старта КИС (если необходимо);
- Подготовка исходных данных, ввод их в Систему и проверка их корректности;
- Проведение подготовки пользователей Системы на курсах консультационных семинаров и в виде инструктажа на рабочих местах.

Методическое сопровождение опытной эксплуатации:

- Контроль ввода пользователями реальных данных за установленный контрольный период, правильности обработки данных и сверки результатов;
- Методическое сопровождение пользователей консультантами по вопросам эксплуатации Системы путём проведения консультирования на рабочем месте;
- Локализация проблем и поиск путей их решения, устранение обнаруженных недоработок;
- Уточнение и доработка регламентов и пользовательских инструкций;
- Анализ результатов опытной эксплуатации;
- Администрирование и техническая поддержка.

31. Требования к инфраструктуре ИС.

КИС должны отвечать целому набору обязательных требований:

1. Среди них, в первую очередь, стоит отметить использование архитектуры клиент-сервер с возможностью применения большинства промышленных СУБД
2. Поддержку распределенной обработки информации
3. Модульный принцип построения из оперативно-независимых функциональных блоков с расширением за счет открытых стандартов (API, COM+, CORBA и другие)
4. Обеспечивать поддержку технологий Internet/intranet.
5. Гибкость

Гибкость, способность к адаптации и дальнейшему развитию подразумевают возможность приспособления информационной системы к новым условиям, новым потребностям предприятия. Выполнение этих условий возможно, если на этапе разработки информационной системы использовались общепринятые средства и методы документирования, так что по прошествии определенного времени сохраняется возможность разобраться в структуре системы и внести в нее соответствующие изменения, даже если все разработчики или их часть по каким-либо причинам не смогут продолжить работу.

Следует иметь в виду, что психологически легче разобраться в собственных разработках, пусть даже созданных давно, чем в чужих решениях, не всегда на первый взгляд логичных. Поэтому рекомендуется фазу сопровождения системы доверять лицам, которые ее проектировали.

Любая информационная система рано или поздно морально устареет, и станет вопрос о ее модернизации или полной замене. Разработчики информационных систем, как правило, не являются специалистами в прикладной области, для которой разрабатывается система. Участие в модернизации или создании новой системы той же группы проектировщиков существенно сократит сроки модернизации.

Вместе с тем возникает риск применения устаревших решений при модернизации системы. Рекомендация в таком случае одна — внимательнее относиться к подбору разработчиков информационных систем.

6. Надежность

Надежность информационной системы подразумевает ее функционирование без искажения информации, потери данных по «техническим причинам». Требование надежности обеспечивается созданием резервных копий хранимой информации, выполнения операций протоколирования, поддержанием качества каналов связи¹ и физических носителей информации, использованием современных программных и аппаратных средств. Сюда же следует отнести защиту от случайных потерь информации в силу недостаточной квалификации персонала.

7. Эффективность

Система является эффективной, если с учетом выделенных ей ресурсов она позволяет решать возложенные на нее задачи в минимальные сроки.

В любом случае оценка эффективности будет производиться заказчиком, исходя из вложенных в разработку средств и соответствия представленной информационной системы его ожиданиям.

Негативной оценки эффективности информационной системы со стороны заказчика можно избежать, если представители заказчика будут привлекаться к проектированию системы на всех его стадиях. Такой подход позволяет многим конечным пользователям уже на этапе проектирования адаптироваться к изменениям условий работы, которые иначе были бы приняты враждебно.

Активное сотрудничество с заказчиком с ранних этапов проектирования позволяет уточнить потребности заказчика. Часто встречается ситуация, когда заказчик чего-то хочет, но сам не знает чего именно. Чем раньше будут учтены дополнения заказчика, тем с меньшими затратами и в более короткие сроки система будет создана.

Кроме того, заказчик, не являясь специалистом в области разработки информационных систем, может не знать о новых информационных технологиях. Контакты с заказчиком во время разработки для него информационной системы могут подтолкнуть заказчика к модернизации его аппаратных средств, применению новых методов ведения бизнеса, что отвечает потребностям как заказчика, так и проектировщика. Заказчик получает рост эффективности своего предприятия, проектировщик — расширение возможностей, применяемых при проектировании информационной системы.

Эффективность системы обеспечивается оптимизацией данных и методов их обработки, применением оригинальных разработок, идей, методов проектирования (в частности, спиральной модели проектирования информационной системы, о которой речь пойдет в следующих главах).

Не следует забывать и о том, что работать с системой придется обычным людям, являющимся специалистами в своей предметной области, но зачастую обладающим весьма средними навыками в работе с компьютерами. Интерфейс информационных систем должен быть им интуитивно понятен. В свою очередь, разработчик-программист должен понимать характер выполняемых конечным пользователем операций. Рекомендациями в этом случае могут служить повышение эффективности управления разработкой информационных систем, улучшение информированности разработчиков о предметной области.

Имеет смысл еще до сдачи информационной системы в эксплуатацию предоставить разработчикам возможность попробовать себя в роли конечных пользователей. Встречались случаи, когда такой подход приводил к отказу от использования на рабочем месте оператора манипулятора типа «мышь», что, в свою очередь, приводило к многократному повышению производительности оператора.

8. Безопасность

Под безопасностью, прежде всего, подразумевается свойство системы, в силу которого посторонние лица не имеют доступа к информационным ресурсам организации, кроме тех, которые для них предназначены, что достигается с помощью различных методов контроля и разграничения доступа к информационным ресурсам.

Защита информации от постороннего доступа обеспечивается управлением доступом к ресурсам системы, использованием современных программных средств защиты информации. В крупных организациях целесообразно создавать подразделения, основным направлением деятельности которых было бы обеспечение информационной безопасности, в менее крупных организациях назначать сотрудника, ответственного за данный участок работы. Система, не отвечающая требованиям безопасности, может причинить ущерб интересам заказчика, прежде всего имущественным.

В этой связи следует отметить, что согласно действующему в России законодательству ответственность за вред, причиненный ненадлежащим качеством работ или услуг, несет исполнитель, то есть в нашем случае разработчик информационной системы. Поэтому ненадлежащее обеспечение безопасности информационной системы заказчика в худшем случае обернется для исполнителя судебным преследованием, в лучшем — потерей клиента и утратой деловой репутации.

Помимо злого умысла, при обеспечении безопасности информационных систем приходится сталкиваться еще с несколькими факторами. В частности, современные информационные системы являются достаточно сложными

программными продуктами. При их проектировании с высокой вероятностью возможны ошибки, вызванные большим объемом программного кода, несовершенством компиляторов, человеческим фактором, несовместимостью с используемыми программами сторонних разработчиков в случае модификации этих программ и т. п. Поэтому за фазой разработки информационной системы неизбежно следует фаза ее сопровождения в процессе эксплуатации, в которой происходит выявление скрытых ошибок и их исправление. Например, при проектировании информационной системы курс доллара США в одной из процедур разработки обозначили константой. На момент ввода в эксплуатацию этой системы курс доллара был стабилен, поэтому ошибка никак себя не проявляла, а была выявлена только через некоторое время в период роста курса. Требование безопасности обеспечивается современными средствами разработки информационных систем, современной аппаратурой, методами защиты информации, применением паролей и протоколированием, постоянным мониторингом состояния безопасности операционных систем и средств их защиты. И наконец, самый важный фактор, влияющий на процесс разработки, — знания и опыт коллектива разработчиков информационных систем.

32. Особенности выбора прототипа ИС.

Прототипы бывают:

- элементные (решают типовые задачи)
- подсистемные (объединяют несколько элементных в некоторый подсистемный)
- отраслевые (направлены на конкретную сферу деятельности)

Прототип КИС – тиражируемое или так называемое “коробочное” решение, требующее доработки при внедрении, т.к. бизнес-процессы каждого предприятия уникальны.

Критерии выбора прототипа:

- 1) Максимальное соответствие логики функционирования готовых подсистем реальному функционированию объекта автоматизации. (При выборе КИС должны быть сформулированы основные требования к ней: какие функциональные области и типы производства она должна охватывать, какие программно-технические платформы использовать, какие отчеты готовить и др.)
- 2) Себестоимость информационной системы (стоимость самой системы, её доработки, стоимость инфраструктуры; стоимость сопровождения системы)
- 3) Преемственность информационной системы – насколько совместима с уже существующими системами или является их новой версией (одной единой КИС на предприятии не бывает, нужна интеграция).
- 4) Индивидуальные предпочтения руководителя.
- 5) Гибкость и открытость. Является одним из важнейших факторов выбора КИС. Система управления предприятием должна меняться вместе с производством. Она должна позволять легко менять АРМы, менять бизнес-процессы и алгоритмы путем параметрической настройки и др. Система должна легко настраиваться и интегрироваться с другими модулями или системами.
- 6) Качество локализации западной системы (если система западная).
- 7) Отношение фирмы к продукту, который она предлагает. (Поставщик должен не просто поставить КИС компании-клиенту, а стать ее долговременным партнером и нести различные виды ответственности за качество и эффективность функционирования установленного и сопровождаемого ПО.)

33. Особенности оценки качества ПО. Подходы к организации оценки качества ИС.

Особенности оценки качества ИС.

Проблемами оценки качества промышленной продукции занимается квалиметрия –

наука, определяющая основные термины в области качества продукции, устанавливающая группы показателей качества, определяющая методы получения количественных

значений показателей качества и методы определения уровня качества различных образцов промышленной продукции.

Для рассмотрения проблемы оценки качества ПО необходимо ввести основные понятия о качестве ПО.

Качество программы (quality) – весь объём признаков и характеристик программы, который относится к её способности удовлетворять установленным или предполагаемым потребностям.

Качество программы не должно быть делом случая. Качество должно гарантироваться процессом разработки.

Контроль качества программного продукта – это систематические действия, подтверждающие пригодность к использованию программного продукта в целом. Цель контроля качества – дать количественные меры качества программной системы.

Уровень качества функционирования (level of performance) – степень, в которой удовлетворяются потребности, представленные конкретным набором значений для характеристик качества. Из приведенной формулировки следует, что не все свойства ПО входят в его качество, а только та их совокупность, которая определяется потребностью в этом ПО. Если по каким-то причинам исчезнет потребность в данном ПО, то его качество будет нулевым.

Характеристики качества программного обеспечения (software quality characteristics) – набор свойств (атрибутов) программной продукции, по которым её качество описывается и оценивается. Характеристики качества программного обеспечения могут быть уточнены на множестве уровней комплексных показателей (подхарактеристик).

Свойства ПО можно условно разделить на функциональные и конструктивные.

Функциональные свойства отражают возможности и специфику применения программы и обуславливают степень её соответствия своему целевому назначению. Они характеризуют программу с точки зрения того, как в действительности она выполняется.

Конструктивные свойства программы более или менее не зависят от её функциональных возможностей и назначения. Они характеризуют программу с точки зрения того, как в действительности она сконструирована, устроена. Наиболее важные свойства разрабатываемого ПО (как функциональные, так и конструктивные) формируются и учитываются заранее, при составлении технического задания и при проектировании.

Подходы к организации оценки качества ИС.

В соответствии с ГОСТ 28195–89 «Оценка качества программных средств» методы определения показателей качества ПО различаются:

- по способам получения информации о ПО – измерительный, регистрационный, органолептический, расчетный;
- по источникам получения информации – традиционный, экспертный, социологический.

Измерительный метод основан на получении информации о свойствах и характеристиках ПО с использованием инструментальных средств. Например, с использованием этого метода определяется объём ПО – число строк исходного текста программ и число строк-комментариев, число операторов и операндов, число исполненных операторов, число ветвей в программе, число точек входа (выхода), время выполнения ветви программы, время реакции и другие показатели.

Регистрационный метод основан на получении информации во время испытаний или функционирования ПО, когда регистрируются и подсчитываются определённые события, например: время и число сбоев и отказов, время передачи управления другим модулям, время начала и окончания работы.

Органолептический метод основан на использовании информации, получаемой в результате анализа восприятия органов чувств (зрения, слуха), и применяется для определения таких показателей, как удобство применения, эффективность и т. п.

Расчетный метод основан на использовании теоретических и эмпирических зависимостей (на ранних этапах разработки), статистических данных, накапливаемых при испытаниях, эксплуатации и сопровождении ПО. При помощи расчетного метода определяются длительность и точность вычислений, время реакции, необходимые ресурсы.

Экспертный метод применяется в случаях, когда задача не может быть решена никаким другим из существующих способов или другие способы являются значительно более трудоёмкими. Экспертный метод рекомендуется применять при определении показателей наглядности, полноты и доступности программной документации, легкости освоения, структурности. Определение значений показателей качества ПО экспертным методом осуществляется группой экспертов-специалистов, компетентных в решении данной задачи, на базе их опыта и интуиции. Социологические методы основаны на обработке специальных анкет-вопросников.

В настоящее время используется подход к оценке качества ПО, основанный на комплексном использовании всех методов получения количественных значений показателей качества.

Оценка качества ПО проводится на фазах жизненного цикла и включает выбор номенклатуры показателей, их оценку и сопоставление значений показателей, полученных в результате сравнения, с базовыми значениями.

34. Документационное сопровождение ИС. Документация передаваемая пользователю.

ГОСТ-ы серии 19.xx – серия Единая Система Проектной Документации, лаконичные требования ко многим видам документов.

ГОСТ 34.201-89 Виды, комплектность и обозначения документов при создании автоматизированных систем — Это базовый документ, в котором приводится полный перечень документации ГОСТ 34, рекомендации по кодированию документов, к каким стадиям проекта относятся документы (стадии описываются в ГОСТ 34.601-90), а также как их можно объединить между собой. Фактически, этот стандарт представляет собой большую таблицу с комментариями. Ее можно загнать в Excel для удобства использования.

РД 50-34.698-90 Автоматизированные системы. Требования к содержанию документов — Объемистый стандарт, с различной степенью детальности описывающий содержание проектных документов. В качестве индекса используется упомянутый выше ГОСТ 34.201-89.

ГОСТ Р 51.904-2002 Программное обеспечение встроенных систем. Общие требования к разработке и документированию — Стандарт кроме полного перечня процессов разработки регламентирует количество и содержание документов сопровождающих процессы.

Документация передаваемая пользователю:

ГОСТ Р ИСО 9127-94 «Документация пользователя и информация на упаковке для потребительских программных пакетов» – наиболее приглянувшийся мне стандарт из наших. Довольно кратко (весь документ – около 20 страниц) указаны основные требования к составу и содержанию документации пользователя.

ГОСТ Р ИСО/МЭК 15910-2002 «Процесс создания документации пользователя программного средства» — стандарт больше отвечает не на вопрос «Что» должно быть в документе, а «Как» должен создаваться документ. Дополнительно есть подробное описание стиля документа с примером – довольно удобная штука для создания шаблона: один раз запариваешься (и забиваешь в шаблон всё: от выравнивания до формата подписей рисунков), а потом клепаешь документы все одного вида, а не с заголовками разного шрифта.

IEEE Std 1063-2001 «IEEE Standard for Software User Documentation» — в документе обозначены требования к структуре, содержанию и формату инструкций пользователя.

IEEE Std 1016-1998 «IEEE Recommended Practice for Software Design Descriptions» — рекомендации к документам, описывающим архитектуру программного обеспечения, то бишь к техническому описанию.

ISO/IEC FDIS 18019:2004 «Guidelines for the design and preparation of user documentation for application software» — рекомендации по созданию документации пользователя. Довольно приятное руководство с большим количеством примеров (больше подходит для чтения до или в самом начале создания документации, так как подходит к процессу основательно, от самого планирования). Также в приложениях есть чеклисты «что не забыть сделать в процессе разработки документации» и «что должно быть в документе»

ISO/IEC 26514:2008 «Разработка программного обеспечения и проектирование систем. Требования к проектировщикам и разработчикам документации для пользователя» охватывает этапы разработки, оформления и выпуска документации для пользователей. Стандарт устанавливает требования к структуре, содержанию информации и формату документации, включая печатные документы и документы, выводимые на экран, применяемые в рабочей среде пользователями систем, содержащих программное обеспечение. Стандарт рекомендует, чтобы разработка документации пользователя являлась частью разработки программного продукта и следовала тем же процессам, относящимся к жизненному циклу программного обеспечения. Документация пользователя остается важнейшим компонентом используемого программного продукта. ISO/IEC 26514:2008 может быть полезным для разработчиков следующих видов документации:

- * документация о продукте, отличная от программного обеспечения;
- * мультимедийные системы с использованием анимации, видео и звука;
- * компьютерные учебные программы и специальный учебный материал;
- * документация для операторов компьютеров или системных администраторов, не являющихся конечными пользователями;
- * документация по обслуживанию, описывающая функционирование системного программного обеспечения