# Понятие информационной системы. Обязательные элементы информационной системы

ИС – вся инфраструктура предприятия, задействованная в процессе управления всеми информационно-документальными потоками.

КИС – совокупность технических и программных средств предприятия, которые реализуют идеи и автоматизацию ее деятельности.

Обязательные элементы ИС:

* Информационная модель – совокупность правил и алгоритмов функционирования ИС. Включает в себя все формы документов, структуру справочников данных и т.д.
* Регламент развития ИМ и правила внесения в нее изменения
* Кадровые ресурсы, отвечающие за настройку и адаптацию программного обеспечения, и его соответствие утвержденной информационной модели
* Аппаратно-техническая база, соответствующая требованиям по эксплуатации программного обеспечения (компьютеры на рабочих местах, периферия, каналы телекоммуникаций, системное программного обеспечение и СУБД)
* Правила использования программного обеспечения и пользовательские инструкции, регламент обучения и сертификацию пользователей

# Понятие ситуативного управления (основные объекты и субъекты, механизмы обеспечения контроля исполнения, цели, глубина автоматизации). Классы программного обеспечения, предназначенные для автоматизации данного управления

Ситуативное управление – реакция на форс-мажоры либо же на возникающие во ходе работы ситуации. Ситуативное управление рассматривают как управление, в котором поставлены определенным образом предвидение опасности, анализ ее симптомов и мер по снижению негативных последствий в ее устранении, а так же использование полученного опыта для последующего развития. В его основе лежат следующие положения:

* не существует универсального подхода к управлению
* разные подходы и проблемные ситуации требуют различных подходов к их разрешению
* результаты одних и тех же управленческих решений могут отличаться друг от друга
* существует более одного пути достижения цели
* ситуационные факторы учитываются в стратегиях, структурах и процессах, благодаря чему достигается эффективное принятие решений

Процесс управления состоит из следующих обязательных шагов:

* получение руководителем необходимых знаний
* анализ ситуации
* выбор подходящего подхода и метода управления в сложившейся ситуации
* оценка вероятных последствий
* создание условий для проведения изменений
* проведение изменений

Центральный объект ситуационного управления – сама управленческая ситуация(совокупность всех сведений о структуре объекта и его функционировании в данный момент времени). Центральный субъект управления – руководитель.

Класс ПО для автоматизации ситуативного управления: почта, CRM.

# Понятие проектного управления (основные объекты и субъекты, механизмы обеспечения контроля исполнения, цели, глубина автоматизации). Классы программного обеспечения, предназначенные для автоматизации данного управления

Проект – целенаправленный, заранее проанализированный и спланированный комплекс взаимосвязанных мероприятий по созданию или изменению какого-либо объекта (совокупности объектов) направленный на достижение заранее определенных целей в течение заданного периода времени, при установленном бюджете и определенном качестве.

Жизненный цикл проекта – промежуток времени между моментом появления проекта и его ликвидацией.

Жизненный цикл традиционных проектов:

* Концептуальная фаза(формулирование идей, разработка концептуального плана, обоснование осуществимости проекта)
* Фаза разработки проекта(планирование, определение структуры работ, формирование бюджета и т.д.)
* Фаза выполнения проекта(работы по реализации проекта)
* Фаза завершения проекта(приемочные испытания, пробная эксплуатация, сдача объекта и анализ накопленной информации)

Объект проектного управления – особым способом организованный комплекс работ, направленный на решение определенной задачи или достижение определенной цели, выполнение которого ограниченно во времени и связано с затратами конкретных ресурсов.

Класс ПО для автоматизации проектного управления – PM.

# Понятие директивного управления (основные объекты и субъекты, механизмы обеспечения контроля исполнения, цели, глубина автоматизации). Классы программного обеспечения, предназначенные для автоматизации данного управления

Директивное управление – основано на том, что субъект управления(управляющие орган) вырабатывает директивы, команды, распоряжения, подлежащие неукоснительному исполнению со стороны объекта управления.

В основе данной модели управления лежит принцип безоговорочного подчинения начальнику. Объект управления – сам человек, личность. Одна из особенностей модели – приказы руководителей по умолчанию считаются правильным и не подлежат обсуждению со стороны подчиненных. Вина за невыполнение ложится на плечи исполнителя конкретного поручения.

Директивный стиль наиболее эффективен:

* Когда речь идет о решении однозначных/прямолинейных задач
* В кризисных ситуациях
* Когда отступление от указаний руководства грозит серьезными проблемами
* При общении с неуживчивыми работниками

Директивный стиль менее эффективен:

* При решении неоднозначных задач
* В долгосрочной перспективе
* При наличии целеустремленных и талантливых работников

Класс ПО для автоматизации директивного управления – СЭД.

# Понятие процессного управления (основные объекты и субъекты, механизмы контроля исполнения, цели, глубина автоматизации). Классы программного обеспечения, предназначенные для автоматизации данного управления

При процессном подходе к управлению вся деятельность организации рассматривается как набор процессов. Для того, чтобы управлять организацией, нужно управлять всеми ее процессами.

Процесс – совокупность взаимосвязанных и взаимодействующих видов деятельности, которые преобразуют входы в выходы. Подразделения и сотрудники, задействованные в одном процессе могут самостоятельно координировать работу в рамках процесса и решать возникающие проблемы без участия вышестоящего руководства. Процессный подход позволяет более оперативно решать возникающие вопросы и действовать на результат.

Ключевые элементы процессного подхода:

* Вход процесса (элементы для изменений в ходе действия)
* Выход процесса (ожидаемые результаты)
* Ресурсы (необходимые элементы для процесса)
* Владелец процесса (человек, имеющий в своем распоряжении необходимое количество ресурсов)
* Потребители и поставщики процесса
* Показатели процесса (количественные и качественные параметры)
* Ориентированность исполнителей и руководителей на получение результата, нужного компании. Мотивационные схемы персонала привязаны именно к результатам
* Четкая система единоначалия – один руководитель сосредотачивает в своих руках руководство всей совокупностью операций и действий, направленных на достижение поставленной цели и получение заданного результата
* Разгрузка руководителей. Они вмешиваются в оперативное управление только в случае значительных отклонений
* Руководители занимаются своими прямыми обязанностями – организацией эффективного управления и стратегией развития
* На порядок бОльшая операционная эффективность по сравнению с другими схемами управления
* Не критичность для компании смены работников, поскольку есть механизм передачи знаний новым сотрудникам (регламенты бизнес-процессов)
* В случае формирования кроссфункциональных подразделений требуются отдельные процедуры для обеспечения профессионального роста сотрудников (обучение)

Класс ПО для автоматизации директивного управления – BPM.

# Понятие операционного управления (основные объекты и объекты, механизмы обеспечения контроля исполнения, цели, глубина автоматизации). Классы программного обеспечения, предназначенные для автоматизации данного управления

Операционное управление – управление операциями, длительностью несколько часов и проблемами, возникающими по мере выполнения этих задач. Операционное управление хорошо в случае наличия в компании стабильных процессов.

Работа идет по принципу “конвейера” – формируются задачи, передаются в конвейер, а дальше самостоятельно распределяются между освободившимися участниками команды.

Эффективность такого управления может быть определена как достижение поставленных целей.

Совершенствование операционного управления позволяет:

* повысить эффективность работы производственных и обеспечивающих подсистем компании за счет согласованности бизнес- процессов
* сократить операционные затраты за счет устранения дублирующих участков и зон простоя
* повысить производительность за счет стандартизации и автоматизации основных участков работы предприятия
* повысить качество работ за счет внедрения системы нормативов для каждого участка, контроля результатов и оперативного управления по отклонениям

Класс ПО для автоматизации процессного управления: BPM.

# Понятие **Case Management** (основные объекты и субъекты, механизмы обеспечения контроля исполнения, цели, глубина автоматизации). Классы программного обеспечения, предназначенные для автоматизации данного управления

Новое направление в автоматизации управления процессами и знаниями.

Включает элементы как ситуативного, так и процессного управления.

Развивающаяся технология в области управления, поскольку основная идея сводится к следующему: когда неизвестно заранее описание процесса, а нужно сформировать какой-то наряд на исполнение, можно зафиксировать некий case, т.е. объем информации, который можно в дальнейшем использовать. Case включает в себя набор определенных данных, структуру процесса, формирование. Т.е. дальнейшее использование таких кейсов походит на процессное управление, но могут играть важную роль как в формировании базы знаний, чтобы потом превратить их в ситуации.

Адаптивный кейс-менеджмент **–** концепция динамического управления бизнес-процессами предприятия. Системы ACM предназначены для решения задач коллективного взаимодействия сотрудников, выдачи задач и поручений и контроля сроков их исполнения. Система адаптивного кейс-менеджмента позволяет управлять всеми текущими корпоративными проектами и сотрудниками, в них участвующими, полностью контролируя исполнение проекта на каждом этапе и формируя реальную библиотеку «лучших практик» в процессе реальной работы.

* АСМ – инструмент формирования корпоративных знаний
* АСМ предоставляет возможность накопления корпоративных знаний для их повторного использования (ситуации документируются, складываются в библиотеку «шаблонов» и при повторном использовании кейса уже существует все окружение для него)
* Автоматизация БП «как есть» благодаря адаптивности и гибкости системы
* Простота и социальность АСМ систем позволяет автоматизировать все текущие БП без их детального обследования, трудоемкой настройки и программирования

Система АСМ позволяет:

* Управлять кейсами
* Хранить в виртуальной папке кейса всю информацию о БП
* Управлять пользователями, входящими в проектную команду кейса
* Поддерживать управление правилами
* Поддерживать корпоративные дискуссионные форумы
* Вести библиотеку шаблонов
* Осуществлять согласование, рецензирование и утверждение документов. Либо интегрироваться с СЭД

Класс ПО для автоматизации: CLM, SharePoint

# Модель корпоративной информационной системы, учитывающая различные методики управления (уровни: приложения, платформы, сервисы и инфраструктура)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Ситуативное | Директивное | Проектное | Процессное | Операционное | Case management |
| Class | CRM | СЭД | PM | BMP | ERP | CLM, SharePoint |
| Platform | 1C, Docsvision, SAP, Oracle | | | | | |
| Services | PKI, OLAP, EOM | | | | | |
| Infrastructure | ОС, СУБД, средства разработки | | | | | |

# Понятие корпоративной информационной системы. Основные принципы построения

Корпоративная информационная система (КИС) – это совокупность информационных систем отдельных подразделений предприятия, объединенных общим документооборотом, таких, что каждая из систем выполняет часть задач по управлению принятием решений, а все системы вместе обеспечивают функционирование предприятия в соответствии со стандартами качества ИСО 9000.

КИС – совокупность тех и программных средств предприятия, который реализует идеи и автоматизацию ее деятельности. Задача – эффективное управление внешними ресурсами для получения максимальной прибыли.

КИС – совокупность ИС отдельных подразделений с общим документооборотом, общим информационным пространством, каждая из которых выполняет часть задач по управлению предприятием, управлению принятия решений, в совокупности – управление предприятием.

К основным принципам построения КИС относятся:

* Принцип интеграции, заключающийся в том, что обрабатываемые данные вводятся в систему только один раз и затем многократно используются для решения возможно большего числа задач
* Принцип однократного хранения информации
* Принцип системности, заключающийся в обработке данных в различных разрезах, чтобы получить информацию, необходимую для принятия решений на всех уровнях и во всех функциональных подсистемах и подразделениях корпорации; внимание не только к подсистемам, но и к связям между ними; эволюционный аспект – все стадии эволюции продукта, в фундаменте КИС должна лежать способность к развитию
* Принцип комплексности, подразумевающий автоматизацию процедур преобразования данных на всех стадиях продвижения продуктов корпорации

# Основные этапы проектирования КИС с указанием видов архитектур, которые можно получить на этих этапах

1. Анализ системы – обследование и создание моделей деятельности организации, анализ (моделей) существующих КИС, анализ моделей и формирование требований к КИС, разработка плана создания КИС. Функциональная архитектура
2. Проектирование – концептуальное проектирование, разработка архитектуры КИС, проектирование общей модели данных, формирование требований к приложениям. Архитектура данных, системная архитектура, информационная архитектура.
3. Разработка – разработка, прототипирование и тестирование приложений, разработка интеграционных тестов, разработка пользовательской документации. Программная архитектура.
4. Интеграция и тестирование – интеграция и тестирование приложений в составе системы, оптимизация приложений и баз данных, подготовка эксплуатационной документации, тестирование системы
5. Внедрение – обучение пользователей, развертывание системы на месте эксплуатации, инсталляция баз данных, эксплуатация
6. Сопровождение – регистрация, диагностика и локализация ошибок, внесение изменений и тестирование, управление режимами работы ИС

# Различные виды корпоративных информационных систем (классификация)

1. По характеру использования:

* заказные (разработка под заказ с нуля)
* адаптируемые (коробочные (только коробка); среднего класса; высшего класса)

1. По масштабу применения:

* настольные
* групповые

1. По режиму использования:

* запросно-ответные
* диалоговые
* пакетные
* реального времени

По классу систем:

* I MRP – Планирование потребности в материалах (в конце 60-х годов)
* II MRP 2 – Планирование производственных ресурсов
* III ERP – Все ресурсы организации
* IV ERP 2 – Управление внутренними ресурсами и внешними связями организации
* ERP2 = ERP + CRM + SCM
* CRM – управление отношений с заказчиком
* SCM – управление отношений с поставщиком

Сначала – монолитные (без выделения отдельных компонентов: сервера, рабочих мест). С появлением ПК появилась необходимость двухуровневой архитектуры (+появилась СУБД)

Затем классическая трехуровневая модель

Сейчас технологические возможности перекрывают требования пользователей.

Архитектура ограничивается не технологическими возможностями, а задачами бизнеса, безопасности, производительности. Современная КИС строится вокруг ядра «Производство-деньги». Основной подход к логике функционирования КИС – цикл PDCA.

Входные данные в КИС:

1. Внутренние
2. Внешние

* Данные экономические (связанные с информацией с финансовых рынков)
* Законодательные данные

# Обязательные требования, которым должна отвечать корпоративная информационная система.

* Использование архитектуры клиент – сервер
* Поддержка распределенной обработки данных
* Модульный принцип построения систем
* Поддержка технологий интернет / интранет
* Гибкость – способность системы к дальнейшему развитию и адаптации
* Требования надежности – сохранения функционирования при сбоях, обеспечивается созданием резервных копий, организацией протоколирования, дублирующее оборудования
* Эффективность – система считается эффективной если при выделенных ресурсах, система позволяет выполнить все задачи в установленный срок. Конечная оценка эффективности производится только заказчиком
* Безопасность – защита к доступу к информационным ресурсам. Разграничения прав доступа
* Производительность системы. Выражается во временных характеристиках выполнения определенных функций. Информации по издержкам по содержанию системы

# Системы класса MRP. Предназначение, функции и недостатки

Начало истории КИС в 60х (MRP), но это не КИС в современном понимании, т.к. не охватывают все предприятие плюс малый объем входных данных.

Первая АКИС – MRP – алгоритм управления заказами на готовые товары, производством, запасами сырья и материалами. На предприятиях с дискретным типом производства (есть отделенная единица товара и она производится в рамках серия).

1. Изготовление на заказ
2. Сборка на заказ
3. Серийное производство

MRP базируется на планировании производства для минимизации глубины склада по материалам и компонентам, необходимым для производства.

Первые MRP были для долгосрочного планирования – от месяца до года.

Алгоритм подразумевает распределение заказов на поставку материалов во времени так, чтобы заказ поставщиком был выполнен к моменту потребности в материале.

Материал – сырье, отдельные комплектующие, составляющие конечный продукт.

Статус материала – основной указатель на текущее состояние материала. 4 основных статуса:

1. Материал есть на складе
2. Есть на складе, но зарезервирован
3. Материал заказан
4. Заказ на материал планируется (или необходим)

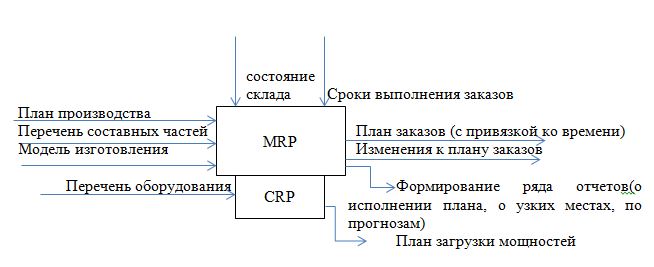
Страховой запас – некое количество материала, необходимое для поддержания процесса производства в случае форс-мажора.

Потребность в материале – количественная мера, отражающая возникающую в некоторый момент времени в текущем периоде планирования потребность в этом материале.

Полная потребность в материале – только то количество материала, которое необходимо пустить в производство.

Чистая потребность – отдельно учитываются страховые и зарезервированный материалы.

Подразумевается автоматизация формирования заказа на товар при возникновении отличной от нуля чистой потребности.

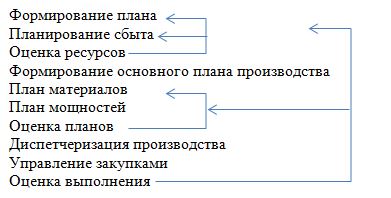


Включение компонента «планирование производственных мощностей» (CRP) – учет загрузки оборудования.

* Нет возможности учитывать в рамках плана изменение текущей ситуации

# Системы класса MRP II. Предназначение, функции, преимущества и недостатки

80е: алгоритм MRP-2 (основа современных КИС) – появилась возможность моделировать ход производства, те управлять производственным процессом.

С развитием компьютерной техники к 80 годам был разработан алгоритм MRP-2, появилась возможность моделировать ход производства, т.е. управлять производственным процессом. Он положил основу современных КИС, раздел управление производством.

С ростом производительности систем и облегчением ввода данных (автоматический ввод данных), увеличением объема обрабатываемых данных появилась возможность перейти к алгоритму MRP-2

Появилась необходимость учета обратной связи (исполнения плана) для оперативного перепланирования.

MRP-2 был направлен на то, чтобы учитывать текущее состояние производства и поток заказов.

Отличия:

1. Текущая обратная связь
2. Жесткая увязка при планировании потребностей в материалах и мощностях в том числе в мощностях учитываются человеческие ресурсы
3. Кроме текущих заказов при планировании учитываются прогнозы спроса, другие маркетинговые исследования
4. При планировании поставок учитываются характеристики поставщиков

Для реализации этого алгоритма были разработаны отдельные модули:

1. Модуль планирования развития бизнеса: маркетинговые оценки финансовых потребностей, анализ рынка, возможности финансирования. Итог – бизнес-план-стратегическое планирование производства.
2. Модуль планирования продаж: на вход – бизнес-план и заказы. Модуль оперирует понятием «готовое изделие». На выходе: конкретные цифры производства готовых изделий с привязкой по времени
3. Модуль планирования производства: обеспечение перевод информации из единицы «готовое изделие» в «комплектующие», последовательность сборки, мощности. Кроме результатов работы 2 модуля на вход подается технология производства.
4. модуля MRP: на входе у– план производства, история заказов, история поставок, возможность собственного производства. на выходе: расписание закупки всех материалов, необходимых для изготовления
5. модуль планирования производственных мощностей – то же, но для оборудования и человеческих ресурсов
6. модуль обратной связи – обеспечение согласование всех планов, контроль исполнения и оперативного внесения изменений

Дальнейшее развитие MRP-2 только за счет развития вычислительных мощностей. За счет автоматизации поступления состоянии мощностей, склада и т.п. появилась возможность более оперативного внесения изменений. Появилась возможность изменения горизонта планирования – период времени, в течение которого система может видеть плановые показатели

Можно увеличивать и улучшать эффективность планирования или уменьшать

* улучшение обслуживания заказчиков за счет своевременного производства готовой продукции
* сокращение цикла производства за счет минимизации простоя из-за отсутствия материалов или мощностей
* сокращение незавершенного производства
* значительное сокращение страховых запасов, за этот счет минимизировать складские помещения и увеличить оборачиваемость средств
* сбалансировать запасы
* уменьшить дефицит и длительность хранения запасов
* повысить производительность за счет минимизации простоев оборудования 100% использования мощностей, в т.ч. людских
* ориентация только на заказ
* слабая интеграция конструирования и проектирования
* слабая интеграция системы технологических процессов
* слабая интеграция планирования кадров и управления финансами

# Системы класса ERP. Предназначение, функции, преимущества и недостатки

ERP – ИС для идентификации и планирования всех ресурсов предприятия, которые необходимы для осуществления продаж, производства, закупок и учета процесса выполнения клиентских заказов.

Идентификация – выявление необходимых ресурсов, определение их количества, качества, автоматизация тестирования этих ресурсов. ERP подразумевает отход от промышленного представления о производстве. Объектом заказа не обязательно являются промышленные товары – это могут быть услуги, финансовые услуги.

ERP-методология – методология эффективного управления ресурсами в сферах производства, дистрибьюции товаром. ERP в чистом виде нет.

ERP системы включают в себя функционал MRP2, плюс добавлено:

* Реализация всех типов производства
* Интегрирование ресурсов по различным направлениям деятельности компании, не только производство
* Многозвенное планирование

Для реализации поставленных задач в дополнение к классическим компонентам MRP систем для выполнения задач, связанных с интегрированным управлением ресурсам, появляются дополнительные компоненты

1. Планирование и управление организацией производственных проектов – анализ проекта (разработка структуры, разделение подпроектов, планирование материальных и трудовых ресурсов)
2. Планирование работы сервисно-технических служб – отвечает за ТО, планирование сервисных работ, планирование ресурсов на ТО, обеспечение аварийного и планового ремонта, замены оборудования. Находится в тесной связи с подсистемой управления производством, т.к. осуществляет блокирование ресурсов типа «оборудование», помогает составлять альт маршруты
3. Планирование у управление распределёнными ресурсами – обеспечение логистических схем работы с бытовых складов, направлена на минимизацию затрат на транспортировку, организацию сбалансированного распределения материалов по складам, оптимизацию логистических схем
4. Планирование и управление постпродажным и специальным обслуживанием – осуществляет сервисное обслуживание (если компания этим занимается)
5. Отдельный аспект – реализация планирования финансовых ресурсов и управление финансовыми потоками

Особенности финансового управления в ERP:

1. Поддержка многозвенной структуры финансового управления
2. Возможность анализа финансового управления в различных разрезах, в том числе в рамках одного проекта
3. ERP обеспечивают гибкость финансового управления: поддержка нескольких систем бухучета, нескольких валют, часовых поясов
4. Особенности ЕРП- обладание полноценным аппаратом бухгалтерского и управленческого учета
5. Ведение финансового планирования.
6. Поддержка дебеторских и кредиторских схем
7. Аппарат управления кредитами,
8. Подсистема поддержки принятия решений. В основе технология OLAP, подразумевающая многомерную аналитику. В основе технологии OLAP лежит доступ ко всем данным ИС и возможность построения отчетов различной конфигурации, поддержка процедур различных анализов, на этом строятся элементы прогнозирования различных показателей. В первую очередь для отслеживания эффективности различных участков, служб, снижение издержек, определение тенденций внутри и на внешнем рынке. OLAP эффективна на больших объемах данных (как по количеству, так и по качеству данных (по номенклатуре позиций))

Отличительная особенность архитектуры ERP: модульность по функциональному признаку. Все модули работают с единым информационным пространством

Чаще всего выделяют подсистемы:

1. MRP-компонент
2. Складская подсистема
3. Логистическая подсистема
4. Финансовая
5. Подсистема документооборота
6. Аналитическая (поддержки принятия решения)

Все модули работают на основе единой СУБД

Особенность – возможность взаимодействия с ERP-системами других предприятий (как на информационном уровне, так и на уровне данных)

50% успеха ERP-системы – это качественное внедрение программного продукта.

Среднестатистическое время внедрения – от полугода до 3-5 лет в зависимости от размеров компании, номенклатуры ее деятельности и сложности процессов внутри компании. (+кто внедряет и насколько это нужно компании)

С точки зрения инфраструктуры системы класса ЕРП взаимодействуют с предприятием посредством операторов, поскольку основная часть информации не поддается автоматическому вводу, а только вручную или взаимодействие с другими системами для получения документов. Еще один способ автоматического ввода информации – фиксация состояния оборудования, иногда продукции (не для всех типов производства). Поэтому системы ЕРП не могут функционировать без операторов.

* единая инфраструктура
* системная и информационная безопасность
* хорошее взаимодействие с CRM-системами
* много ресурсов –> обучение персонала
* очень дорогая
* долгое внедрение
* возможные проблемы с совместимостью

# Системы класса CRM. Предназначение, функции

CRM (Customer Relationship Management) – прикладное программное обеспечение для организаций, предназначенное для автоматизации стратегий взаимодействия с заказчиками (клиентами).

Основной целью внедрения, как правило, ставится увеличение степени удовлетворённости клиентов за счёт анализа накопленной информации о клиентском поведении, регулирования тарифной политики, настройки инструментов маркетинга. Благодаря применению автоматизированной централизованной обработки данных появляется возможность эффективно и с минимальным участием сотрудников учитывать индивидуальные потребности заказчиков, а за счёт оперативности обработки – осуществлять раннее выявление рисков и потенциальных возможностей.

Основные принципы:

* Наличие единого хранилища информации, куда собираются сведения о взаимодействии с клиентами – клиентской базы
* Использование многих каналов взаимодействия: обслуживание на точках продаж, телефонные звонки, электронная почта, мероприятия, встречи, регистрационные формы на веб-сайтах, рекламные ссылки, чаты, социальные сети
* Анализ собранной информации о клиентах и подготовка данных для принятия соответствующих организационных решений – например, сегментация клиентов на основе их значимости для компании, потенциальном отклике на те или иные промоакции, прогнозе потребности в тех или иных продуктах компании

# Системы класса CAD/CAM/CAE. Предназначение, функции

Конструкторские системы.

CAD (Computer Aided Design) – общий термин для обозначения всех аспектов проектирования с использованием средств вычислительной техники. Обычно охватывает создание геометрических моделей изделия. А также генерацию чертежных изделий и их сопровождений.

CAM (Computer Aided Manufacturing) – общий термин для обозначения системы автоматизированной подготовки производства, общий термин для обозначения ПС подготовки информации для станков с ЧПУ. Традиционно исходными данными для таких систем были геометрические модели деталей, полученных из систем CAD.

CAE (Computer Aided Engineering) – система автоматического анализа проекта. Общий термин для обозначения информационного обеспечения условий автоматизированного анализа проекта, имеет целью обнаружение ошибок (прочностные расчеты) или оптимизация производственных возможностей.

# Системы класса MES. Предназначение, функции

Оперативное планирование и управление производством.

MES (manufacturing execution system) — специализированное прикладное программное обеспечение, предназначенное для решения задач синхронизации, координации, анализа и оптимизации выпуска продукции в рамках какого-либо производства. MES-системы относятся к классу систем управления уровня цеха, но могут использоваться и для интегрированного управления производством на предприятии в целом.

Среди основных задач MES выделяются:

1. Активация производственных мощностей на основе детального пооперационного планирования производства
2. Отслеживание производственных мощностей
3. Сбор информации, связанной с производством, от:
4. систем автоматизации производственного процесса
5. датчиков
6. оборудования
7. персонала
8. программных систем
9. Отслеживание и контроль параметров качества
10. Обеспечение персонала и оборудования информацией, необходимой для начала процесса производства
11. Установление связей между персоналом и оборудованием в рамках производства
12. Установление связей между производством и поставщиками, потребителями, инженерным отделом, отделом продаж и менеджментом
13. Реагирование на:

* Требования по номенклатуре производства
* Изменение компонентов, сырья и полуфабрикатов, применяемых в процессе производства
* Изменение спецификации продуктов
* Доступность персонала и производственных мощностей

1. Гарантирование соответствия применимым юридическим актам
2. Соответствие вышеперечисленным индустриальным стандартам

# Системы класса СЭД. Предназначение, функции

ИС в начале истории управляло атомарной единицей данных, в дальнейшем встал вопрос о формировании обобщенной сущности, представляющей набор данных – документа.

Жизненный цикл документа:

1. Создание документа
2. Утверждение документа
3. Использование документа
4. Архивное хранение

Деятельность по управлению ЖЦ документа – управление документооборотом. В общем случае, управление документооборотом заключается в том, что все обновления документов и их частей должны утверждаться и фиксироваться. Для ускорения процессов управления ЖЦ документооборотом появилась необходимость их автоматизации.

СЭД – комплекс программ для контролируемого управления и создания документов на предприятии в соответствии с правилами работы с документами

ЭД – контейнер, объединяющий структурированную информацию. Структура представления информации – шаблон документа. В шаблон документа входит кол-во, содержание элементов, правила оформления

Документ обладает рядом специализированных свойств:

1. Обязательные характеристики: дата создания, автор, тип шаблона, тип ЖЦ-формализованная процедура утверждения (согласование – последовательность сотрудников или ролей, которые обязаны выполнить определенные операции над документом, связанные с согласованием или утверждением) целевая группа (в ЖЦ), … (гриф секретности и т.д.)
2. Переменное свойство – история документа, где фиксируются все события, связанные с любыми операциями над документом

СЭД содержит:

1. Управление пользователями – т.к. для каждого документа осуществлять проверку и контроль за правами пользователей (отдельных или групп)
2. Система хранения полнотекстовых версий документа (копии физических документов)
3. Подсистема контроля исполнения поручений (документов)
4. Интеграция с системами офисных приложений (Word, Excel) и системами передачи данных (почтовые клиенты, веб-порталы)
5. Система ЭЦП на документах и ее контроля

Основные принципы электронного документооборота

* Однократная регистрация документа, позволяющая однозначно идентифицировать документ
* Возможность параллельного выполнения операций, позволяющая сократить время движения документов и повышения оперативности их исполнения
* Непрерывность движения документа, позволяющая идентифицировать ответственного за исполнение документа (задачи) в каждый момент времени жизни документа (процесса)
* Единая (или согласованная распределённая) база документной информации, позволяющая исключить возможность дублирования документов
* Эффективно организованная система поиска документа, позволяющая находить документ, обладая минимальной информацией о нём
* Развитая система отчётности по различным статусам и атрибутам документов, позволяющая контролировать движение документов по процессам документооборота и принимать управленческие решения, основываясь на данных из отчётов

# Виды архитектур ИС и основные объекты, характеризующие описание ИС в рамках каждого вида архитектуры

Часто архитектурные решения имеют следующие критерии:

1. Масштабируемость и универсальность
2. Обратная совместимость с предыдущими версиями
3. Внешняя совместимость с аппаратно-программным обеспечением
4. Соблюдение отраслевых или иных стандартов

Три уровня архитектуры системы:

* Системная архитектура задает отношения всех высокоуровневых компонентов (аппаратной части, программной, сетевой и т. д. Описываются общие уровни)
* Программная архитектура – обсуждается как каждый компонент реализован на программном уровне.
* Архитектура данных. Как будут хранится данные? (Какую модель реляционную или нет? Производить индексирование или нет? И т.д.)

# Файл-серверная архитектура. Описание. Преимущества и недостатки

Файл-серверные приложения – приложения, схожие по своей структуре с локальными приложениями и использующие сетевой ресурс для хранения данных в виде отдельных файлов. Функции сервера в таком случае обычно ограничиваются хранением данных (возможно также хранение исполняемых файлов), а обработка данных происходит исключительно на стороне клиента. Количество клиентов ограничено десятками ввиду невозможности одновременного доступа на запись к одному файлу. Однако клиентов может быть в разы больше, если они обращаются к файлам исключительно в режиме чтения.

* низкая стоимость разработки
* высокая скорость разработки
* невысокая стоимость обновления и изменения ПО
* многопользовательский режим работы с данными
* централизованное управление доступом
* рост числа клиентов резко увеличивает объём трафика и нагрузку на сети передачи данных
* высокие затраты на модернизацию и сопровождение сервисов бизнес-логики на каждой клиентской рабочей станции
* низкая надёжность системы

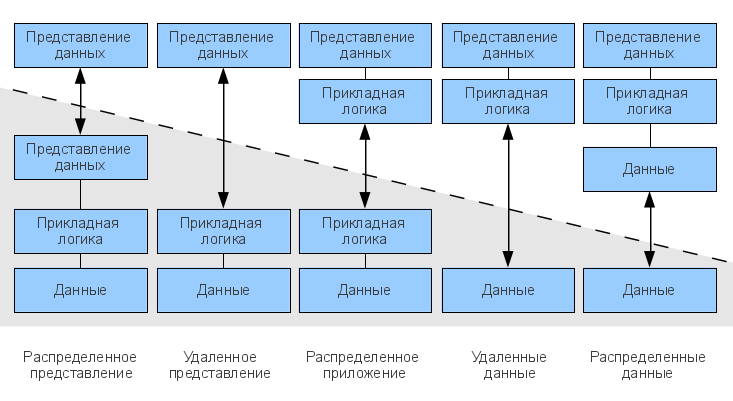
# http://www.4stud.info/networking/img/2-tier.pngКлассическая (двухзвенная) клиент-серверная архитектура. Распределение функций ИС по слоям (звеньям)

Двухзвенной она называется из-за необходимости распределения трех базовых компонентов между двумя узлами (клиентом и сервером).

Двухзвенная архитектура используется в клиент-серверных системах, где сервер отвечает на клиентские запросы напрямую и в полном объеме, при этом используя только собственные ресурсы. Т.е. сервер не вызывает сторонние сетевые приложения и не обращается к сторонним ресурсам для выполнения какой-либо части запроса

Расположение компонентов на стороне клиента или сервера определяет следующие основные модели их взаимодействия в рамках двухзвенной архитектуры:

* сервер терминалов — распределенное представление данных
* файл-сервер — доступ к удаленной базе данных и файловым ресурсам
* сервер БД — удаленное представление данных
* сервер приложений — удаленное приложение



* Многопользовательский режим работы
* Гарантия целостности данных
* Простота организации
* Высокие требования к пропускной способности, клиентским машинам
* Высокая сложность администрирования и разработки
* Слабая защита данных от взлома

# http://www.4stud.info/networking/img/3-tier.pngТрехзвенная (трехслойная) клиент-серверная архитектура. Описание. Преимущества и недостатки

Архитектурная модель программного комплекса, предполагающая наличие в нём трёх компонентов: клиента, сервера приложений (к которому подключено клиентское приложение) и сервера баз данных (с которым работает сервер приложений).

* представление данных – на стороне клиента
* прикладной компонент – на выделенном сервере приложений (как вариант, выполняющем функции промежуточного ПО)
* управление ресурсами – на сервере БД, который и представляет запрашиваемые данные
* тонкий клиент
* минимизация потока данных между клиентом и сервером
* масштабируемость, производительность
* снижение нагрузки на сервер данных
* простота реализации
* высокие расходы на администрирование и разработку серверной части

# Толстый и тонкий клиент. Преимущества и недостатки реализации трехслойной архитектуры с тонким и толстым клиентом

Тонкий клиент – переносит все или большую часть задач по обработке информации на сервер:

* сложность с администрированием
* при обновлении надо переустановить всех клиентов
* требования к клиентскому аппаратному обеспечению

Толстый клиент – приложение, обеспечивающее расширенную функциональность независимо от центрального сервера:

* более высокие требования к пропускному каналу и к производительности оборудования
* более сложные в разработке

# Виды распределенных архитектур ИС. Области их применения

Распределённые ИС, в свою очередь, разделяют на:

* файл-серверные ИС (ИС с архитектурой «файл-сервер»)
* клиент-серверные ИС (ИС с архитектурой «клиент-сервер»)

В файл-серверных ИС база данных находится на файловом сервере, а СУБД и клиентские приложения находятся на рабочих станциях.

В клиент-серверных ИС база данных и СУБД находятся на сервере, а на рабочих станциях находятся только клиентские приложения.

В свою очередь, клиент-серверные ИС разделяют на двухзвенные и многозвенные.

В двухзвенных ИС всего два типа «звеньев»: сервер базы данных, на котором находятся БД и СУБД (back-end), и рабочие станции, на которых находятся клиентские приложения (front-end). Клиентские приложения обращаются к СУБД напрямую.

В многозвенных ИС добавляются промежуточные «звенья»: серверы приложений (application servers). Пользовательские клиентские приложения не обращаются к СУБД напрямую, они взаимодействуют с промежуточными звеньями. Типичный пример применения трёхзвенной архитектуры — современные веб-приложения, использующие базы данных. В таких приложениях помимо звена СУБД и клиентского звена, выполняющегося в веб-браузере, имеется как минимум одно промежуточное звено — веб-сервер с соответствующим серверным программным обеспечением.

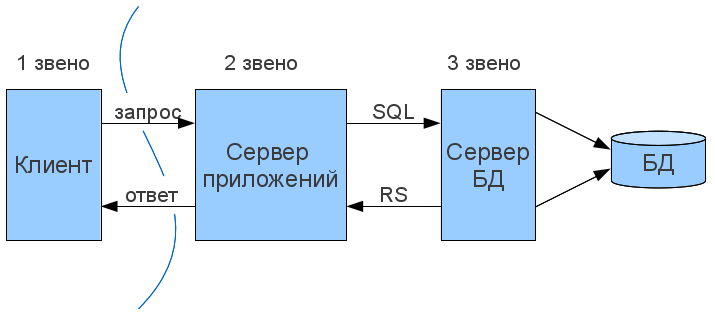
# Цели построения распределенных ИС

1. Повысить производительность за счёт параллельной обработки
2. Снизить телекоммуникационные расходы за счёт приближения аппаратных узлов к пользователю (проблема последней мили)
3. Повысить надёжность за счёт дублирования обрабатывающих узлов и хранилищ данных
4. Повысить масштабируемость за счёт абстрагирования функций и узлов

# Проблемы, возникающие при построении распределенных ИС и подходы к их устранению

1. Обеспечение целостности данных. Решение: транзакция/синхронизация или на уровне архитектуры – модели данных с невозможностью удаления/изменения, только добавления
2. Оптимизирование использования ресурсов при неравномерной нагрузке. Решение: миграция, перемещение программных узлов по аппаратным ресурсам, балансировка запросов
3. Синхронизация событий и взаимоблокировка (deadlock, кто первый)
4. Стандартизация интерфейса
5. Сложности с мониторингом и диагностикой

# Методы организации взаимодействия программных компонентов при построении трехзвенной клиент-серверной архитектуры

RPC (Remote Procedure Call): App имеет заглушку; обеспечивает связь с другими приложениями (не ООП)

ООП – удалить объекты, удалить вызов метода:

1. сериализация (передать объекты с одного узла на другой)
2. RMI (Remote Method Invocation): удалить вызов метода у объекта, ссылка у объекта

CORBA (Common Object Request Broker Architecture): чтоб не обновлять часто таблицы ссылок на другие приложения.

IDL (Interface Definition Language): позволяет унифицировано описать интерфейсы.

# Балансировка нагрузки при построении распределенных ИС. Виды балансировки, их преимущества и недостатки

* Синхронный. Один балансировщик нагрузки + мониторинг
* узкое место – балансировщик
* Полусинхронный. Запросы принимает один узел, результат приложение отправляет соответствующему клиенту (не через балансировщик)
* распределенность на выходе
* не оценить реальное состояние узла
* Асинхронный. Нет балансировщика, есть пул запросов и пул ресурсов, которые с помощью алгоритмов захватывают свободный ресурс.
* узкое место – хранилище

# Системы с распределенным хранением данных. Виды систем и особенности их реализации и использования

Все, что знаете о Cassandra – сюда.

Если прошла тяжелая транзакция, то много мелких будут долго ждать, разобьем хранилище на куски, которые достаточно автономны. И если к какому-то узлу большая нагрузка, то перемещаем этот кусок на другой узел, где нагрузка меньше.

Или можно хэшировать на другие узлы для чтения, и при обновлении старые хэши удаляются и по требованию создается новый хэш.

1. Единственность перемещаемого элемента данных (не копии, а перемещение)
2. Дублирование на чтение (вариант кэширования)
3. Дублирование на запись (можно записывать в копии, а потом в фоновом режиме синхронизироваться между кэшами)

# Сервисно-ориентированная архитектура ИС. Особенности построения, преимущества и недостатки

Сервис ориентированная архитектура SOA – модульный подход к разработке программного обеспечения, основанный на использовании распределённых, слабо связанных заменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам.

Сервис – черный ящик, идентифицируемый своим API. Главное, что отличает SOA – это использование независимых сервисов с чётко определёнными интерфейсами, которые для выполнения своих задач могут быть вызваны неким стандартным способом, при условии, что сервисы заранее ничего не знают о приложении, которое их вызовет, а приложение не знает, каким образом сервисы выполняют свою задачу.

Интерфейсы компонентов в сервис-ориентированной архитектуре инкапсулируют детали реализации (ОС, платформу, ЯП) от остальных компонентов, обеспечивая комбинирование и многократное использование компонентов для построения сложных распределённых программных комплексов, обеспечивая независимость от используемых платформ и инструментов разработки, способствуя масштабируемости и управляемости создаваемых систем.

Программные комплексы, разработанные в соответствии с сервис-ориентированной архитектурой, обычно реализуются как набор веб-служб, взаимодействующих по протоколу SOAP (WSDL) или REST (CRUD).

Сервисы могут обмениваться информацией посредством RabbitMQ при горизонтальной кластеризации сервисов, что позволит организовывать систему очередей по степени нагрузки и производительности.

* Достаточно большое время отклика
* Некоторой избыточностью пересылаемой информации, что при больших нагрузках может привести к перегруженности сетевого трафика
* Сложность развертывания
* Затратность развертывания

# Облачная архитектура построения ИС. Применение технологий виртуализации при реализации облачных систем

Сервис ориентированная архитектура SOA – модульный подход к разработке программного обеспечения, основанный на использовании распределённых, слабо связанных заменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам.

Сервисам разрешено мигрировать по аппаратным ресурсам.

* SAAS (Software as a Service) – Абстрагировано все, кроме интерфейса приложения
* PASS (Platform as a Service) – собираем свою систему из компонентов платформы
* IAAS (Infrastructure as a Service) – абстрагировано железо (виртуальные сервера)