Кафедра «Техника и технологии»

РЕФЕРАТ

по дисциплине:

«Архитектура информационных систем»

Тема: Теоретические вопросы к экзамену

Направление подготовки/специальность 09.03.02 Информационные системы и технологии

(код, наименование)

Обучающийся Рыжова Мария Владиславовна

(ФИО полностью)

Группа И-107а Подпись\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(номер группы)

Форма обучения Очная

Проверил Ефимов М. А.

(Фамилия И.О. преподавателя)

Должность \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Челябинск, 2025 г.**Выделенные жёлтым вопросы**

1. **Основные понятия архитектуры информационных сетей**   
   Архитектура информационных сетей — это совокупность принципов, стандартов и технологий, определяющих структуру, взаимодействие компонентов и функционирование сетей передачи данных.   
   Основные понятия:  
   1. Топология сети – способ соединения узлов сети (шина, звезда, кольцо, ячеистая и др.).   
   2. Протоколы передачи данных – набор правил, определяющих формат и порядок обмена данными (TCP/IP, HTTP, FTP, Ethernet).   
   3. Уровни модели OSI (Open Systems Interconnection) – 7-уровневая модель, описывающая процесс передачи данных:   
   - Физический (кабели, сигналы)   
   - Канальный (MAC-адреса, Ethernet)   
   - Сетевой (IP-адреса, маршрутизация)   
   - Транспортный (TCP/UDP)   
   - Сеансовый (управление соединением)   
   - Представительский (кодирование, шифрование)   
   - Прикладной (HTTP, FTP, SMTP)   
   4. Сетевое оборудование – устройства, обеспечивающие передачу данных (маршрутизаторы, коммутаторы, модемы, точки доступа).   
   5. IP-адресация и DNS – система идентификации узлов в сети (IPv4/IPv6) и преобразования доменных имен в IP-адреса.   
   6. LAN (Local Area Network) – локальная сеть (офис, дом).   
   7. WAN (Wide Area Network) – глобальная сеть (Интернет).   
   8. VPN (Virtual Private Network) – защищенное соединение через публичные сети.   
   9. QoS (Quality of Service) – управление приоритетами трафика для обеспечения качества связи.   
   10. SDN (Software-Defined Networking) – программно-конфигурируемые сети.   
   **2. Безопасность информации в системе**   
   Безопасность информации – защита данных от несанкционированного доступа, повреждения или утечки.   
   Основные аспекты безопасности:   
   1. Конфиденциальность – доступ к информации только для авторизованных пользователей.   
   2. Целостность – защита данных от несанкционированного изменения.   
   3. Доступность – обеспечение работоспособности системы для легальных пользователей.   
   Методы и технологии защиты:   
   1. Аутентификация и авторизация (пароли, биометрия, двухфакторная аутентификация).   
   2. Шифрование данных (SSL/TLS, AES, RSA).   
   3. Межсетевые экраны (Firewall) – фильтрация трафика.   
   4. Системы обнаружения вторжений (IDS/IPS) – мониторинг атак.   
   5. Антивирусные программы – защита от вредоносного ПО.   
   6. Резервное копирование – восстановление данных после сбоев.   
   7. Политика безопасности – регламенты и правила доступа.   
   8. Защита от DDoS-атак – распределенные атаки на доступность.   
   9. VPN и защищенные протоколы (IPSec, SSH).   
   10. Аудит и мониторинг – анализ событий безопасности.   
   Стандарты и нормативы:   
   - ISO 27001 – международный стандарт информационной безопасности.   
   - GDPR – защита персональных данных в ЕС.   
   - ФЗ-152 (Россия) – закон о персональных данных.   
   **3. Предметные области ИС**   
   Предметная область информационной системы (ИС) – это сфера деятельности, для которой проектируется и используется ИС. Она определяет данные, процессы, правила и требования к системе.   
   Основные предметные области ИС:   
   1. Корпоративные системы (ERP, CRM, SCM)   
   - Управление ресурсами предприятия (финансы, логистика, персонал).   
   - Примеры: SAP, 1С, Oracle ERP.   
   2. Государственные и муниципальные ИС   
   - Электронное правительство (Госуслуги, налоговые системы).   
   - Системы учета населения, соцобеспечения.   
   3. Банковские и финансовые системы   
   - Платежные системы (SWIFT, Visa, Mastercard).   
   - Интернет-банкинг, биржевые торговые системы.   
   4. Медицинские ИС (МИС, EHR)   
   - Электронные медкарты (EMR).   
   - Телемедицина, системы диагностики.   
   5. Образовательные системы (LMS, СДО)   
   - Moodle, Blackboard, электронные журналы.   
   - Системы дистанционного обучения.   
   6. Транспортные и логистические ИС   
   - Управление грузоперевозками (TMS).   
   - Навигационные системы (GPS, ГЛОНАСС).   
   7. Промышленные ИС (АСУ ТП, SCADA)   
   - Автоматизация производства.   
   - Управление технологическими процессами.   
   8. Телекоммуникационные системы   
   - Управление сетями (OSS/BSS).   
   - Системы биллинга (например, у операторов связи).   
   9. Научные и исследовательские ИС   
   - Системы обработки больших данных (Big Data).   
   - Моделирование и симуляция (например, в физике, биологии).   
   10. Интернет вещей (IoT) и умные города   
   - Управление умными домами, транспортными системами.   
   - Датчики, автоматизация городской инфраструктуры.   
   **4. Архитектуры информационных систем**   
   Архитектура ИС – это структура системы, включающая компоненты, их взаимодействие и принципы построения.   
   Основные типы архитектур:   
   1. Монолитная архитектура   
   - Единое приложение, все компоненты связаны в одном процессе.   
   - Плюсы: Простота разработки и развертывания.   
   - Минусы: Сложность масштабирования, низкая отказоустойчивость.   
   - Пример: Традиционные desktop-приложения.   
   2. Клиент-серверная архитектура   
   - Клиент (интерфейс) и сервер (обработка данных).   
   - Двухзвенная (2-tier): Клиент + сервер БД.   
   - Трехзвенная (3-tier): Клиент + сервер приложений + сервер БД.   
   - Пример: Веб-приложения (браузер – клиент, backend – сервер).   
   3. Сервис-ориентированная архитектура (SOA)   
   - Система состоит из независимых сервисов, обменивающихся данными (часто через API).   
   - Плюсы: Гибкость, повторное использование компонентов.   
   - Минусы: Сложность управления.   
   - Пример: Корпоративные системы с веб-сервисами (SOAP, REST).   
   4. Микросервисная архитектура   
   - Развитие SOA: приложение разбито на мелкие независимые сервисы.   
   - Плюсы: Масштабируемость, отказоустойчивость.   
   - Минусы: Сложность развертывания и мониторинга.   
   - Пример: Netflix, Uber (каждый модуль – отдельный микросервис).   
   5. Облачная архитектура (Cloud Computing)   
   - Использование облачных сервисов (IaaS, PaaS, SaaS).   
   - Пример: AWS, Azure, Google Cloud.   
   6. Событийно-ориентированная архитектура (EDA)   
   - Компоненты реагируют на события (асинхронная обработка).   
   - Пример: Системы реального времени (биржевые роботы).   
   7. Peer-to-Peer (P2P) архитектура   
   - Равноправные узлы без центрального сервера.   
   - Пример: BitTorrent, блокчейн-сети.   
   8. Гибридная архитектура   
   - Сочетание нескольких подходов (например, микросервисы + облако).   
   Тренды в архитектуре ИС:   
   - Serverless-архитектура (FaaS – функции как сервис, например, AWS Lambda).   
   - Контейнеризация (Docker, Kubernetes).   
   - Edge Computing (обработка данных ближе к источнику).  
   **5. Прикладной уровень модели OSI**  
   Прикладной уровень является верхним (7-м) уровнем модели OSI и обеспечивает взаимодействие пользовательских приложений с сетевой средой. Основная функция этого уровня - предоставление сетевых сервисов непосредственно прикладным процессам.  
   Ключевые характеристики прикладного уровня:  
   1. Обеспечивает интерфейс между сетевыми службами и прикладными программами  
   2. Определяет протоколы для конкретных сервисов  
   3. Реализует семантику обмена данными  
   Основные протоколы прикладного уровня включают:  
   - HTTP/HTTPS для веб-трафика  
   - FTP для передачи файлов  
   - SMTP, POP3, IMAP для электронной почты  
   - DNS для преобразования имен  
   - DHCP для автоматической настройки сети  
   - SSH для безопасного удаленного доступа  
   Прикладной уровень не занимается собственно передачей данных, а лишь предоставляет интерфейс для работы приложений с сетью, делегируя задачи транспортировки нижележащим уровням.  
   **6. Представительский уровень модели OSI**  
   Представительский уровень (6-й уровень модели OSI) отвечает за представление и преобразование данных, обеспечивая их корректную интерпретацию различными системами.  
   Основные функции представительского уровня:  
   1. Преобразование данных (кодирование/декодирование)  
   2. Шифрование и дешифрование информации  
   3. Сжатие и распаковка данных  
   Важные аспекты работы уровня:  
   - Обеспечивает единое представление данных для разных платформ  
   - Выполняет синтаксический перевод между различными форматами  
   - Реализует механизмы защиты информации  
   Технологии представительского уровня включают:  
   - Системы кодирования (ASCII, Unicode)  
   - Форматы представления данных (JSON, XML)  
   - Алгоритмы шифрования (AES, RSA)  
   - Методы сжатия (GZIP, JPEG)  
   Представительский уровень работает в тесной связке с прикладным, преобразуя данные в форму, пригодную для обработки приложениями, и обратно - в форму, подходящую для передачи по сети.  
   **7. Сеансовый уровень OSI**  
   Сеансовый уровень (5-й уровень модели OSI) обеспечивает управление диалогом между взаимодействующими системами. Основные функции:  
   Установление, поддержание и завершение сеансов связи  
   - Организация логического соединения между приложениями  
   - Синхронизация обмена данными  
   Контроль диалога:  
   - Определение очередности передачи (полудуплексный/дуплексный режим)  
   - Управление токенами (правами на передачу)  
   Синхронизация:  
   - Расстановка контрольных точек для восстановления прерванных сеансов  
   - Маркировка данных для последующей синхронизации  
   Примеры протоколов: RPC (Remote Procedure Call), PPTP (для VPN), NetBIOS.  
   **8. Транспортный уровень OSI**  
   Транспортный уровень (4-й уровень) обеспечивает сквозную передачу данных между конечными системами:  
   Основные функции:  
   - Гарантированная доставка данных  
   - Управление потоком и перегрузками  
   - Сегментация и сборка данных  
   - Мультиплексирование соединений  
   Режимы работы:  
   - С установлением соединения (TCP)  
   - Без установления соединения (UDP)  
   Ключевые протоколы:  
   - TCP (надежная передача с подтверждением)  
   - UDP (быстрая передача без гарантий)  
   - SCTP, SPX  
   **9. Сетевой уровень OSI**  
   Сетевой уровень (3-й уровень) решает задачи маршрутизации в составных сетях:  
   Основные функции:  
   - Логическая адресация (IP-адреса)  
   - Маршрутизация пакетов  
   - Фрагментация и сборка пакетов  
   Ключевые протоколы:  
   - IP (IPv4/IPv6)  
   - Маршрутизационные протоколы (OSPF, BGP, RIP)  
   - Вспомогательные протоколы (ICMP, ARP)  
   Особенности работы:  
   - Оперирует пакетами (датаграммами)  
   - Обеспечивает межсетевое взаимодействие  
   - Реализует функции межсетевого экранирования  
   **10. Канальный уровень OSI**  
   Канальный уровень (2-ой уровень) отвечает за надежную передачу данных между соседними узлами сети. Основные характеристики:  
   Функции и задачи:  
   - Обрамление данных в кадры (фреймы)  
   - Обнаружение и коррекция ошибок передачи  
   - Управление доступом к общей среде передачи  
   - Физическая адресация устройств (MAC-адреса)  
   Подуровни:  
   - LLC (Logical Link Control) - управление логической связью  
   - MAC (Media Access Control) - управление доступом к среде  
   Типовые протоколы:  
   - Ethernet (IEEE 802.3)  
   - Wi-Fi (IEEE 802.11)  
   - PPP (Point-to-Point)  
   - HDLC  
   - Frame Relay  
   Особенности работы:  
   - Использует MAC-адреса для идентификации устройств  
   - Обеспечивает локальную доставку кадров в пределах одного сегмента сети  
   - Реализует механизмы контроля ошибок (CRC)  
   Оборудование: Коммутаторы (L2), мосты, сетевые адаптеры.  
   **11. Физический уровень OSI**  
   Физический уровень (1-ый уровень) обеспечивает физическую передачу битов данных. Основные характеристики:  
   Функции и задачи:  
   - Передача неструктурированного битового потока  
   - Определение электрических и физических характеристик  
   - Кодирование и модуляция сигналов  
   - Синхронизация передачи  
   Основные технологии:  
   - Медные кабели (витая пара, коаксиал)  
   - Оптоволоконные линии  
   - Беспроводные технологии (радио, ИК)  
   - Разъемы и интерфейсы (RJ-45, SFP)  
   Характеристики передачи:  
   - Методы кодирования (NRZ, Manchester)  
   - Типы модуляции (ASK, FSK, PSK)  
   - Физические параметры (напряжение, частота)  
   Особенности работы:  
   - Не анализирует содержимое передаваемых данных  
   - Обеспечивает только физическое соединение  
   - Не выполняет обработку ошибок  
   Оборудование: Концентраторы, повторители, сетевые адаптеры, медиаконвертеры.  
   **12. Протоколы TCP/IP**  
   Стек протоколов TCP/IP — это набор сетевых протоколов, используемых в современных компьютерных сетях и интернете. Основные характеристики:  
   Архитектура:  
   - 4-уровневая модель (в отличие от 7-уровневой OSI)  
   - Уровни: Прикладной, Транспортный, Сетевой, Канальный  
   Ключевые протоколы:  
   - IP (Internet Protocol):  
   - Сетевой уровень  
   - Обеспечивает логическую адресацию (IPv4/IPv6)  
   - Отвечает за маршрутизацию пакетов  
   - TCP (Transmission Control Protocol):  
   - Транспортный уровень  
   - Гарантированная доставка с установлением соединения  
   - Контроль перегрузок, управление потоком  
   - UDP (User Datagram Protocol):  
   - Транспортный уровень  
   - Передача без установления соединения  
   - Минимальные накладные расходы  
   Вспомогательные протоколы:  
   - ICMP — диагностика и сообщение об ошибках  
   - ARP — преобразование IP в MAC-адреса  
   - DNS — преобразование доменных имен  
   - DHCP — автоматическая настройка сети  
   Особенности:  
   - Открытый стандарт  
   - Масштабируемость  
   - Независимость от оборудования  
   - Поддержка маршрутизации  
   **13. Протоколы IPX/SPX**  
   Стек протоколов IPX/SPX — разработан Novell для сетей NetWare. Основные характеристики:  
   Архитектура:  
   - Аналог TCP/IP в сетях Novell NetWare  
   - Использовался преимущественно в локальных сетях  
   Ключевые протоколы:  
   - IPX (Internetwork Packet Exchange):  
   - Сетевой уровень  
   - Обеспечивает адресацию и маршрутизацию  
   - 80-битные адреса (сеть + узел)  
   - SPX (Sequenced Packet Exchange):  
   - Транспортный уровень  
   - Гарантированная доставка с установлением соединения  
   - Аналог TCP в стеке TCP/IP  
   - NCP (NetWare Core Protocol):  
   - Прикладной уровень  
   - Обслуживание запросов файлов/печати  
   Особенности:  
   - Простая конфигурация  
   - Эффективная работа в локальных сетях  
   - Встроенная поддержка в старых ОС (Windows 95/98/NT)  
   - Не требует централизованного сервера имен  
   **14. Понятие спецификаций ИС**  
   Спецификации информационной системы — это формализованное описание требований, параметров и характеристик разрабатываемой или эксплуатируемой ИС.   
   Основные аспекты:  
   Назначение:  
   - Четкое определение функциональных возможностей системы  
   - Установление технических требований  
   - Формализация взаимодействия компонентов  
   Виды спецификаций:  
   - Функциональные (описание возможностей системы)  
   - Технические (аппаратные требования, ПО)  
   - Архитектурные (структура компонентов)  
   - Интерфейсные (правила взаимодействия)  
   Содержание:  
   - Описание бизнес-процессов  
   - Требования к производительности  
   - Протоколы взаимодействия  
   - Ограничения и условия работы  
   Значение:  
   - Основа для проектирования системы  
   - Критерий приемки при разработке  
   - Руководство для модернизации  
   **15. Понятие базы данных**  
   База данных (БД) — организованная совокупность структурированных данных, предназначенных для хранения, обработки и управления.  
   Ключевые характеристики:  
   Основные признаки:  
   - Логическая структурированность данных  
   - Минимальная избыточность информации  
   - Независимость от прикладных программ  
   - Централизованное управление  
   Компоненты:  
   - Данные (контент)  
   - Метаданные (описание структуры)  
   - СУБД (система управления)  
   Типы баз данных:  
   - Реляционные (табличные)  
   - Иерархические  
   - Сетевые  
   - Объектно-ориентированные  
   - NoSQL (документоориентированные и др.)  
   Функции:  
   - Хранение информации  
   - Обеспечение целостности данных  
   - Управление доступом  
   - Обработка запросов

**Остальные вопросы**

**1. Понятие информации**

**Информация** – это осмысленные и обработанные данные, которые несут полезную смысловую нагрузку для человека или системы. В отличие от "сырых" данных, информация обладает такими характеристиками, как актуальность, достоверность, полнота и понятность.

В информационных системах информация возникает в результате обработки, фильтрации и структурирования данных. Она используется для принятия решений, управления процессами и автоматизации бизнес-задач. Например, показатели из базы данных после анализа превращаются в отчёт, который помогает руководству оценить эффективность работы компании.

**2. Отличие информации от данных**

**Данные** – это необработанные факты, числа, символы или сигналы, которые сами по себе не несут смысла. Например, последовательность чисел "01062023" – это просто данные.

**Информация** – это данные, преобразованные в понятную и полезную форму. Если интерпретировать "01062023" как "1 июня 2023 года", это уже информация, так как она имеет конкретное значение.

**Ключевые различия:**

* Данные – это "сырой" материал, а информация – результат их обработки.
* Данные не имеют ценности без контекста, информация – осмысленна и полезна.
* Данные могут быть неструктурированными, информация всегда структурирована и пригодна для использования.

**Пример в ИС:**

* Данные – записи в таблице БД (ID, дата, сумма).
* Информация – отчёт о продажах за месяц, построенный на основе этих данных.

Таким образом, информация – это данные, преобразованные в знание, а информационные системы служат инструментом этого преобразования.

### **Ответы на билет по предмету "Архитектура информационных систем"**

#### ****3. Статическое и динамическое состояние информации****

**Статическое состояние информации** – это неизменяемые (фиксированные) данные, которые сохраняются в постоянной памяти и не модифицируются в процессе работы системы.

**Примеры:** Архивные документы, справочники, исторические записи в БД.

**Применение:** Хранение эталонных данных, резервное копирование, аудит.

**Динамическое состояние информации** – это изменяемые данные, которые обновляются, обрабатываются и передаются в реальном времени.

**Примеры:** Показания датчиков IoT, транзакции в банковских системах, лог-файлы.

**Применение:** Оперативные системы (биржевые котировки, чаты), потоковая аналитика.

**Ключевое отличие:**

* Статическая информация устойчива и предназначена для долгосрочного хранения.
* Динамическая информация изменчива и требует постоянной обработки.

#### ****4. Основные характеристики и виды информации****

**Характеристики информации:**

1. **Актуальность** – соответствие текущему моменту.
2. **Достоверность** – отсутствие ошибок и искажений.
3. **Полнота** – достаточность для принятия решений.
4. **Понятность** – возможность интерпретации пользователем.
5. **Ценность** – полезность для конкретной задачи.

**Виды информации:**

1. **По форме представления:**
   * Текстовая (документы, сообщения).
   * Числовая (статистика, финансовые данные).
   * Графическая (изображения, схемы).
   * Мультимедийная (аудио, видео).
2. **По способу обработки:**
   * Структурированная (таблицы, SQL-базы).
   * Неструктурированная (электронные письма, соцсети).
3. **По назначению:**
   * Управляющая (команды в АСУ).
   * Осведомительная (отчёты, уведомления).

#### ****5. Архитектура открытых систем****

**Архитектура открытых систем (АОС)** – это стандартизированный подход к проектированию ИС, обеспечивающий:

* **Совместимость** компонентов от разных производителей.
* **Масштабируемость** за счёт модульности.
* **Переносимость** данных и приложений между платформами.

**Основные принципы:**

1. **Стандартизация интерфейсов** (например, API REST, протоколы TCP/IP).
2. **Децентрализация** – независимость компонентов (микросервисы).
3. **Открытость** – публичность спецификаций (как в Linux или HTTP).

**Примеры:**

* Сетевые модели OSI и TCP/IP.
* Облачные платформы (AWS, Azure), поддерживающие кросс-платформенные сервисы.

**Преимущества:**

* Снижение зависимости от вендоров.
* Упрощение интеграции новых технологий.

#### ****7. Класс информационных систем и сетей как открытые информационные системы****

Открытые информационные системы - это системы, построенные по стандартизированным принципам, которые позволяют легко интегрировать их с другими системами и расширять функционал. Главные особенности таких систем - это использование общепринятых протоколов и интерфейсов, что делает их независимыми от конкретных производителей.

Например, современные корпоративные системы (ERP, CRM) работают как открытые системы, потому что используют стандартные API для обмена данными. Сети тоже являются открытыми системами - интернет построен на открытых протоколах TCP/IP, HTTP, что позволяет разному оборудованию и программам работать вместе.

Ключевое преимущество открытых систем в том, что они дают свободу выбора: можно комбинировать компоненты разных производителей и легко модернизировать систему.

#### ****8. Модели и структуры информационных систем****

Информационные системы можно описывать разными способами. Функциональные модели показывают, какие процессы происходят в системе. Динамические модели описывают, как система меняется со временем. Информационные модели определяют, какие данные хранятся и как они связаны.

Что касается структуры, системы бывают разными. В централизованных системах всё управляется из одного места - это просто, но ненадежно. Клиент-серверные системы разделяют хранение данных и работу с пользователем. Самые современные - микросервисные системы, где каждая часть работает независимо, что делает систему гибкой и устойчивой к сбоям.

Сейчас главный тренд - переход от больших монолитных систем к гибким микросервисным архитектурам, особенно в облачных сервисах. Это позволяет быстрее обновлять отдельные компоненты и лучше масштабировать систему.

### **9. Информационные ресурсы**

Информационные ресурсы — это организованные совокупности данных, документов и знаний, которые используются для решения задач в различных сферах деятельности. Их можно классифицировать по нескольким критериям:

1. **По форме представления**:
   * Текстовые (документы, книги)
   * Числовые (базы данных, статистика)
   * Графические (чертежи, схемы)
   * Мультимедийные (аудио, видео)
2. **По способу хранения**:
   * Электронные (базы данных, файлы)
   * Бумажные (архивы, библиотеки)
3. **По доступности**:
   * Открытые (общедоступные)
   * Ограниченного доступа (конфиденциальные)
4. **По назначению**:
   * Научные (исследования, публикации)
   * Управленческие (отчеты, документация)
   * Образовательные (учебные материалы)

Особую ценность представляют государственные информационные ресурсы (реестры, кадастры) и корпоративные базы знаний. Современные технологии позволяют эффективно управлять этими ресурсами, обеспечивая их актуальность, сохранность и доступность.

### **12. Компоненты информационных систем**

Любая информационная система состоит из взаимосвязанных компонентов, которые можно разделить на несколько групп:

1. **Аппаратные средства**:
   * Серверы и рабочие станции
   * Сетевое оборудование
   * Периферийные устройства
2. **Программное обеспечение**:
   * Системное ПО (ОС, СУБД)
   * Прикладное ПО (специализированные программы)
   * Инструментальные средства разработки
3. **Информационные ресурсы**:
   * Базы данных и хранилища
   * Знания и документы
   * Метаданные
4. **Человеческие ресурсы**:
   * Пользователи
   * Администраторы
   * Разработчики
5. **Организационные компоненты**:
   * Регламенты работы
   * Методики обработки данных
   * Политики безопасности
6. **Коммуникационные средства**:
   * Сетевые протоколы
   * Интерфейсы взаимодействия
   * Каналы передачи данных

### **17. Классификация ИС по виду информации**

Информационные системы можно разделить на несколько типов в зависимости от вида информации, с которой они работают. Фактографические системы обрабатывают структурированные данные - числа, даты, формализованные показатели. Это могут быть бухгалтерские программы или системы складского учета. Документальные системы работают с неструктурированной текстовой информацией - договорами, отчетами, технической документацией. Геоинформационные системы специализируются на обработке пространственных данных - карт, схем, спутниковых снимков. Существуют также мультимедийные системы для работы с аудио, видео и графикой, а также экспертные системы, оперирующие знаниями в конкретной предметной области.

### **20. Эталонная модель взаимодействия открытых систем**

Эталонная модель взаимодействия открытых систем (OSI) представляет собой концептуальную основу для проектирования сетевых коммуникаций. Она разделяет процесс передачи данных на семь уровней, каждый из которых выполняет определенную функцию. Физический уровень отвечает за передачу битов по каналу связи, канальный - за надежную доставку данных между соседними узлами, сетевой - за маршрутизацию пакетов в сети. Транспортный уровень обеспечивает сквозную надежность, сеансовый - управление соединением, представительный - преобразование данных, а прикладной - интерфейс для конечных приложений. Хотя реальные протоколы (например, TCP/IP) не всегда строго следуют этой модели, она остается важным инструментом для понимания сетевых технологий.

### **1. Текстовые интерфейсы информационных систем**

Текстовые интерфейсы представляют собой способ взаимодействия пользователя с системой через ввод и вывод текстовой информации. Это один из самых старых типов интерфейсов, который до сих пор сохраняет актуальность в определенных сферах. Основная особенность таких интерфейсов - работа в командной строке, где пользователь вводит текстовые команды, а система выдает текстовые ответы.

Преимущества текстовых интерфейсов включают низкие требования к аппаратным ресурсам, возможность автоматизации через скрипты, высокую скорость работы для опытных пользователей. Они широко используются в серверных операционных системах (например, Linux-терминал), системах управления базами данных, сетевом оборудовании (Cisco IOS), а также в специализированном ПО для разработчиков и системных администраторов.

### **2. Смешанные интерфейсы информационных систем**

Смешанные интерфейсы сочетают в себе элементы различных подходов к взаимодействию пользователя с системой. Чаще всего они комбинируют графические и текстовые компоненты, предлагая гибкий способ работы. Например, это может быть графический интерфейс с возможностью ввода команд в специальной строке или консольное приложение с элементами псевдографики.

Такие интерфейсы особенно полезны в профессиональном ПО, где важно сохранить скорость работы через командную строку для опытных пользователей, но при этом обеспечить удобство графического управления для новичков. Примеры включают современные IDE для программистов (Visual Studio Code, Eclipse), системы автоматизированного проектирования (AutoCAD), а также некоторые CRM- и ERP-системы.

### **3. Графические интерфейсы информационных систем**

Графические интерфейсы (GUI) являются наиболее распространенным и интуитивно понятным способом взаимодействия пользователей с информационными системами. Они используют визуальные элементы управления - окна, кнопки, меню, иконки, которые пользователь может активировать с помощью мыши или сенсорного ввода.

Основные преимущества графических интерфейсов включают простоту освоения, наглядность представления информации, возможность визуализации сложных данных. Они доминируют в пользовательских операционных системах (Windows, macOS), офисных приложениях, мобильных приложениях, веб-интерфейсах. Современные тенденции развития включают адаптивный дизайн, поддержку сенсорного управления, использование анимации и интерактивных элементов для улучшения пользовательского опыта.

**4. Многозвенные архитектуры информационных систем.**    
Многозвенные архитектуры делят систему на несколько логических уровней, каждый из которых выполняет свою функцию. Чаще всего используют трехзвенную модель:   
- **Уровень представления (клиентский)** – отвечает за взаимодействие с пользователем (например, веб-интерфейс или мобильное приложение).   
- **Бизнес-логика (сервер приложений)** – обрабатывает данные, выполняет расчеты и управляет процессами.   
- **Уровень данных (СУБД)** – хранит и управляет информацией.   
Бывают и более сложные модели (n-звенные), где бизнес-логика или хранение данных дополнительно разделяются на подуровни. Это повышает масштабируемость, безопасность и гибкость системы.   
**5. "Толстые" и "тонкие" клиенты.**    
"Толстый" клиент выполняет большую часть обработки данных на стороне пользователя. Он требует установки специального ПО, обладает богатым функционалом, но зависит от мощности устройства и сложнее обновляется (пример: десктопные приложения, игры).   
"Тонкий" клиент переносит основную нагрузку на сервер, а на устройстве пользователя работает минимальное ПО (например, браузер). Он легче в поддержке, не зависит от железа, но требует стабильного соединения и ограничен в офлайн-работе (пример: веб-приложения, терминальные решения).   
Существуют также "гибридные" клиенты, сочетающие черты обоих подходов (например, PWA-приложения).