**Вопросы к лабораторной работе №1**

1. **SOA: Service-oriented architecture**, **сервис-ориентированная архитектура**

Парадигма проектирования и разработки программного обеспечения, основанная на применении распределенных слабосвязанных компонентов, обеспечивающих стандартные интерфейсы (открытые протоколы).

1. **Основные свойства**

* независимость от технологий;
* независимость от аппаратной реализации узлов;
* независимость от операционной системы в узлах;
* независимость от языка программирования разработки сервиса;
* масштабируемость.

1. **Сервис**

Видимый ресурс, выполняющий повторяющуюся задачу и описанный внешней инструкцией.

1. **Основные стандарты W3С**:

XML, SOAP, WSDL, UDDI.

1. **Спецификации второго уровня W3C (WS\*):**

* WS-Policy, WS-PolicyAttachment, WS-PolicyAssertion (описание политик web-сервиса);
* WS-Addressing (механизм адресации web-cthdbcf);
* WS-Security (целостность и конфиденциальность web-сервисов);
* WS-Trust (механизм получения маркеров защиты);
* WS-SecureConversion (создание безопасной сессии обмена сообщениями);
* WS-SecurityPolicy (определяет набор утверждений политики безопасности);
* WS-Federation (объединение защищенных доменов);
* WS-Transfer (механизм обновления, создания и удаления ресурсов);
* WS-ResourceTransfer, WS-Fragment (обеспечивает частичный доступ к ресурсам);
* WS-MetadataExchange (механизм получения метаданных);
* WS-Enumeration (механизм получения данных больших размеров);
* WS-Eventing (механизм уведомления о событиях web-сервисов);
* WS-Management (SOAP-управление системами);
* WS-Discovery (механизмы публикации и поиска web-сервисов);
* WS-ReliableMessaging, WS-ReliableMessagingPolicy (механизмы надежной передачи сообщений между web-сервисами);
* WS-MakeConnection (установка соединения с сервисом не имеющего доступный адрес);
* WS-Coordination (механизмы взаимодействия web-сервисов);
* WS-AtomicTransaction (поддержка транзакций web-сервисов);
* WS-BusinessActivity (координация бизнес-взаимодействия web-сервисов).

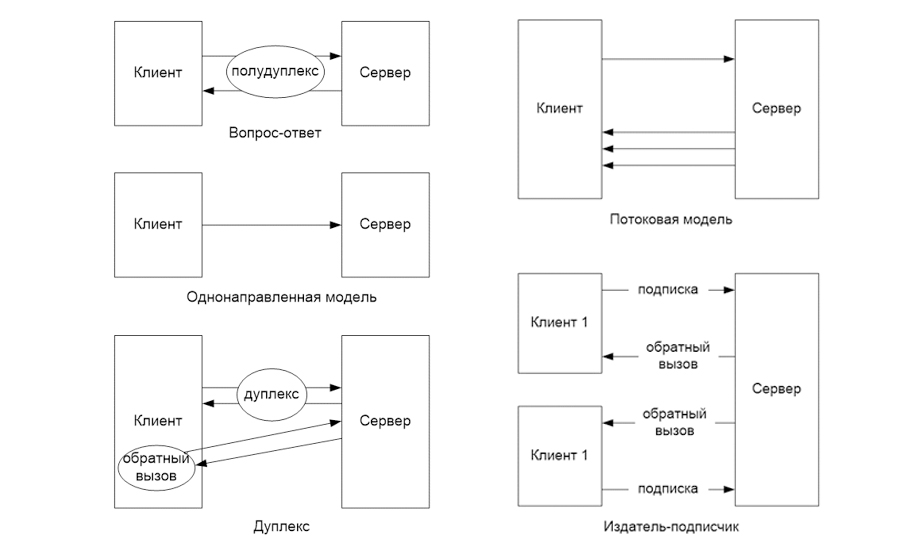
1. **Интерфейсы**

REST, SOAP, JSON-RPC

1. **Специальные компоненты** (как правило часть ESB)

* ***SOA Registry*** — реестр сервисов, информация о сервисах и их интерфейсах;
* ***Workflow Engine*** — программный компонент, предназначенный для того, чтобы построить модель бизнес процесса и выполнить бизнес процесс на основе группы сервисов. Другими словами, разработать новый сервис на основе последовательного выполнения нескольких сервисов;
* ***Service Broker*** — программный компонент, позволяющий трансформировать запрос пользователя в системе в запуск и координированную работу;
* ***SOA Supervisor*** — служебный сервис, предназначенный для управления и мониторинга других служебных сервисов.
* ***Health Check Service*** – служебный сервис для проверки работоспособности других сервисов, часто в составе Supervisor; **Health Check API:** API для проверки работоспособности. Системы мониторинга серверов: Nagios, Zabbiks, Icinga (Nagios Core), dotcom.monitor.
* ***Identity Service* -** сервис аутентификации, предназначен для реализации SSO (Single Sign On), протоколы токен-авторизации (JWT), OpenID Connect, OAuth2. IDaaS. (AWS IAM).

1. **Способы клиент-серверного взаимодействия**

****