**Открытые информационные системы**

**Свойства открытости:**

* Мобильность ПО на различные аппаратные платформы;
* Возможность модификаций;
* Интегрируемость.

**Аспекты открытости отражены в стандартизации:**

* API {Application Program Interface) - интерфейсов прикладных программ с операционным окружением, в том числе системных вызовов и утилит операционной системы (ОС), т.е. связей с ОС;
* Межпрограммного интерфейса, включая языки программирования;
* Сетевого взаимодействия;
* Пользовательского интерфейса, в том числе средств графического взаимодействия пользователя с ЭВМ;
* Средств зашиты информации.

**Введение в SOA**

**SOA** — это архитектурный подход к определению, связыванию и интеграции повторно используемых бизнес-сервисов, имеющих четкие границы и самодостаточных по своей функциональности.

**Что лучше всего подходит для SOA**

* Централизованные бизнес-функции, используемые несколькими субъектами;
* Интеграция с партнерами;
* Наличие работающих старых технологий.

**Чем обусловлена быстрая настройка под меняющиеся бизнес-требования**

**Слабое связывание:**

1. Устраняет жесткие связи, препятствующие изменениям.
2. Меньше вложений в реализацию и больше в повторное использование.
3. Улучшает возможности удаленного доступа к оригинальным источникам информации, уменьшая задержки и зависимости.
4. Проекты по интеграции управляются бизнес-требованиями (то есть бизнес-деятельность является основной движущей силой).
5. Благодаря отображению и совместному использованию информации, слабое связывание позволяет компаниям извлекать в режиме реального времени больше данных об эффективности бизнес-деятельности.
6. Облегчает партнерам взаимодействие с вашей компанией.
7. Способствует продвижению и публикации ваших сервисов, облегчая клиентам обнаружение их и вашей компании.
8. Облегчает поиск новых партнеров и сервисов, помогая найти более подходящий под ваши требования сервис.

**Повторное использование:**

1. Делает процессы более согласованными, поскольку они базируются на одних и тех же компонентах.
2. Способствует повышению качества благодаря конкуренции между провайдерами сервисов.
3. Позволяет изменять систему независимо от изменений бизнес-деятельности.
4. Уменьшает влияние изменений, поскольку они выполняются централизовано и охватывают все участвующие стороны.

**Расширяемость:**

1. Делает SOA-решения доступными организациям любого размера.
2. Изменяет процесс разработки на более динамичный, более подходящий для ведения бизнес-деятельности.
3. Ускоряет слияния и поглощения.

**В каких случаях применение SOA не обосновано**

1. Когда информационная среда гомогенна;
2. Когда критична производительность в режиме реального времени;
3. Когда ничего не меняется;
4. Когда тесное связывание не является недостатком.

**Концепция SOA**

**Сервис** — это функция, являющаяся четко определенной, самодостаточной и не зависящей от контекста или состояния других сервисов.

**Концепция слабого связывания в SOA:**

* Виртуализация сервиса (Service Mediation);
* Независимость местоположения;
* Независимость передачи данных;
* Независимость сообщений.

**Типовые функции виртуального сервиса**

**Виртуальный сервис** — наилучшее место реализации некоторых технических условий или обеспечения качества сервиса (QualityOfService):

1. Проверка XML сообщений на корректность формата и соответствие интерфейсу сервиса.
2. Аутентификация и авторизация: идентификация потребителя сервиса и проверка наличия у него прав для вызова сервиса.
3. Расшифровка сообщений и проверка подписи.
4. Балансировка нагрузки и гарантии наличия ресурсов для работы сервиса.
5. Маршрутизация сообщений. Передача сообщений различным реализациям сервиса в зависимости от содержимого сообщений или внешних условий.
6. Мониторинг работы сервиса, производительности, а также проверка предоставления поставщикам требуемых услуг (SLA)

**Что такое слабое связывание и его значение:**

* Сущность связывается, если ее изменение одной взаимодействующей стороной требует изменений другими сторонами (например, модели данных).
* Сущность объявляется, если ее поведение определяется в интерфейсе к сервису, а инициаторы запросов и провайдеры могут взаимодействовать только при совпадении объявленного поведения.
* Сущность преобразовывается, если она объявляется как инициаторами запросов, так и провайдерами сервисов, но инициаторами и провайдерами заявлено различное поведение сущности и инфраструктура обеспечивает определенные возможности преобразования, позволяющие наладить взаимодействие.
* Сущность согласовывается, если как инициатор запросов, так и провайдер сервиса объявляют спектр поведений, которые они могут поддерживать, а посредническая инфраструктура может договориться о согласовании их поведения для каждого взаимодействия.
* Сущность развязывается, если изменения аспекта одной взаимодействующей стороной не требуют соответствующих изменений другими сторонами.

**Слабое связывание проявляет себя в парадигме SOA следующим образом:**

* Оно помогает организовать уровень абстракции между производителями и потребителями сервисов.
* Оно способствует реализации гибкости в изменении реализации сервисов без воздействия на потребителей сервисов.
* В архитектуре SOA функциональность организуется как набор модульных повторно используемых общих сервисов. Эти сервисы имеют четко определенные интерфейсы, инкапсулирующие ключевые правила доступа к ним. Они также строятся без каких-либо допущений о том, кто будет использовать или потреблять эти сервисы. Таким образом, они слабо связаны с потребителями сервисов.