

## Открытые системы. Сервис-ориентированная архитектура

### Открытые информационные системы

Свойство открытости означает:

- 1) переносимость (мобильность) ПО на различные аппаратные платформы;
- 2) приспособленность системы к ее модификациям (модифицируемость или собственно открытость);
- 3) приспособленность системы к комплексированию с другими системами в целях расширения ее функциональных возможностей и (или) придания системе новых качеств (интегрируемость).

*Реинжиниринг* - замена длительной и дорогостоящей разработки новых систем по одному циклу их компоновкой из ранее спроектированных подсистем или быстрой модернизацией уже существующих систем.

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 защиты информации.

*Профилем* открытой системы называют совокупность стандартов и других нормативных документов, обеспечивающих выполнение системой заданных функций.

## Введение SOA

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

Обратится к SOA нужно в следующих ситуациях.

1. Централизованные бизнес-функции, используемые несколькими субъектами.
2. Интеграция с партнерами.
3. Наличие работающих старых технологий.

*Подвижность бизнеса или agility* - способность адаптироваться к изменениям и быть готовым к ним.

Адаптация происходит благодаря:

- слабому связыванию;
- повторному использованию;
- расширяемости.

Применение SOA не обосновано, когда:

- информационная среда гомогенна;
- критична производительность в режиме реального времени;
- ничего не меняется;
- тесное связывание не является недостатком.

## Концепции SOA

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

Концепция слабого связывания подразумевает:

- виртуализацию сервиса;
- независимость местоположения;
- независимость передачи данных;
- независимость сообщений.

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

### Типовые функции виртуального сервиса

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).

### Принципы слабого связывания

- 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. К объявляемым аспектам относятся система защиты, транзакционное поведение и качество сервиса (например, время реакции и доставки).
- 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 в концепции SOA способствует:

- организации уровней абстракций между производителями и потребителями сервисов;
- реализации гибкости в изменениях сервисов без воздействия на потребителей сервисов;
- организации интерфейсов и инкапсуляции сервисов;
- определению того, кто будет использовать или потреблять сервисы.