

- 1. Каковы бизнес цели и задачи переезда системы на новую архитектуру?**
Определение правильных бизнес целей во многом является основополагающим при выборе той или иной архитектуры для проекта.
- 2. Каков бюджет на расширение инфраструктуры и персонала?**
Может ли компания себе позволить расширить штат сотрудников, например, dev-ops инженеров, или увеличить финансирование на реализацию системы шардирования СУБД для более быстрой скорости отклика?
- 3. Каким софтверным решениям наиболее отдано предпочтение: Open Source или лицензированным?**
Здесь может быть во главе угла какая-либо политика безопасности, по которой не используются Open Source продукты, либо наоборот, сокращенное финансирование, не позволяющее использовать дорогие решения.
- 4. Данные как сервис или данные как продукт?**
В зависимости от того, какое концептуальное устройство данных в рамках текущей компании, принимаются решения об архитектуре. Сервис непрерывен. Работает, получая и обрабатывая данные. Продукт напротив, статичен до выхода последующего обновления. В нашем же случае, скорее всего это именно сервис.
- 5. Есть ли какие-либо особенности в текущем структурировании данных?**
Крайне важно, чтобы хранение в новой системе было без «костылей». Для этого все ошибки и недочеты в структурировании данных мы должны учесть в текущей системе.
- 6. Экономика данных. Требуется ли интеграция к внешним сервисам? Если да, производим интеграцию через непрерывное получение данных или асинхронное?**
От реализации интеграции к внешним сервисам зависят многие характеристики архитектуры приложения. До и в целом, сложно представить себе веб сервис розничной торговли, существующий в «вакууме».
- 7. Трансформация данных. Есть ли дубликаты в текущих БД?**
Если да, делаем дедубликацию. Один из вопросов, относящихся к правильному структурированию данных.
- 8. Трансформация данных. Есть ли какие-то особенности в текущем выборе формата хранения данных?**
Если да, возможно удастся максимизировать ценность БД, сменив формат в одном из компонентов системы, проведя тем самым оптимизацию при обновлении.
- 9. Каков жизненный цикл данных в рамках текущей компании?**
Жизненный цикл включает в себя процессы, которые осуществляют перемещение, хранение, преобразование, а также включают в себя обслуживание данных и предоставление совместного доступа. Оптимизация этапов жизненного цикла в новой системе поможет максимизировать ценность данных и, возможно, сократить затраты компании на их обслуживание. Что несомненно является успехом правильно выбранной архитектуры.

10. Какова текущая дата архитектура?

Будь то корпоративное хранилище данных, или централизованная БД. Архитектор должен понимать все достоинства и недостатки текущего решения для его оптимизации, либо при построении нового решения.

11. Какой объем данных ожидается?

Возможно, имеет смысл обратить внимание на СУБД с легкой масштабируемостью при обновлении системы, чтобы сократить издержки на последующем обновлении нашей системы.

12. Каков текущий язык программирования?

Возможно переписать систему на новом и современном языке программирования будет проще и дешевле, ввиду большего выбора фреймворков и библиотек. Что так же положительно скажется на архитектуре в целом.

13. Какие функции безопасности должны быть предусмотрены?

Возможно, имеются определенные требования к хранению и шифрованию данных о пользователях и их паролях. Возможно, в старой системе имелись какие-либо ошибки и недочеты в функциях безопасности, архитектурная оптимизация которых позволит сократить материальные и репутационные издержки.

14. Какие пожелания бизнеса?

Например, максимальная простота внешнего сервиса или сжатые сроки разработки. Зачастую от пожеланий заказчика зависит выбор сложности и устройства той или иной архитектуры.