

МЕХАНИЗМЫ ИНТЕГРАЦИИ СИСТЕМ

Абрамов В.В.

ГОУВПО «Мордовский государственный университет им. Н. П. Огарева»

Тел. (926) 858 8832. E-mail: angelinfo@yandex.ru

Аннотация: В статье рассматриваются схемы и типы интеграции информационных систем. Рассматривается алгоритм принятия решения о выборе концепции интеграции и основные преимущества и недостатки рассмотренных решений.

Ключевые слова: интеграция систем, схемы интеграции, линейная интеграция, комплексная интеграция, обмен данными, файловый обмен, база данных, обмен сообщениями.

Annotation: The article discusses the scheme and the types of information system integration. An algorithm for deciding on the concept of integration and the advantages and disadvantages of the considered solutions.

Keywords: systems integration, schema integration, line integration, complex integration, data sharing, file exchange, database, messaging.

Введение

В настоящее время, информационные системы все больше проникают в систему управления предприятием. При этом специфичные программные продукты используются в соответствующих областях. В виду этого, задача интеграции существующих систем в единый ландшафт является важной, решение которой позволяет выполнять управление и учет в глобальном масштабе, избежать дублирования данных, обеспечить целостность и непротиворечивость данных. В целом эффективность ИТ-инфраструктура зависит от уровня интеграции существующих систем.

Постановка

Так как отдельные системы являются звеньями единой цепи, и работа последующих элементов сильно зависит от входящих данных из предыдущих систем, то мы имеем дело со сквозными процессами, которые являются основополагающими для полной автоматизации работы. В виду этого, мы должны иметь методологию для выбора способа интеграции систем с учетом собственной специфики.

1. Схемы интеграции систем

В целом, процесс интеграции может быть построен с использованием: Односторонней интеграции

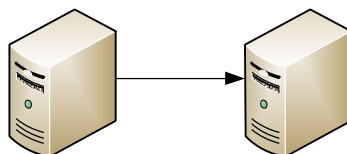


рис. 1

Плюсы	Минусы
Низкие трудозатраты на реализацию	Отсутствуют механизмы контроля. Невозможно определить статус результата процесса в системе выгрузки.
	В случае сбоя, получить информацию можно будет только с определенной временной задержкой, равной интервалам между сеансами выгрузки, или инициализировать вручную.

Двухсторонней интеграции

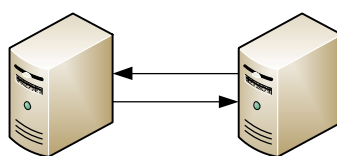


рис. 2

Плюсы	Минусы
Реализация механизмов мониторинга. Прозрачность процесса.	Высокие трудозатраты по сравнению с односторонней интеграцией.
Стресс-устойчивость – реализуется за счет механизмов обратной связи.	Необходимость совместимости протоколов в 2х направлениях.

В настоящее время более распространен механизм двухсторонней интеграции, что обусловлено наличием протоколов высокого уровня поддерживающих обмен информацией в обоих направлениях и так же требованиями к целостности данных.

Далее переходим непосредственно к рассмотрению схем интеграции.

Вариант 1 - Линейная интеграция.

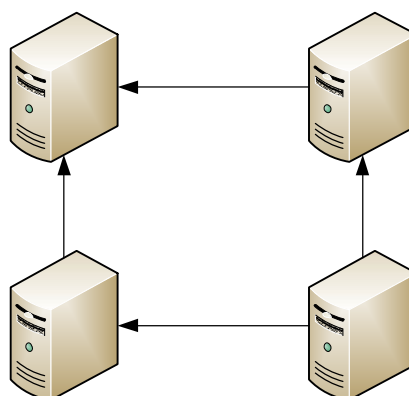


рис. 3

Данная схема взаимодействия используется, как правило, в случае обмена данными с большим временным интервалом (день, месяц) или в качестве временного решения. Данная схема не позволяет оперативно реагировать на сбои при передаче и гибко расширять интеграцию на другие системы входящие в цепочку.

Вариант 2 - Интеграция по типу звезда.

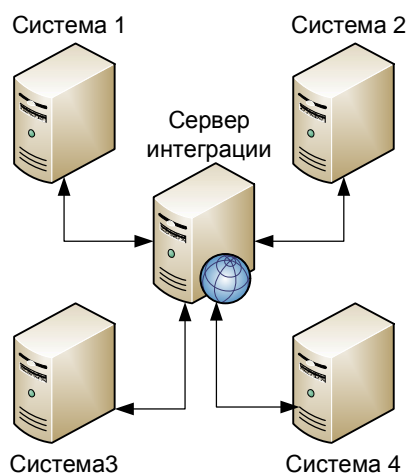


рис. 4

Отличительной особенностью данной схемы является наличие сервера интеграции. Сервер имеет обширный набор интерфейсов для взаимодействия с различными системами в обоих направлениях. Кроме этого, на сервере реализованы механизмы контроля и мониторинга процесса интеграции.

В настоящее время ядром сервера интеграции являются решения на основе SOA или программные платформы с большим набором интерфейсов, протоколов, API.

Интерфейс компонентов SOA-программы предоставляет инкапсуляцию деталей реализации конкретного компонента (ОС, платформы, языка программирования, вендора, и т. п.) от остальных компонентов. Таким образом, SOA предоставляет гибкий и элегантный способ комбинирования и многократного использования компонентов для построения сложных распределённых программных комплексов. [1]

Вариант 3 - Комплексная интеграция.

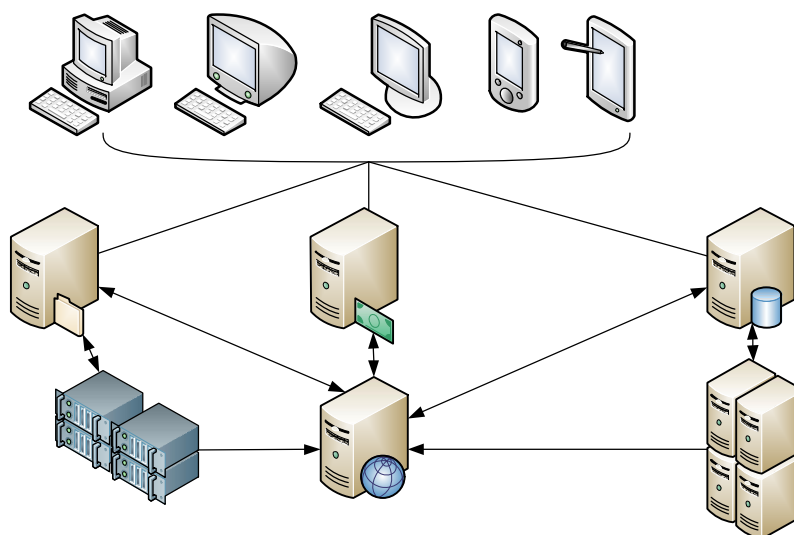


рис. 5

В комплексной интеграции использованы элементы линейной интеграции и интеграции по типу звезда. Данная схема обеспечивает наилучшую масштабируемость информационной среды. Кроме этого, использованы все плюсы централизации с сохранением возможности прямого соединения, что может быть необходимо в случае ограничений связанных с каналом связи и параметрами безопасности.

2. Типы обмена данными

1) Файловый обмен

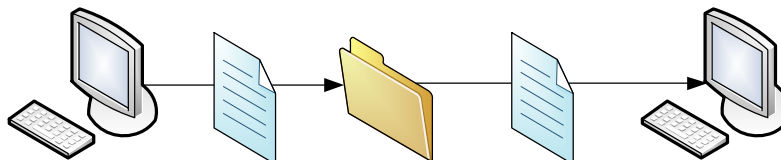


рис. 6

Основной принцип - системы экспортируют данные, пригодные для импорта в другие системы. В качестве формата данных, как правило, выбирают плоские файлы или XML файлы.

Преимуществом плоских файлов:

Универсальность

Отсутствие обработки при импорте

Преимущества XML[2]:

Наличие стандартов, описывающих формат документа.

Поддержка Unicode.

Наличие средств разбора.

Наличие средств преобразования документа.

Платформно-независимы.

В настоящее время использование XML файлов становится новым стандартом, в виду его официальной поддержки такими производителями ПО как Microsoft, Oracle, SUN и т.д. Данные производители включают в свои продукты средства работы с XML файлами и обеспечивают поддержку версий.

Недостатками данного типа обмена данными является преобразование форматов файлов, которые должны быть адаптированы для совместимости. Далее написание инструментов мониторинга процесса для обеспечения контроля, если данные инструменты отсутствуют. В целом, главным плюсом данного типа обмена является простота, так как все современные системы могут принимать файлы. Главным минусом является ненадежность и асинхронность, так как когда файл выгружен, целевая система узнает об этом только после опроса директории, а в случае потери файла, придется принудительно инициализировать повторный экспорт. Так же стоит учесть, что время на обращение к файловой системе при частом опросе директории будет значительно уменьшать скорость работы.

2) Общая база данных

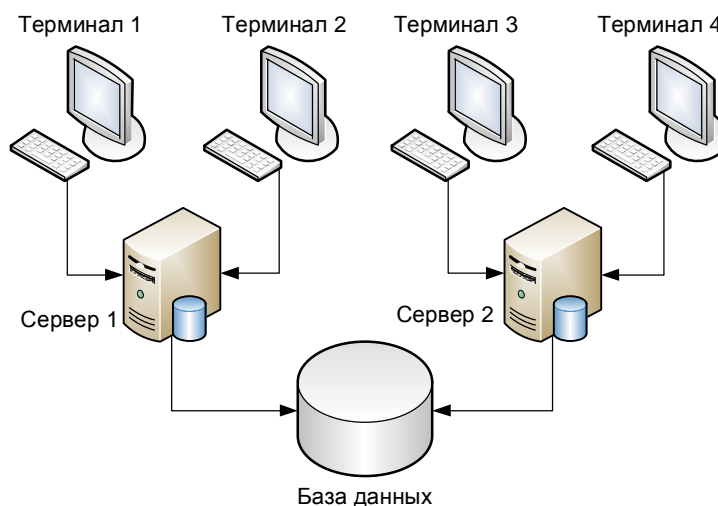


рис. 7

Основной принцип – создается полностью интегрированная информационная среда на основе единой базы данных. Таким образом, мы поддерживаем абсолютную актуальность данных и простоту доступа. Контроль данных единый для всех и осуществляется на уровне базы данных.

Плюсы: Удобство использования для систем ландшафта.

Минусы: Невозможность реализации в случае большого количества систем, в виду специфических требований к базам данных, которые будут конфликтовать между собой. (Например, многие системы работают только с MS SQL, в то время как другие только с Oracle). Проблемы объединения уже существующих систем.

3) Вызов удалённых процедур

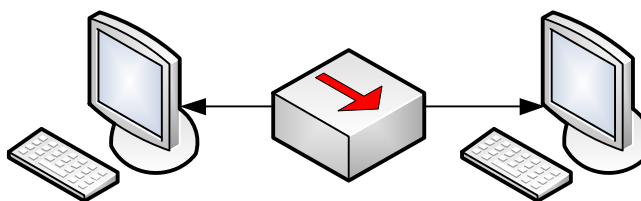


рис. 8

Данная технология позволяет компьютерным программам вызывать функции или процедуры в другом адресном пространстве (как правило, на удалённых компьютерах). Таким образом, мы можем управлять действиями в другой системе через заранее предусмотренный для этого интерфейс.

Основной принцип состоит в расширении хорошо известного и понятного механизма передачи управления и данных внутри программы, выполняющейся на одной машине, на передачу управления и данных через сеть. Средства удалённого вызова процедур предназначены для облегчения организации распределённых вычислений и

создания распределенных клиент-серверных информационных систем. Наибольшая эффективность использования удаленного вызова процедур достигается в тех приложениях, в которых существует интерактивная связь между удалёнными компонентами с небольшим временем ответов и относительно малым количеством передаваемых данных.

Характерными чертами вызова локальных процедур являются:

Асимметричность, то есть одна из взаимодействующих сторон является инициатором;

Синхронность, то есть выполнение вызываемой процедуры приостанавливается с момента выдачи запроса и возобновляется только после возврата из вызываемой процедуры.[3]

Плюсы: скорость работы и синхронность выполнения.

Минусы: требование по наличию необходимых процедур интеграции и описания правил доступа к ним.

4) Обмен сообщениями

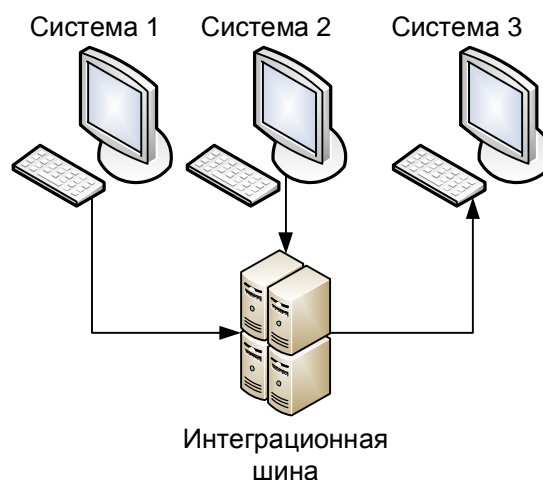


рис. 9

Основной принцип – интеграция через интеграционную шину, которая берет на себя задачу адаптации и преобразования данных. При данном подходе, мы сводим доработки в существующих системах к минимуму.

Плюсы: поддержка широкого спектра протоколов интеграционной шиной. Минимальные изменения в существующих системах. Масштабируемость и универсальность существующего ландшафта.

Минусы: наличие интеграционной шины, невозможно обойтись имеющимися системами. (Если только интегрированные системы дополнительно не являются интеграционными шинами).

Вывод:

В данной работе осуществляется подробный анализ существующих схем и видов интеграции. Описание схем и типов интеграции построено таким образом, что позволяет на этапе проектирования процесса обмена данными выбрать наиболее подходящую схему и реализацию интеграции, учитывая описанные минусы и плюсы.

Список использованной литературы:

[1] http://ru.wikipedia.org/wiki/Сервис-ориентированная_архитектура

[2] <http://shedar.info/2007/07/02/xml-preimushhestva-i-nedostatki/>

[3] http://ru.wikipedia.org/wiki/Remote_Procedure_Call