

Список использованных источников:

1. Jonathan Lewis, Cost-Based Oracle Fundamentals;
2. Tariq Farooq, Charles Kim, Nitin Vengurlekar, Sridhar Avantsa, Guy Harrison, Syed Jaffar Hussain, Oracle Exadata Expert's Handbook;
3. Oleg Ivanov, Sergey Bartunov, Adaptive Cardinality Estimation.

СЕРВИСЫ РАСШИРЕНИЯ ФУНКЦИОНАЛА В ОБЛАЧНЫХ ТЕХНОЛОГИЯХ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Янчевский В.И.

Селезнёв И.Л. – к. техн. н, доцент

В докладе рассматривается построение локального облака (private cloud) с обеспечением высокого уровня безопасности данных, предоставляющего функционал веб-сервиса, которое используются в системах с SOA (сервис-ориентированной архитектурой); локальное облако предоставляет услуги через программные интерфейсы SOAP и REST другим веб-сервисам или приложениям-клиентам. Предлагается принцип построения многосервисной облачной структуры, как совокупности набора элементов, представляющих функционально-независимые многопользовательские облачные сервисы, отвечающие требованиям архитектуры SOA.

В процессе развития облачных технологий стало ясно, что концепция облака гораздо шире, чем просто использование виртуальных вычислительных ресурсов. Следуя за спросом, требуется разрабатывать более широкий функционал решений для облачных технологий, включая инструменты для полноценного управления виртуальными машинами и базовыми сценариями.

SOA (сервис-ориентированная архитектура) рассматривается в виде совокупности веб-сервисов, часто строящихся как распределённые системы и работающие на разных платформах. Веб-сервисы могут взаимодействовать как друг с другом, так и с приложениями, созданными на основе SOA, посредством сообщений. Эти сообщения передаются стандартными протоколами, которые получили наибольшее распространение – SOAP (протокол обмена xml-данными) и REST (стиль архитектуры программного обеспечения для распределённых систем), в формате (языка разметки) XML и/или JSON.

В качестве транспорта сообщений используют протокол HTTP. В SOAP применяется специальный язык описания веб-сервисов и доступа к ним (WSDL) в формате XML. В REST нет такого описания типа, так как используется фиксированный набор методов доступа к веб-сервисам: GET, POST, PUT, DELETE. Указанные веб-сервисы основаны на открытых международных стандартах, которые поддерживаются современными операционными системами.

Программное обеспечение как сервис (Software software as a service, SaaS) является перспективным направлением в облачных технологиях стимулирующим развитие архитектуры SOA.

SaaS предполагает, что пользователь получает в распоряжение ПО, функции которого доступны через веб-интерфейс, в то время, как основная программная часть приложения находится на сервере разработчика. (Рис.1)

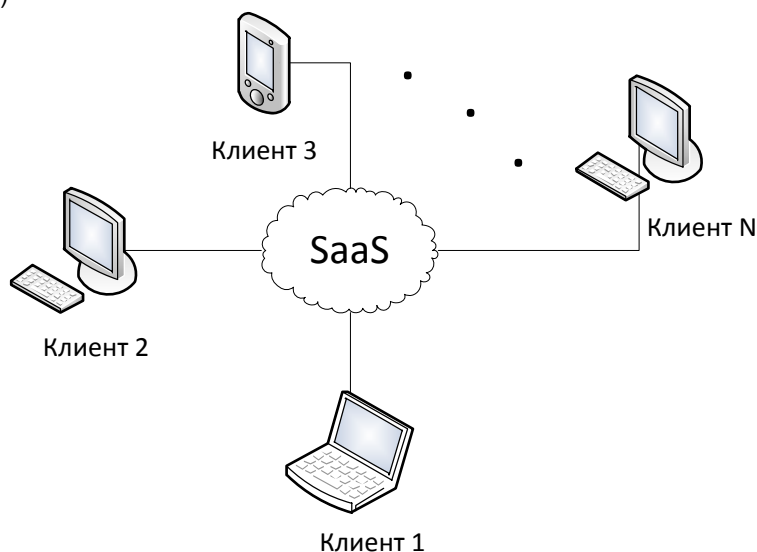


Рисунок 1 – Однооблачная структура сервиса SaaS

Наиболее перспективными средствами интеграции приложений SaaS в системы с архитектурой SOA

являются специализированные средства интеграции приложений — платформы интеграции SaaS на базе PaaS (Platform as a Service), что предполагает использование аппаратных компонентов на стороне поставщика услуг — например, дисковых накопителей или процессора. В PaaS для взаимодействия сервисов используется Middleware (ПО среднего слоя) как связующее ПО.

Платформа PaaS, на базе которой функционируют приложения SaaS может обеспечить средства для интегрирования веб-сервисов, предоставлять сервер приложений или веб-сервер для размещения интегрированной веб-службы (SaaS mashup), а также может организовать систему управления SOA для совместного использования ресурсов SaaS.

На рис. 2 представлен пример сервис-ориентированной архитектуры (SOA) интегрированной службы или SaaS mashup, которая использует в качестве источников информации сторонние (SaaS-1...SaaS-3) и собственные веб-сервисы (SaaS-4, SaaS-5).

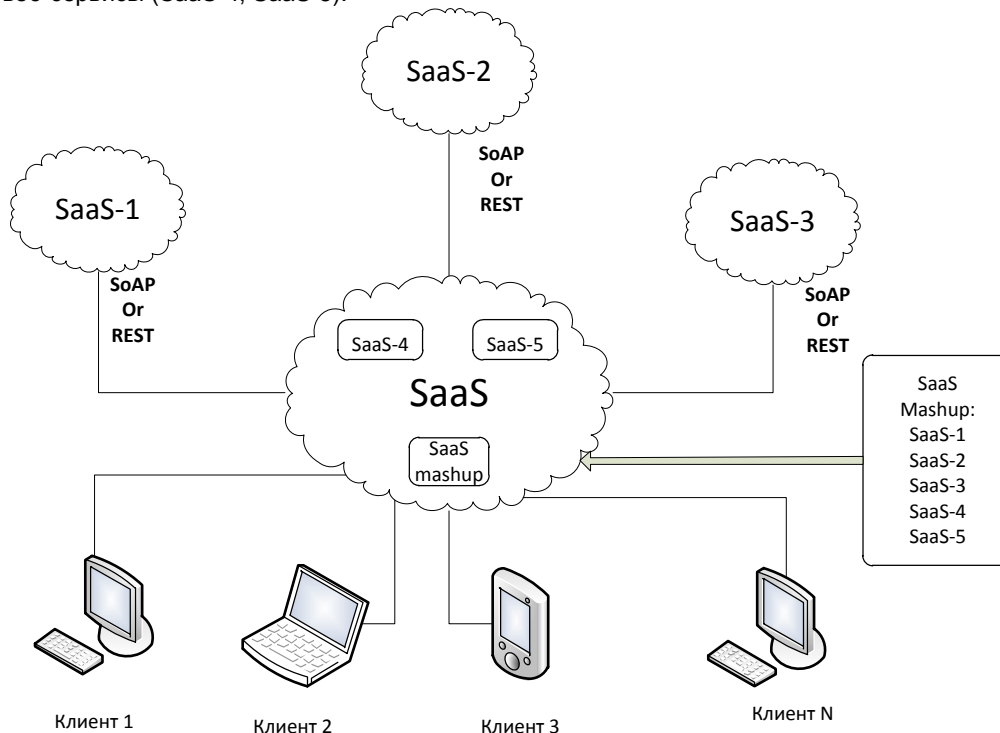


Рис.2 – Многооблачная структура системы SaaS

Веб-сервисы (SaaS-1...SaaS-5) взаимодействуют с интегрированным веб-сервисом SaaS mashup через Middleware. SaaS mashup или связанное распределенное приложение размещается на сервере приложений (App server) платформы PaaS, доступ к этому сервису осуществляется через пользовательский интерфейс. Таким образом, сервис-ориентированное распределенное приложение SaaS mashup представляет собой результат интегрирования веб-служб (SaaS-1...SaaS-5) в одно, логически завершенное, связанное приложение.

К примеру, можно разработать собственный SaaS-сервис, взаимодействующий с github репозиторием.

Современная парадигма построения IT-инфраструктуры для управления бизнес-процессами ориентирована на динамическую (гибкую), а не на статическую (жесткую) инфраструктуру систем, которые могут своевременно реагировать на изменение рыночных условий или особенностей ведения бизнеса. Отсюда следует, что IT-инфраструктура должна проектироваться как приложение с сервис-ориентированной архитектурой в виде набора веб-сервисов, так как именно архитектура SOA обеспечивает требуемую адаптацию к изменяющимся условиям ведения бизнеса.

Сервис-ориентированная архитектура приложений обеспечивает возможность переориентации их конфигурации на решение новых задач бизнеса. Таким образом, принципы сервис-ориентированных архитектур являются наиболее современным способом интеграции крупномодульных, слабосвязанных и общедоступных по сети удаленных веб-сервисов. Помимо этого, системы на основе SOA способствуют развитию BPM (Business Process Management) как класспрограммного обеспечения для управления бизнес-процессами и административными регламентами.

Список использованных источников:

1. Веб-сервисы и облачные вычисления [Электронный ресурс]. – Режим доступа:<http://www.lessons-tva.info/archive/nov032.html>. – Дата доступа: 01.04.2018.
2. Янчевский В.И., Стасилевич М.Н. Эффективное описание границ объектов постоянной яркости: 51-я научная конференция аспирантов, магистрантов и студентов по направлению 8: Информационные системы и технологии: 18.04.15. Минск, 2015. С. 19.
3. REST [Электронный ресурс]. – Режим доступа:<http://blogger.sapronov.me/2014/02/rest.html>. – Дата доступа: 05.04.2018.