

УДК 004.75

Сравнительное нагрузочное тестирование интеграционных решений для банковских систем

***Шилин Г.М.**, студент*

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Информационные системы и телекоммуникации»*

Научный руководитель: Недашковский В.М., к.т.н, доцент

Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана

ymniu3@bmstu.ru

1. Введение

Тенденция роста рынка банковских услуг, всё большая автоматизация этой отрасли, а также повышение качества сервиса приводят к появлению в банке все новых информационных систем. Как следствие — возникает задача их интеграции как между собой, так и с остальными банковскими системами. Появляется необходимость обеспечивать согласованную работу большого количества разнородных приложений, которые в свое время создавались без учета каких-либо требований к их возможной интеграции. Это влечет за собой увеличение расходов на содержание и эксплуатацию плохо совместимых друг с другом приложений, а также времени на обработку бизнес-задач, тогда как надежность системы снижается в связи с возросшим количеством возможных точек отказа [1].

Существует много подходов к решению этой проблемы, однако все больше компаний и банков склоняются в сторону использования интеграционной шины (*Enterprise Service Bus*) — передовой на сегодняшний день технологии в области интеграции бизнес-процессов и приложений [3].

Решая задачу интеграции, важно уделять внимание не только функциональным возможностям, но и нагрузочным показателям, так, чтобы шина при своём полном функциональном покрытии не стала «узким местом» в плане пропускной способности [2]. При выборе вендора интеграционного решения необходимо обращать внимание, в том числе, на то, на каких платформах работают другие системы банка, с какими ИТ-компаниями банк поддерживает партнерские отношения, а также руководствоваться стратегическими целями банка. Поэтому для исследования были выбраны три

программных продукта от лидеров рынка интеграционных решений и проведено их сравнительное нагрузочное тестирование.

2. Цели тестирования

Нагрузочное тестирование (НТ) проводилось на примере первоначального процесса обработки кредитной заявки с целью выявить и сравнить показатели производительности интеграционных решений (интеграционных шин), разработанных на базе трех программных продуктов:

- *Oracle SOA Suite 11g*;
- *Oracle Service Bus*;
- *IBM Integration Bus*.

Одной из задач тестовых испытаний было оценить работу сервера в режиме максимальной (пиковой) нагрузки с количеством вызовов сервисов, эквивалентным количеству, которое было получено в результате предыдущего тестирования, состоявшегося в ноябре 2012 года в лаборатории IBM (г. Монпелье, Франция) [1]. Тогда проводилось нагрузочное тестирование автоматизированной банковской системы *RS-Bank V.6*, фронт-офисного решения *InterBank FrontOffice* и интеграционной платформы *Business Universe RS* от известного российского поставщика банковского ПО *R-Style Softlab*. Показатели тестирования в *IBM* были получены при выполнении полного тестового процесса, предусматривающего вызов 21 сервиса (из них 10 синхронных) на одну кредитную заявку (на один экземпляр процесса). В нашем случае для тестирования был выбран сокращенный сценарий, предполагающий вызов 8 сервисов (из них 4 синхронных).

Экземпляр процесса состоит из следующих сервисов:

- сервис получения данных по клиенту, синхронный;
- сервис сохранения клиента (вызывает сервис поиска клиента внутри себя), синхронный;
- сервис создания кредитного договора (внутри себя вызывает сервисы создания СПИ, графика платежей и счета), асинхронный.

С учётом того, что часть запросов на шине отрабатывается асинхронно или во время вызова другого сервиса, получается, что процесс включает 8 запросов/договоров (1+1+2+4), из них 4 асинхронных. Целевые показатели нагрузочного тестирования приведены в таблице 1.

Целевые показатели нагрузочного тестирования

Требование	Показатель
150 000 вызовов сервисов	19000 заявок/час

Описываемые далее работы по нагрузочному тестированию интеграционных решений, комплекса *Business Universe RS* и его компонентов также выполнялись на базе продуктов компании *R-Style Softlab* на представленном ею оборудовании и в ее собственной лаборатории.

3. Нагрузочное тестирование интеграционных решений

В рамках нагрузочного тестирования проводился запуск эмуляции сокращенного процесса выдачи кредита в *InterBank FrontOffice* в несколько этапов с повышением нагрузки, то есть с увеличением числа параллельных потоков, эмулирующих одновременную работу многих пользователей. Также происходила эмуляция процесса взаимодействия с конечными системами (бэк-офисная система автоматизации кредитования *RS-Loans V.6*). Вместо реальных конечных систем использовались «заглушки». При тестировании ориентировались на оценочные данные по количеству вызовов интеграционных сервисов шины в рамках одного экземпляра процесса, соответствующие среднестатистическим показателям при работе оператора с модулем *InterBank FrontOffice*. Эмуляция процесса и «заглушки» реализованы с помощью *SoapUI*. Продолжительность выполнения каждого отдельного бизнес-сервиса шины фиксировалась. Время работы каждого этапа теста ограничивалось 60 минутами. Тест прерывался в случае возникновения ошибок на шине.

Тестирование проводилось в идеальных условиях (т.е. с «заглушками» внешних систем). Для правомерности сравнения результатов тестов использовался один и тот же физический сервер и единый функциональный сценарий, для чего все три приложения и соответствующие базы данных были установлены на одном физическом сервере (характеристики сервера приведены в таблице 2). При выполнении тестов для одного приложения остальные простаивали, не нагружая ресурсы сервера.

Характеристики сервера

ОС	<i>Windows Server 2008 R2 Enterprise</i>
Тип процессора	<i>Intel Xeon E5645 2.4GHz</i>
Количество ядер на процессоре	4
Количество процессоров	2
Объем ОЗУ	8 Гб
Объем HDD	89 Гб

Программная архитектура *ESB*, которая связывает между собой другие системы, входящие в единый программный комплекс, изображена на рисунке 1, и включает в себя:

- сервер базы данных (БД);
- сервер приложений.

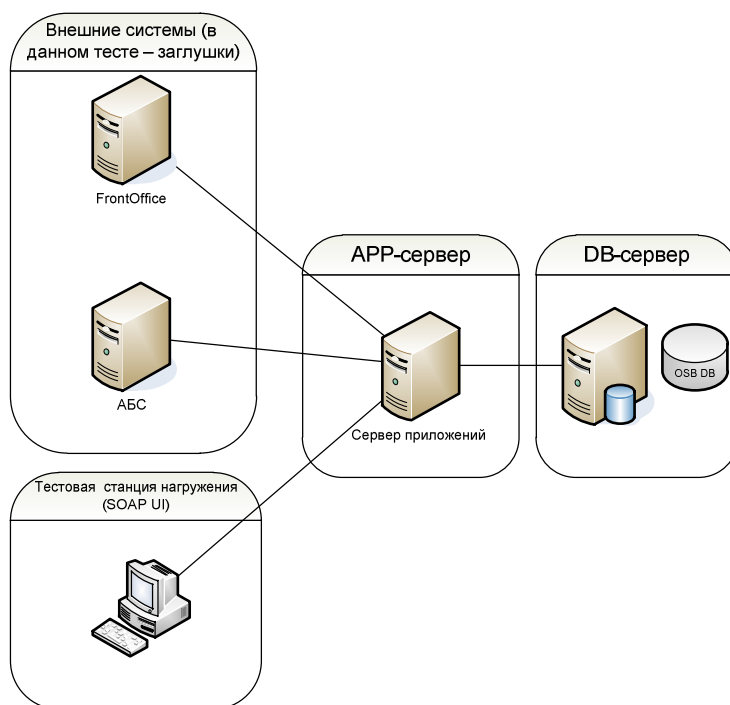


Рис. 1. Принципиальная схема размещения компонентов *ESB*

Сценарий последовательно (согласно очередности вызова в *InterBank FrontOffice*) запускает сервисы выдачи кредита. Между вызовами сервисов были выставлены временные задержки в 5 секунд, что соответствует максимальной интенсивности работы пользователей в *InterBank FrontOffice*.

Эмуляторы («заглушки») реализуются сервисами с минимальной логикой (или вообще без нее – в идеальном случае) в *SoapUI* посредством *Mock*-сервисов. Часть «заглушек» вызывает базу данных для получения идентификаторов, которые затем возвращаются в ответе *Mock*-сервиса. Ответ *Mock*-сервиса зависит от параметра *methodName* в запросе, логика ответа реализована с помощью скрипта в *SoapUI*.

В ходе тестирования замерялось количество выполненных за отведенное время сервисов для заданного количества потоков, определялось время выполнения всех сервисов (включая асинхронные, которые могли обрабатываться с запозданием в случае переполнения очередей) и время выполнения каждого сервиса в отдельности (минимальное, среднее, максимальное). Синхронная часть отслеживалась по данным *SoapUI*, а время выполнения асинхронных сервисов записывалось в БД.

Кроме того, были зафиксированы графики загрузки ЦП и памяти сервера, расходуемые *ESB*-приложениями.

4. Результаты тестирования, их анализ и выводы

Для удобства восприятия полученные в ходе тестирования показатели производительности трех интеграционных решений (интеграционных шин) представлены в виде графиков. Предлагаем с ними ознакомиться.

На рисунке 2 приведен график зависимости пропускной способности интеграционных решений (заявок в час) от количества потоков.

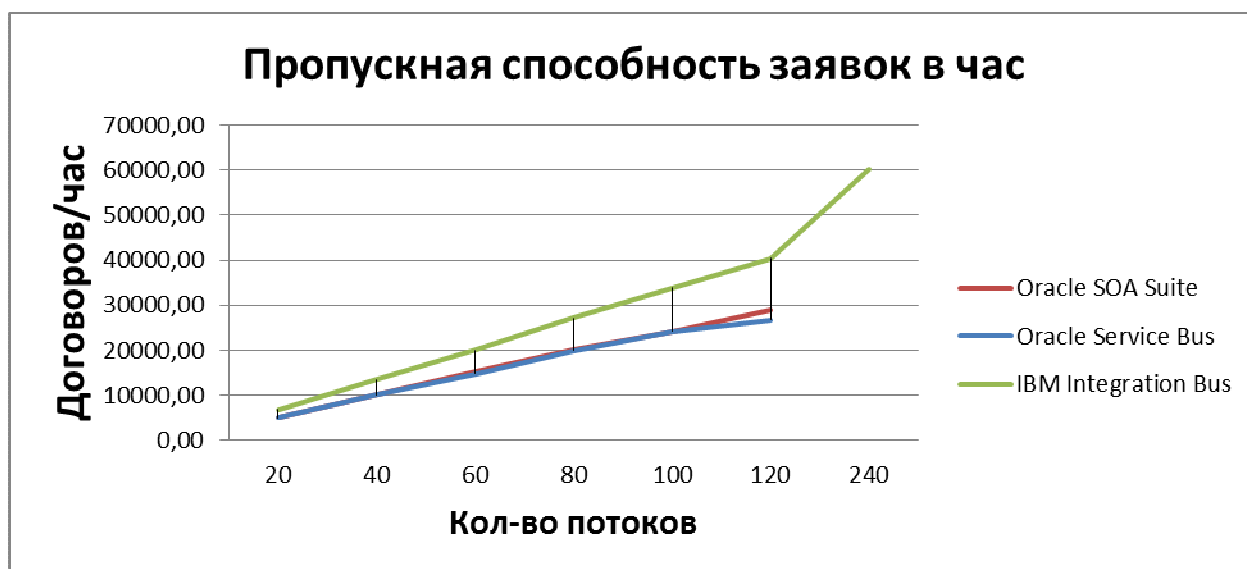


Рис. 2. Зависимость пропускной способности от количества потоков

На рисунках 3, 4, 5 приведены графики зависимости среднего времени работы синхронных сервисов от количества потоков.



Рис. 3. Зависимость среднего времени работы сервиса, отвечающего за получение данных клиента, от количества потоков

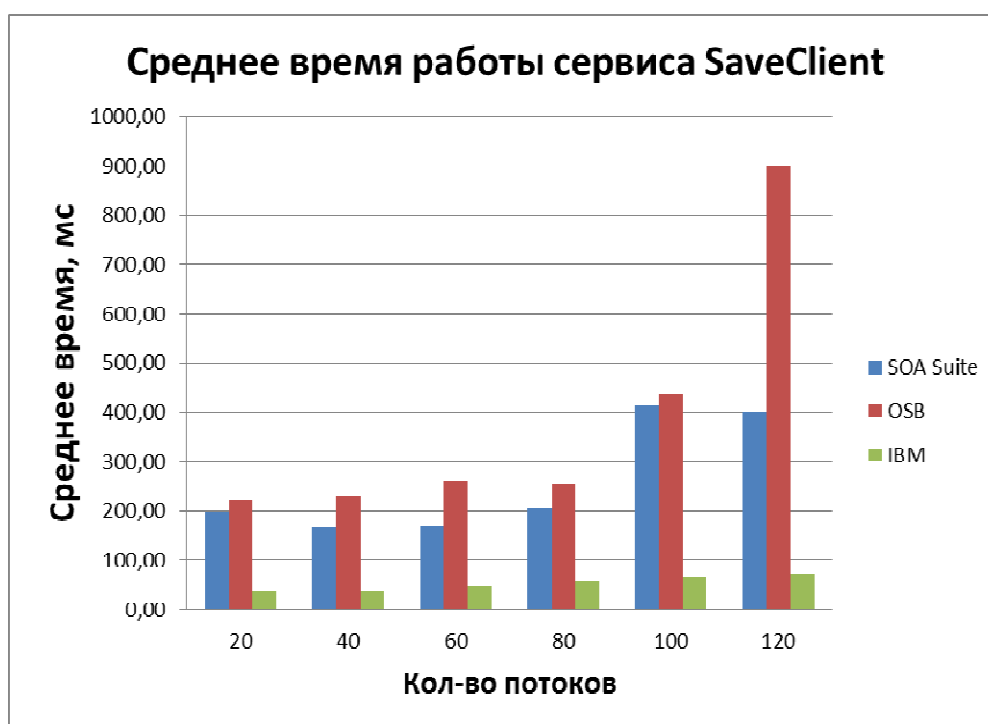


Рис. 4. Зависимость среднего времени работы сервиса, отвечающего за сохранение клиентских данных, от количества потоков

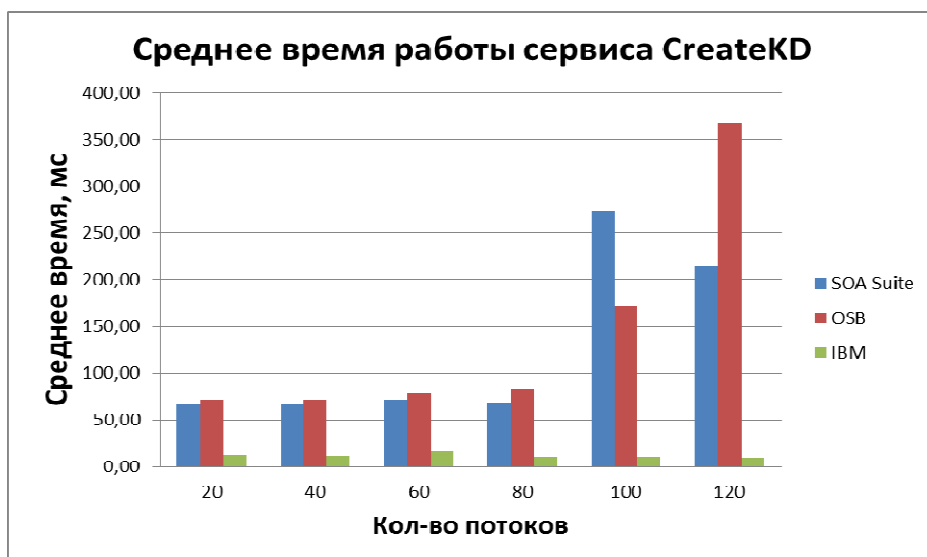


Рис. 5. Зависимость среднего времени работы сервиса, отвечающего за создание кредитного договора, от количества потоков

График зависимости среднего времени работы асинхронных сервисов от количества потоков приведен на рисунке 6.

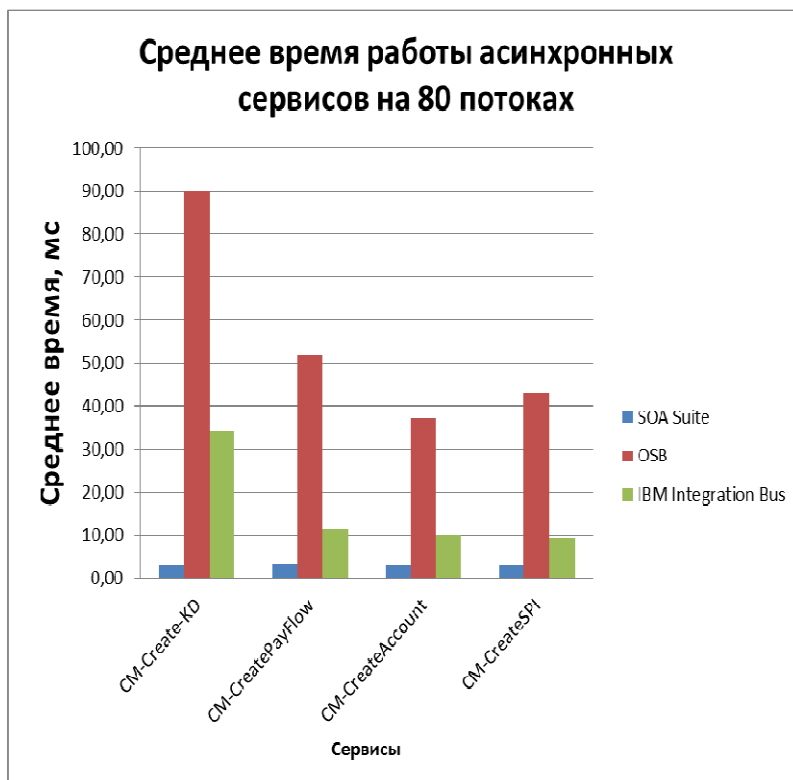


Рис. 6. Зависимость среднего времени работы асинхронных сервисов от количества потоков

На рисунке 7 приведен график зависимости загрузки процессора от количества потоков. *Oracle Service Bus* и *Oracle SOA Suite* продемонстрировали примерно одинаковые показатели как по временам работы сервисов, так и по загрузке процессора. Но график для *Oracle Service Bus* по мере увеличения нагрузки растет значительно быстрее и при 120 потоках практически упирается в доступный максимум. Для *IBM Integration Bus* аналогичные показатели принимают на порядок меньшие значения. Поэтому для него был проведен дополнительный тест на 240 потоков, и он также показал приемлемые результаты — загрузка процессора составила всего 30%. При этом отмечались пики загрузки подсистемы ввода-вывода на 100%, а в генераторе тестовых запросов SoapUI 44 раза была получена ошибка, сообщающая об отсутствии ответа от сервиса, что для общего количества запросов в 40 000 составляет крайне малую долю и, возможно, связано с перегрузкой машины, с которой осуществлялись запуски SoapUI.

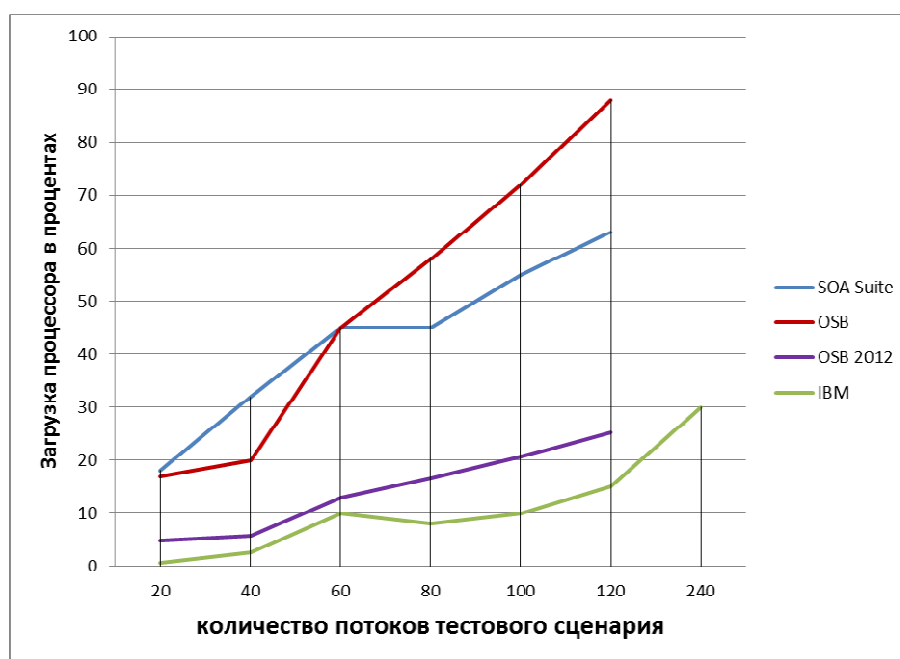


Рис. 7. Зависимость загрузки процессора от количества потоков

Надо заметить, что версия OSB проекта, используемая в сравнительном тестировании, содержала ряд доработок, целью которых было повысить возможности по настройке потоков сообщений. Попытка создания внутри конструктора шины конструктора процессов, который позволил бы без изменения самого интеграционного процесса, а только корректируя его настройки, влиять на поведение процесса (например, менять набор вызываемых адаптеров внешних систем), негативно отразилась на быстродействии процессов. По результатам предыдущего тестирования в Монпелье (на

рисунке 7 они показаны OSB 2012) видно, что эти доработки привнесли неоправданно большую дополнительную нагрузку на шину. Более того, эта идея оказалась утопичной, так как не могла предусмотреть настройку большого количества нюансов, требуемых для реализации бизнес-процессов разных заказчиков. Поэтому, при оценке результатов данного тестирования следует учитывать, что при откате указанных доработок (что было уже сделано в ходе проектов в банках «Глобэкс» и «Земский») показатели OSB по времени выполнения сервисов и загрузке процессора улучшаются в 3–4 раза.

Таким образом, можно заключить, что нагрузочные показатели для интеграционных решений удовлетворяют даже самым высоким требованиям, когда сервер работает в режиме максимальной (пиковой) нагрузки. Поэтому при выборе платформы для интеграции в банке можно сосредоточиться на бизнес-требованиях, ориентироваться на партнерские отношения с интеграторами и вендорами, а также на стратегические цели банка, не думая о том, что решение, реализованное на Oracle SOA Suite, Oracle Service Bus, или IBM Integration Bus может стать «узким местом» в системе.

Список литературы

1. Шилин Г. М. Оценка эффективности интеграции высоконагруженных банковских систем с использованием промышленной интеграционной шины // Молодежный научно-технический вестник. МГТУ им. Н.Э. Баумана. Электрон. журн. 2014. № 3. Режим доступа: <http://sntbul.bmstu.ru/doc/716894.html> (дата обращения 01.12.2014).
2. Белошицкий Д. А. Интеграция данных в информационных системах // Молодежный научно-технический вестник. МГТУ им. Н.Э. Баумана. Электрон. журн. 2013. № 8. Режим доступа: <http://sntbul.bmstu.ru/doc/602635.html> (дата обращения 04.11.2014).
3. Леонов В. С. Методы интеграции корпоративных приложений // Молодежный научно-технический вестник. МГТУ им. Н.Э. Баумана. Электрон. журн. 2013. № 10. Режим доступа: <http://sntbul.bmstu.ru/doc/636907.html> (дата обращения 05.12.2014).