

**Описание решения**

Пример реализации асинхронного метода для системы V2 обеспечивающего смену тарифного плана в биллинговой системе

|  |  |
| --- | --- |
| **Версия** | ***1.0*** |

Листов 5

# Оглавление

[1](#_Toc371643060)

[1 Оглавление 2](#_Toc371643061)

[2 Определения и сокращения 3](#_Toc371643062)

[3 Условия задачи 3](#_Toc371643063)

[4 Краткое описание решения 3](#_Toc371643064)

[5 Установка и запуск интеграционного приложения. 4](#_Toc371643065)

[6 Настройка ресурсов интеграционного приложения. 4](#_Toc371643066)

# Определения и сокращения

|  |  |
| --- | --- |
| **Термин** | **Описание** |
| ESB | Интеграционная шина предприятия |
| GlassFish | Сервер приложений Java для запуска интеграционных приложений в кластере |
| Java Message Service (JMS) | стандарт промежуточного ПО для рассылки сообщений, позволяющий приложениям ESB создавать, посылать, получать и читать сообщения. |
| Адаптер | Модуль или совокупность модулей, реализующих интерфейс к определенной системе |
| Интеграционный слой | Логически законченный набор взаимодействующих модулей, реализующих определенные бизнес-процессы в рамках интеграции систем. |
| Модуль | Единица деплоймента (jar, ear, war) |
| OpenMQ | Приложение, обеспечивающее надёжную доставку сообщений JMS согласно установленных стандартов обмена |

# Условия задачи

***Предоставить пример реализации асинхронного метода для системы V2 обеспечивающего смену тарифного плана в биллинговой системе, для абонента по заданному MSDDN в рамках текущей инфраструктуры ESB.***

*Платформа присылает для смены маркетинговые коды ТП. Существует N региональных инстансов биллинговой системы. Внешняя платформа должна уведомляться обо всех этапах прохождения заявки: регистрация на шине, отправка в биллинговую систему, результат обработки в биллинговой системе. Сервис запросов со стороны биллинговой системы разработан и предоставлен интерфейс. Интерфейс сервиса также разработан и предоставлен. Опишите как можно более подробно предлагаемую Вами реализацию.*

*Форма представления решения:*

*Maven проект на языке Java реализующий задание. Желательная среда разработки NetBeans 7.3*

# Краткое описание решения

Интеграция системы V2 обеспечивается созданием веб-сервиса, принимающим запрос из системы V2, размещающим его в буферной очереди JMS запросов, формирующим и получающим параметры из ESBConfiguration (через клиента веб-сервиса системы ESBConfiguration), необходимые для доступа через Foris.IL интерфейс. Далее полученными параметрами конфигурируется веб-сервис , запрашивающий данные о балансах, производится запрос в Foris.IL и результат направляется в систему V2.

Поскольку имеющаяся документация не содержит данных о точных интерфейсах систем CustomerData3 и ESBConfiguration, задача решалась в предположении о существовании возможности запросить непосредственно ESBConfiguration, а все веб-интерфейсы систем, представляющие результат длительной работы программистов МТС заменены упрощёнными, с целью лишь продемонстрировать идею решения. Вызовы методов веб-сервисов приложения снабжены вызовами средств мониторинга системы для сбора статистики и данных мониторинга.

Отправка извещений системе V2 о прохождении запроса направляется в момент получения сообщения запроса из очереди перед отправкой запроса о данных конфигурации и перед вызовом сервиса смены тарифного плана.

Диаграмма прохождения заявок на смену ТП при запросе параметров подключения через CustomerData.



Средства разработки включали среду разработки NetBeans 7.3.1 и сервера приложений Glassfish 3.1 и 4.0. Сборка проекта осуществлялась посредством Apache Maven 3.0.4 и 3.0.5. Хранилище версий Git, логирование в проекте - slf4j 1.6.1.

# Установка и запуск интеграционного приложения.

Осуществляется посредством стандартных команд asadmin либо через административную консоль сервера приложений Glassfish.

# Настройка ресурсов интеграционного приложения.

Приложение требует создания и настройки ресурсов JMS буферной очереди запросов.

Необходимо создать ресурсы очереди сообщений от V2 к Foris :

фабрику соединений ***V2WFQueueFactory*** c JNDI именем "***jms/V2WFQueueFactory***" и

очередь ***V2WFQueue*** с JNDI именем ***jms/V2WFQueue***.

Параметры указанных ресурсов могут задаваться дополнительно в зависимости от нагрузки.