

# ESB: полное руководство #1.1

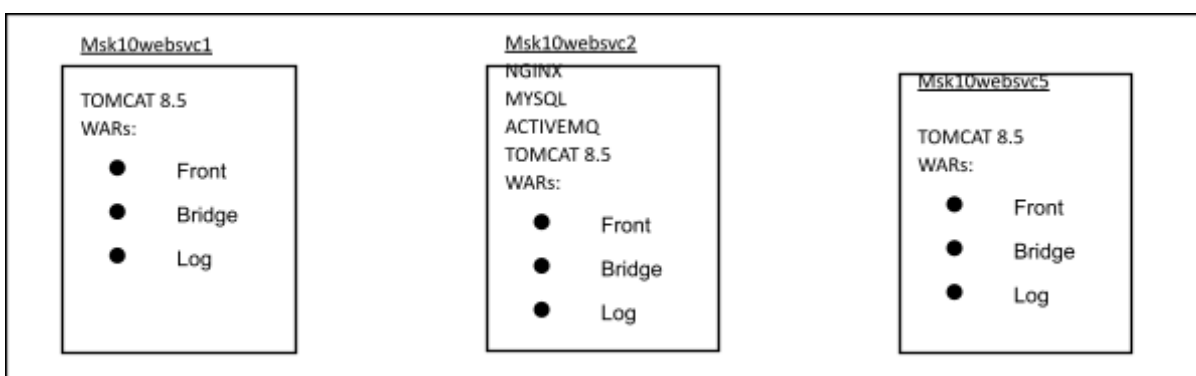
## 1. Основная конфигурация и установка

### Конфигурация

Состоит из 3-х виртуальных серверов (VM) под управлением Windows Server, включает:

- Msk10websvc2 – мастер-нода (ActiveMQ + NGNINX + MYSQL)
- MSK10websvc1 – slave-нода
- MSK10websvc5 - slave-нода

Продукт - стек ПО в разрезе нод (#1.1):



Основные требования при работе с текущей конфигурацией:

1. Допустимо выход из строя slave-ноды шины (msk10websvc [1,5])
2. Критична работа основной ноды (msk10websvc2) из-за наличия в ней ActiveMQ, NGNIX.
3. Для работы любой ноды важно дисковое пространство не менее 10 GB для выгрузки лог-файлов
4. Обязательно необходимо периодически очищать диски от старых лог-файлов

### Установка ESB

Изначальный план развертывания кластера ESB:

- Развернуть одну основную ноду – msk10websvc2 (Если отсутствует основная нода)
- При необходимости развернуть slave-ноду – msk10websvc1 + msk10websvc5 (для масштабирования нагрузки)
- Сконфигурировать доступ к основной ноды в slave-нодах через конфиг-файлы (ТОЛЬКО в случае изменения доменного имени сервера «msk10websvc2» + порт 8888!)

Установка мастер-ноды включает:

- MYSQL (дефолтная установка) + восстановление базы данных ESB
- Настройка прав доступа MYSQL для доменного пользователя jservice
- Установка ActiveMq последней версии (дефолтная установка)
- Tomcat 8.5 + настройка конфиг-файла (дефолтная установка)
- NGNIX + настройка конфиг-файла (дефолтная установка)
- Задеплоить архивные WAR-файлы ESB на Tomcat + настройка конфиг-файлов для доступа к MYSQL + NAV

Конфигурационные файлы хранятся на <путь к шаре>

Подробные шаги установки мастер-ноды (#1.1):

## JDK

- install latest jdk ( from official java «jdk-8u211-windows-x64» or later (<https://www.oracle.com/technetwork/java/javase/downloads/index.html>)
- add windows system environment variables:  
JAVA\_HOME - <path to jdk>  
CATALINA\_HOME - <path to TOMCAT>  
JRE\_HOME - <path to jre>

## SYSTEM DRIVERS

- install vc\_redist.x64

## ADD JSERVICE USER AS ADMIN OF WIN SERVER

- add domain "jservice" in windows local security policy - "LOGON AS A SERVICE"

## MYSQL

- install MYSQL SERVER (Windows Installer) COMMUNITY EDITION (default installation):
  - my sql server
  - connections: odbc-j
  - workbench
 (<https://dev.mysql.com/downloads/installer>)
- recovery JDB database

## TOMCAT

- download TOMCAT from official (>= v. 8.5)
- move to Program Files
- run C:\Program Files\<TOMCAT>\bin\Tomcat<version>w.exe
- add domain user "jservice" on [Log On] Tab
- add memory pool on [Java] Tab
- replace C:\Program Files\<TOMCAT>\conf\tomcat-users.xml with specific file "tomcat-users.xml"
- replace C:\Program Files\<TOMCAT>\conf\server.xml with specific file "server.xml"
- run tomcat
- check from browser - localhost:8899

## NGINX

<http://oss-world.blogspot.com/2015/10/install-nginx-as-windows-service-using.html>

- download and unzip "nssm"
- download unzip nginx (v.1.12.2)

- run CMD
- cd to nssm dir
- run “-nssm.exe install nginx»
- push button INSTALL SERVICE
- replace C:\Program Files\<NGINX> \conf\nginx.conf with spec file “nginx.conf”
- restart nginx service from WIN SERVICES

## ACTIVEMQ (#1.2)

<https://activemq.apache.org/getting-started.html>

<http://activemq.apache.org/version-5-installation>

<https://wiki.eveoh.nl/pages/viewpage.action?pageId=14287030>

- run CMD, cd activemq\bin dir
- run InstallService.bat
- open Windows Service, select ActiveMQ, run it
- check <http://127.0.0.1:8161/admin/> with admin/admin

## DEPLOYING ESB

- add config.properties to C:\Program Files\Java\env\_vars
- open TOMCAT localhost:8899/manager
- deploy WARs files from Manager Dashboard

## Меры в случае сбоя ESB

Если сервер перестал отвечать на запросы:

1. Срабатывает PRTG-монитор – заранее оповещает об недопустимом уменьшении дискового про-ва
2. Отключить Tomcat и очистить диск
3. Снова включить Tomcat

## Среда разработки

Все исходники хранятся на GITHUB <http://github.com/sim-repo>

Для развертывания среды необходим:

- ECLIPSE JEE или Spring Tool Suite
- Java SDK
- Tomcat v.8.0 и старше
- Git Tool

Исходники через GIT Pull-реквесты копируются и открываются студией (см полную официальную документацию работы с GIT и различные tutorиалы: <https://eclipsesource.com/blogs/tutorials/egit-tutorial/>)

## 2. Список интеграционных потоков

Потоки делятся по принципу доступа синхронный/асинхронный.

При синхронных клиент находится в состоянии ожидания получения запрашиваемых данных (GET-requests). При асинхронных клиенту приходят данные, когда произойдет определенное событие и клиент не знает точно, когда это событие наступит.

### 1.1. Список асинхронных потоков по схеме «публикации-подписки» (pub-sub)

N	Событие ID	Описание	Отправитель ID	Получатель ID	Частота
1	CHANGE_SORDER	Синхронизация изменения SO	NAV	BTX	Часто за день
2	CRM_SORDER	Создание SO	NAV	CRM	Не активен
3	NAV_FULL_CUST	Создание карточки клиента	NAV	ONE	Не активен
4	NAV_FULL_VEND	Создание карточки поставщика	NAV	ONE	Не активен
5	NAV_VEN_Addr	Заведение адреса поставщика	NAV	ONE	Раз в месяц

6	NAV_CUST_BANK	Изменение КлиентБанкСчет	NAV	ONE	Не активен
7	NAV_CUST	Изменение Клиента	NAV	ONE	Не активен
8	NAV_VEN_BANK	Изменение ПоставщикБанкСчет	NAV	ONE	Не активен
9	NAV_CUST_Addr	Заведение адреса клиента	NAV	ONE	Не активен
10	NAV_AGREEMENT	Заведение договора	NAV	ONE	Не активен
11	DOC_TRANSFERSHIPHEADER	Перемещение Расх. Накладная	NAV	ONE	Часто за день
12	DOC_TRANSFERRECEIPTHEADER	Перемещение Прих. Накладная	NAV	ONE	Часто за день
13	DOC_SALESINVOICEHEADER	Счета-Фактуры Продажи	NAV	ONE	Часто за день
14	DOC_SALESCRMEMOHEADER	Клиент Кредит-Ноты	NAV	ONE	Часто за день
15	DOC_RETURNSHIPHEADER	Клиент Возвраты	NAV	ONE	Редко за день
16	DOC_PURCHINVHEADER	Счета-Фактуры Покупки	NAV	ONE	Часто за день
17	DOC_PURCHCRMEMOHEADER	Поставщик Кредит-Ноты	NAV	ONE	Редко за день
18	DOC_ITEMSHIPHEADER	Товарная Накладная	NAV	ONE	Редко за день
19	DOC_ITEMRECEIPTHEADER	Акт Оприходования	NAV	ONE	Редко за день
20	DOC_CASHCHECKEHEADER	Касса Чек	NAV	ONE	Часто за день
21	EKB_CLIENT_NAV	Заведение Клиента	NAV	CRM	Редко за день
22	EKB_CLIENT_BITRIX	Заведение Клиента	BTX	CRM	Средне за день
23	EKB_CLIENT_1C	Заведение Клиента	ONE	CRM	Не активен
24	EKB_CLIENT_BPM	Заведение Клиента	CRM	CRM	Не активен
25	LK_SORDER	Изменение Заказа	NAV	BTX	Редко за день
26	LK_CHANGE_ACTIVE_LOGIN	Активация Логина	CRM	BTX	Редко за день
27	LK_CHANGE_CUST	Изменение Клиента	CRM	BTX	Редко за день
28	CREATE_SORDER	Создание Заказа	BTX	NAV	Редко за день
29	ONE_NAV_DDS	ДДС	ONE	NAV	Не активен
30	BTX_NAV_USER_CREATE	Создание Клиента	BTX	NAV	Не активен

## 1.2. ПОЛУЧЕНИЕ ДАННЫХ ПО ТРЕБОВАНИЮ (СИНХРОННЫЕ ПОТОКИ «ON-DEMAND»)

N	Источ ник	Описание	URL шины	Метод источника
1	Navision	Группы скидок по сотрудникам	http://msk10websvc2:8888/jservice/empl/list	[dbo].[web_getDiscGroupsByEmpl]
2		Статусы по заказам продажи	http://msk10websvc2:8888/jservice/so/state/{sorderId}	[dbo].[web_GetStateSorderQueue]
3		Клиентские операции (CLE)	http://msk10websvc2:8888/jservice/cust/operation <u>параметры:</u> <ul style="list-style-type: none"> <li>• custId - обязателен</li> <li>• shipmentType IN (0,1,2) - обязателен</li> <li>• companyName - НЕ обязателен, по умолчанию 'СИМПЛ'</li> </ul> Пример: http://msk10websvc2:8888/jservice/cust/operation?custId=X3710&shipmentType=0	[dbo].[web_getCustOperations]
4		История клиентских продаж (CLE)	http://msk10websvc2:8888/jservice/cust/history/{custId} custId - код клиента	[dbo].[web_GetSalesHistory]
5		Статусы по заказам-продажи (лайт)	http://msk10websvc2:8888/jservice/so/get_state?btxSorderNo=БРК0000000000000198	

6		Все товарные остатки по всем компаниям	http://msk10websvc2:8888/jservice/items/rem	[dbo].[web_getAllInvQty]
7		Товарные остатки	<a href="http://msk10websvc2:8888/jservice/item/available?itemNo=99896">http://msk10websvc2:8888/jservice/item/available?itemNo=99896</a> <b>параметры:</b> <ul style="list-style-type: none"> <li>itemNo - обязателен (код товара NAV)</li> <li>shipmentType - не обязателен (тип отгрузки), по умолчанию: "2"</li> <li>locationCode - не обязателен (код склада), по умолчанию: "РЦ1"</li> <li>companyName - не обязателен (название компании) по умолчанию: "СИМПЛ"</li> </ul> <b>Пример:</b> <a href="http://msk10websvc2:8888/jservice/item/available?itemNo=99896&amp;shipmentType=1">http://msk10websvc2:8888/jservice/item/available?itemNo=99896&amp;shipmentType=1</a>	[dbo].[f_GetAvailInvQty]
8		Список менеджеров	http://msk10websvc2:8888/jservice/cust/managers	[dbo].[web_getSalespersons]
9		Ограничения выписки	<a href="http://msk10websvc2:8888/jservice/item/restrict/company?companyName=&lt;...&gt;&amp;itemNo=&lt;...&gt;&amp;cfo=&lt;...&gt;&amp;custId=&lt;...&gt;&amp;userId=&lt;...&gt;&amp;orderPostingDate=&lt;...&gt;">http://msk10websvc2:8888/jservice/item/restrict/company?companyName=&lt;...&gt;&amp;itemNo=&lt;...&gt;&amp;cfo=&lt;...&gt;&amp;custId=&lt;...&gt;&amp;userId=&lt;...&gt;&amp;orderPostingDate=&lt;...&gt;</a>	web_IsItemForbiddenForCompany
10		Список клиентов	http://msk10websvc2:8888/jservice/cust	[dbo].[web_GetCustomers]
11		Менеджера по клиенту	http://msk10websvc2:8888/jservice/cust/manager/X3540	[dbo].[web_GetSalesperson]
12		Клиентские адреса доставки	http://msk10websvc2:8888/jservice/cust/ship_addr/X3540	[dbo].[web_getCustomerPrice]
13		<b>Хенд-Бук</b>		[dbo].[web_getHandBook]
		Деканация	http://msk10websvc2:8888/jservice/handbook/decantation	[dbo].[web_getHandBook] @_type = 17
		Производство	http://msk10websvc2:8888/jservice/handbook/manufacture	[dbo].[web_getHandBook] @_type = 21
		Категория	http://msk10websvc2:8888/jservice/handbook/category	[dbo].[web_getHandBook] @_type = 27
		Тип вина	http://msk10websvc2:8888/jservice/handbook/type_wine	[dbo].[web_getHandBook] @_type = 14
		Тип упаковки	http://msk10websvc2:8888/jservice/handbook/pack_type	EXEC [dbo].[web_getHandBook] @_type = 1
		Оттенок	http://msk10websvc2:8888/jservice/handbook/raw	EXEC [dbo].[web_getHandBook] @_type = 24
		Выдержка способ	http://msk10websvc2:8888/jservice/handbook/aging	EXEC [dbo].[web_getHandBook] @_type = 22
			http://msk10websvc2:8888/jservice/handbook/item_char_rating	EXEC [dbo].[web_getHandBook] @_type = 51
		Материалы	http://msk10websvc2:8888/jservice/handbook/material	EXEC [dbo].[web_getHandBook] @_type = 32
		Регионы	http://msk10websvc2:8888/jservice/handbook/region	EXEC [dbo].[web_getHandBook] @_type = 20
		Рекомендации	http://msk10websvc2:8888/jservice/handbook/recommend_drink	EXEC [dbo].[web_getHandBook] @_type = 33
		Рейтинги	http://msk10websvc2:8888/jservice/handbook/rate_agency	EXEC [dbo].[web_getHandBook] @_type = 18
		Способ культивации	http://msk10websvc2:8888/jservice/handbook/cultivation_type	EXEC [dbo].[web_getHandBook] @_type = 34
		Сахар	http://msk10websvc2:8888/jservice/handbook/sugar_type	EXEC [dbo].[web_getHandBook] @_type = 19
		Страна	http://msk10websvc2:8888/jservice/handbook/country	EXEC [dbo].[web_getHandBook] @_type = 50
		Категория напитка	http://msk10websvc2:8888/jservice/handbook/drink_category	EXEC [dbo].[web_getHandBook] @_type = 26
		Аксессуары для стекла	http://msk10websvc2:8888/jservice/handbook/accessory_class	EXEC [dbo].[web_getHandBook] @_type = 37
		Стиль тип	http://msk10websvc2:8888/jservice/handbook/style_type	EXEC [dbo].[web_getHandBook] @_type = 16
		Метод производства	http://msk10websvc2:8888/jservice/handbook/production_method	EXEC [dbo].[web_getHandBook] @_type = 31

		Цвета	http://msk10websvc2:8888/jservice/handbook/color	EXEC [dbo].[web_getHandBook ] @_type = 0
		Серийность	http://msk10websvc2:8888/jservice/handbook/series	EXEC web_getHandBook @_type = 30
		Класс	http://msk10websvc2:8888/jservice/handbook/class	EXEC [dbo].[web_getHandBook ] @_type = 23
		Сорт винограда	http://msk10websvc2:8888/jservice/handbook/grape_kind	EXEC [dbo].[web_getHandBook ] @_type = 12
		Винтаж	http://msk10websvc2:8888/jservice/handbook/rate_vintage	EXEC [dbo].[web_getHandBook ] @_type = 13
		Общие типы	http://msk10websvc2:8888/jservice/handbook/wine_type	EXEC [dbo].[web_getHandBook ] @_type = 2
		Стиль	http://msk10websvc2:8888/jservice/handbook/style	EXEC [dbo].[web_getHandBook ] @_type = 25
		Коды сортов винограда	http://msk10websvc2:8888/jservice/handbook/item_char_grape	EXEC [dbo].[web_getHandBook ] @_type = 52
		Апелласьон	http://msk10websvc2:8888/jservice/handbook/appelason	EXEC [dbo].[web_getHandBook ] @_type = 15
14		ДхВх Отправка инвойса	POST-запрос: http://msk10websvc2:8888/jservice/dxbx/invoice2/{invoiceNo}	[EXEC [dbo].[web_dxbx_postInvoice]
15		ДхВх Проверка соединения	http://msk10websvc2:8888/jservice/dxbx/connection	[dbo].[web_dxbx_getConnectionBody ]
16		ДхВх Проверка юридических лиц	http://msk10websvc2:8888/jservice/dxbx/inn/{customerNo}	[dbo].[web_dxbx_getInnList]

## 3. Архитектура

### 2.1 *Концепция технической реализации*

В условиях ограниченного бюджета основополагающим принципом при построении лежит абстрагирование от реализации каких-либо бизнес-правил внутри самого ПО.

Это сделано по следующим соображениям:

- исключение необходимости внесения изменений вслед за изменениями в учетных системах и передаваемых данных – шины данных должна быть в полной мере адаптирована к динамике изменения бизнес-процессов
- шаблонная интеграция – ускорение проекта, уход от «подводных камней», уменьшение возможных багов за счет обкатанных решений

## 2.2 Минимальные требования к работе шине и основные функции

На момент написания документации (#1.00) система выполняет следующие функции:

- выдача данных по требованию (on-demand) GET-запросы в sync/async режимах
- прямая рассылка сообщений между подписчиками (pub-sub)
- конвертация сообщений в форматах JSON/XML

Для работы шины нужно обеспечить следующий минимум:

- сервисы front-, bridge-, log- должны быть доступны
- настроенные и работающие зависимые 3rd-party компоненты

## 2.3 Архитектура приложения

В основе лежит подход микросервисов (MSA). Построено на REST-архитектуре.

Используются собственные и внешние программные решения в качестве базы для платформы. Сервисы спроектированы по принципу ролей, общаются через общие интерфейсы взаимодействия (контракты).

С т.з. взаимозависимости программные компоненты делятся на:

### *Зависимые компоненты*

- front-service – фронт, принимает входящие запросы, обрабатывает простые
- bridge-service – сервис, отвечающий за маршрутизацию сообщения в endpoint-ы
- log-service – логирующий сервис в БД MySQL

### *Независимые компоненты*

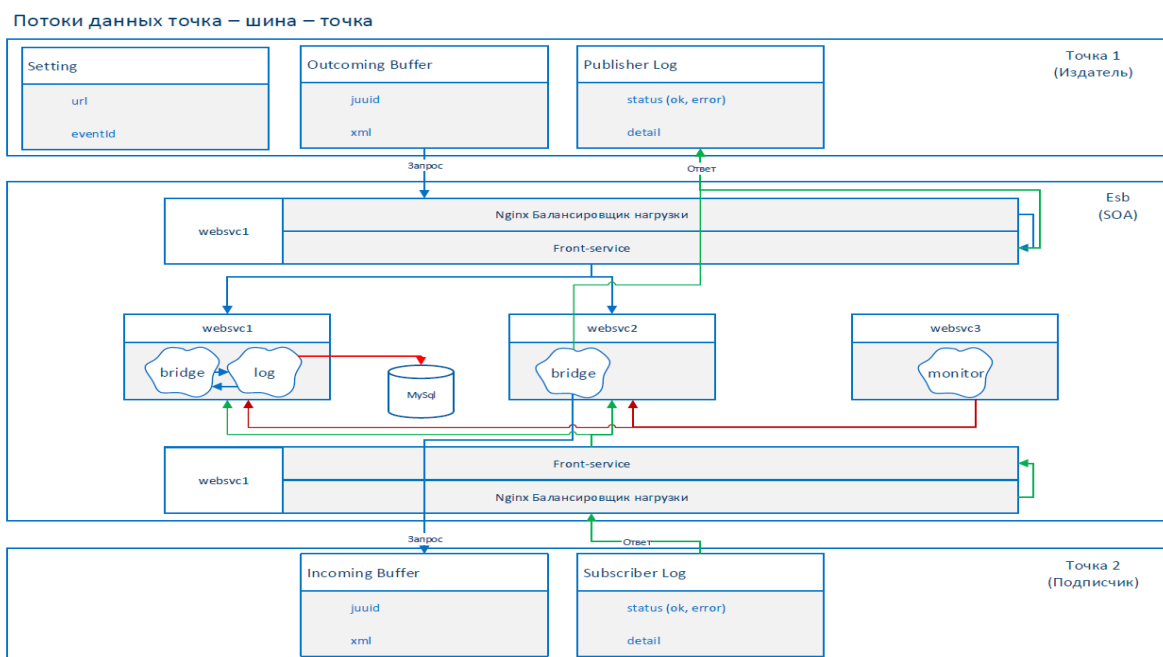
- mon-service – health-state monitor, включающий планировщик заданий

### *3<sup>rd</sup>-party базовые компоненты*

- СУБД MySQL (база-данных JDB)
- Apache ActiveMQ
- Spring (DI, Integration)
- Hibernate ORM
- Maven
- NGNIX

Общая схема показывает одну из возможных конфигураций, когда бэкенды разнесены по нескольким серверам:





Общая схема архитектуры (#1.00)

### 2.3.1 Горизонтальная масштабируемость

Предусмотрена горизонтальная масштабирование: на уровне Нод (Nodes), а также разнесение сервисов по разным физическим машинам. В качестве балансировщика нагрузки используется NGINX.

База для обеспечения функций масштабирования:

- проектирование сервисов на принципах ролей
- слабая связанность между сервисами через очереди/топики
- введение общих интерфейсов взаимодействия (контракты)

### 2.3.2 Управление производительностью

Управления производительностью на момент написания документации (#1.00) происходит в ручном режиме, пул потоков определяется в момент инициализации (старта) сервисов.

Текущая база:

- многопоточность на уровне задач
- batch-insert

Планируется:

- реализация функций фидбека от сервисов к балансировщику «обратный шторм»
- серверное кэширование

- отслеживание статистики

### 2.3.3 Управление отказоустойчивостью

Есть определенная зависимость между ролями, поэтому общая логика определяет, что для обеспечения синхронных запросов все роли должны функционировать. Требования в случаях асинхронных запросов предусматривает случайное аварийное отключение какого-либо из ролей, т.к. все запросы сохраняются в персистентных очередях.

База:

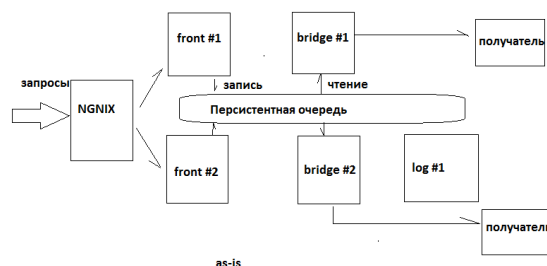
- балансировщик (Proxy-server)
- возможности масштабирования
- персистентные очереди

Планируется:

- функции оповещения
- health-state мониторинг

Для персистентных очередей используется ActiveMQ.

В текущем релизе первое обращение к ActiveMQ происходит после того, как фронт-сервис принял сообщение в обработку:



### 2.3.4 Используемые каналы сообщений

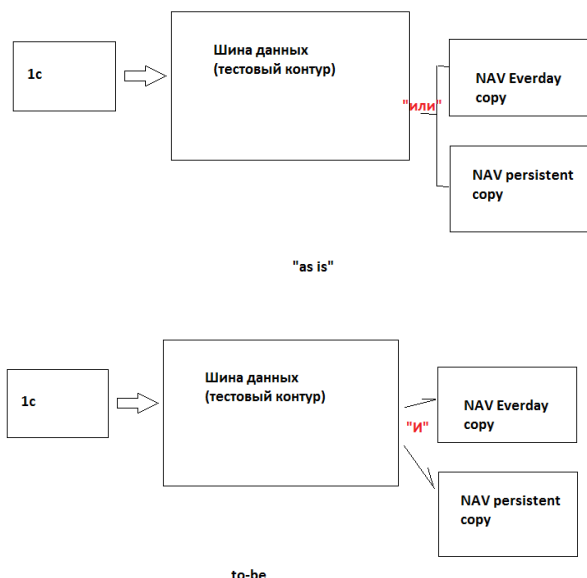
Механизм взаимодействия между сервисами использует Producer-Consumer Design Pattern и включает:

- персистентные очереди посредством ActiveMQ
- экспортирование компонентов в виде удаленных служб JMS RPC (JmsInvokerServiceExporter, JmsInvokerProxyFactoryBean )
- внутренние потокозащищенные очереди

### 2.3.5 Системы хранения

Все данные хранятся в БД «JDB» (MySQL Community Edition). Данные делятся на настроечные данные и логи. Для чтения/записи используется ORM Hibernate с конфигурацией, основанной на XML-маппинге.

В будущем релизе планируется ввести поддержку экземпляров учетной системы, а не тип учетной системы как сейчас, таким образом на одном тестовом контуре шины возможно будет подключать разные тестовые базы данных, например NAVISION: ежедневная копия и персистентная:

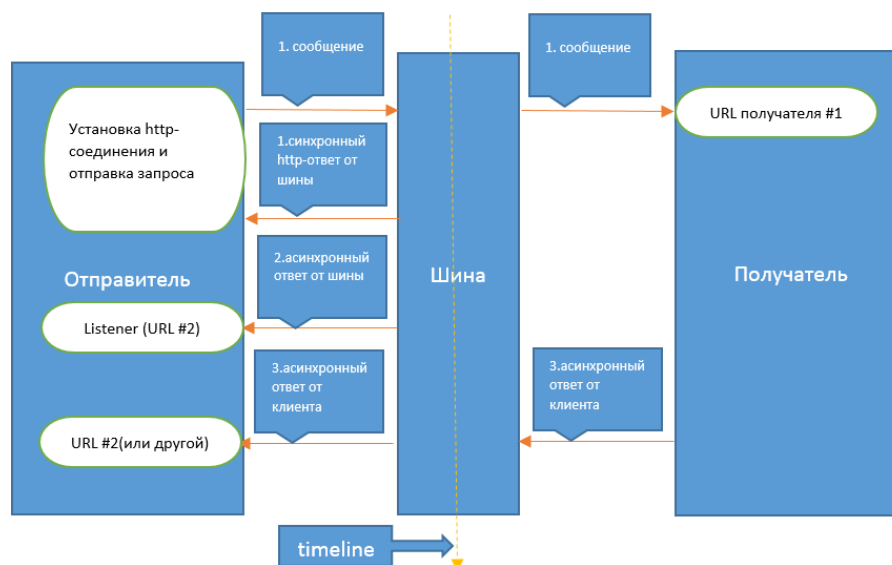


## 4. Форматы обменов

В этом разделе описывается набор правил определения *интерфейсов* компонентов и их реализаций, а также правил, по которым компоненты работают в системе и взаимодействуют друг с другом.

Общие фазы взаимодействия между отправителем и получателем при рассылках следующие:

- установка http-соединения; отправка клиентом запроса; получение **отправителем** ответа от шины, что сообщение принято в обработку
- обработка запроса шиной и доставка **получателю**
- получение ответа **отправителю** от шины, что сообщение было успешно доставлено
- отправка ответа от получателя к **отправителю**, что сообщение было обработано



### 2.3.6 Установка http-соединения и отправдение запроса

Любые сообщение – отправляются в едином формате (структура полей).

В текущем релизе установка http-соединения не требует авторизации и электронно-цифровой подписи (ЭЛТ).

При отправке сообщения приходит стандартный HTTP-ответ «200», что сообщение было только принято в обработку шины.

*Синхронный ответ «200» от шины не означает, что сообщение будет гарантировано доставлено до адресата.*

*Это обусловлено тем, что может произойти ситуация, когда шине не удастся установить соединение с URL-получателя (http «404»).*

#### JSON – отправка

URL шины для отправки сообщения:

<http://msk10websvc3:8888/front/async/json/pub/uni> (ТЕСТ)

<http://msk10websvc2:8888/front/async/json/pub/uni> (РАБОЧАЯ)

Формат сообщения (#1001, v.1.0):

```

{
  "senderId": "<BTX|CRM|ONE|NAV>", -- зависит от системы, кто отправитель
  "juuid": "AE77D02F-E7D7-4176-88E2-F9ACC4C600D1", --уникальная метка сообщения, генерируется отправителем
  "eventId": "<в кавычках id-event>", --событие, которое определяет сущность сообщения
  "body": "< любой текст в JSON,XML etc.. (кавычки/спец-символы экранируются обратным слэшем > ",
  "contentType": "< ApplicationJson|ApplicationXml|PlainText|Other> " -- нужен для получателя, чтобы понимать какой формат сообщения скрытан в теге «body»(опционально, можно не использовать)
}

```

#### XML-отправка

URL шины для отправки сообщения:

<http://msk10websvc3:8888/front/async/xml/pub/uni> (ТЕСТ)

<http://msk10websvc2:8888/front/async/xml/pub/uni> (РАБОЧАЯ)

Формат сообщения (#1002, v.1.0):

```
<Message>
<clazz> com.simple.server.domain.contract.UniMsg</clazz>
<eventId>ONE_NAV_DDS</eventId>
<juuid>4765B0DF-A5B0-4865-8691-BF09787B9CB4</juuid>
<senderId>ONE</senderId>
<body>
  <![CDATA[
    <sorders>
      <sorder>
        <sorderNo>SL00117058205</sorderNo>
        <outerSorderId />
        <sorderStatus>delivered</sorderStatus>
        <itemOuterLineId>98024</itemOuterLineId>
        <itemLineId>10000</itemLineId>
        <itemUnitPrice>6990.0000</itemUnitPrice>
        <itemQuantity>1</itemQuantity>
      </sorder>
    </sorders>
  ]>
</body>
</Message>
```

обработка запроса шиной и доставка **получателю**

### 2.3.7 Обработка запроса шиной и доставка получателю

**Конвертированное JSON-сообщение для получателя**

Формат (#1101, v.1.0):

```
{
  "juuid": "..", -- уникальная метка сообщения
  "eventId": "..", -- тип события
  "body": ".." -- полезная нагрузка payload
  "contentType": "ApplicationJson|JsonPlainText|ApplicationXml|XmlPlainText|Other", --формат в теле body
  "url": "..", -- это поле будет удалено!
  "clazz": "com.simple.server.domain.contract.UniMinMsg" -- класс для понимания что за сообщение (это поле
  будет удалено!)
}
```

В конечном виде к получателю приходит сообщение

*пример 1 (клиенту нужен JSON в теле body):*

```
{
  "juuid": "77777", -- уникальная метка сообщения
  "eventId": "MY_TEST", -- тип события
  "body": "{\"navClientId\":\"Vasya\"}", -- полезная нагрузка payload
  "contentType": "ApplicationJson", --формат в теле body
  "url": "http://msk10websvc3:8180/front/async/json/test", -- это поле будет удалено!
  "clazz": "com.simple.server.domain.contract.UniMinMsg" -- класс для понимания что за сообщение (это поле
  будет удалено!)
}
```

*пример 2 (клиенту нужен XML в теле body):*

```

{
  "juuid": "77777",
  "eventId": "MY_TEST",
  "body": "<sorders><sorder><sorderNo>SL00117058205</sorderNo><outerSorderId /><sorderStatus>delivered</sorderStatus><itemOuterLinId>98024</itemOuterLinId><itemLinId>10000</itemLinId><itemUnitPrice>6990.0000</itemUnitPrice><itemQuantity>1</itemQuantity></sorder></sorders>",
  "contentType": "ApplicationJson",
  "url": "http://msk10websvc3:8180/front/async/json/test",
  "clazz": "com.simple.server.domain.contract.UniMinMsg"
}

```

*пример 3 (клиенту нужен закодированный бинарник в теле body):*

```

{
  "juuid": "77777",
  "eventId": "MY_TEST",
  "body": "1110100 1100101 1110011 1110100",
  "contentType": "Other",
  "url": "http://msk10websvc3:8180/front/async/json/test",
  "clazz": "com.simple.server.domain.contract.UniMinMsg"
}

```

### Конвертированное XML-сообщение для получателя

Формат (#1102, v.1.0):

```

<UniMinMsg>
  <clazz>com.simple.server.domain.contract.UniMinMsg</clazz>
  <juuid>..  
  <eventId>..  
  <body>..  
  <url>..  
  <datetime>/>
  <contentType> ApplicationJson | ApplicationXml | PlainText | Other </contentType>
</UniMinMsg>

```

Конечный вид сообщения для получателя:

*Пример 1 (все скобки « < » заменены на « &lt; » ):*

```

<UniMinMsg>
  <clazz>com.simple.server.domain.contract.UniMinMsg</clazz>
  <juuid>4765B0DF-A5B0-4865-8691-BF09787B9CB4</juuid>
  <eventId>ONE_NAV_DDS</eventId>
  <body>
    &lt;sorders>
      &lt;sorder>
        &lt;sorderNo>SL00117058205&lt;/sorderNo>
        &lt;outerSorderId />
        &lt;sorderStatus>delivered&lt;/sorderStatus>
        &lt;itemOuterLinId>98024&lt;/itemOuterLinId>
        &lt;itemLinId>10000&lt;/itemLinId>
        &lt;itemUnitPrice>6990.0000&lt;/itemUnitPrice>
      &lt;/sorder>
    &lt;/sorders>
  </body>
  <url>http://msk10websvc3/front/async/json/pub/uni</url>
  <datetime>/>
  <contentType>ApplicationXml</contentType>
</UniMinMsg>

```

*пример 2:*

```

<UniMinMsg>

```

```

<clazz>com.simple.server.domain.contract.UniMinMsg</clazz>
<juid>4765B0DF-A5B0-4865-8691-BF09787B9CB4</juid>
<eventId>ONE_NAV_DDS</eventId>
<body>
    Любой текст, в т.ч. JSON-сообщение
</body>
<url>http://msk10websvc3/front/async/json/pub/uni</url>
<datetime/>
<contentType>ApplicationXml</contentType>
</UniMinMsg>

```

### 2.3.8 Асинхронный ответ от шины

Настраивается листенер (прослушка) у отправителя для получения информации, что сообщение было успешно отправлено от шины к получателю. Если возникла ошибка на этапе отправки, то будет возвращено описание на этот же URL (опционально можно настроить другой URL).

Пример успешный отправки из CRM в ВТХ:

```

{
// эти поля в планах убрать из сообщения:
"clazz": "com.simple.server.domain.contract.SuccessPubMsg",
"publisherId": "CRM",
"endPointId": "<NAV| ВТХ..>",
"operationType": " PUB ",
"responseURL": "http://msk10bpm1.simple.ru:888/Service.svc/ProcessMessage",
"responseContentType": "ApplicationJson",

// поля, которые останутся:
"juid": "32D9C6AE-1E0E-41FB-A91D-2F89172DB3CD",
"eventId": "CRM_BT_X_CHANGE_CUST",
"senderId": "CRM",
"subscriberId": "ВТХ",
"logDatetime": "11.09.2017 15:32:59",
"errorId": "NULL", -- ошибок нет
"subscriberHandler": "http://msk10bpm1.simple.ru:888/Service.svc/ProcessMessage", --url подписчика, прописанный в настройках шины
}

```

*В редких случаях будет возвращено поле «subscriberStoreClass» – указывается, когда у подписчика нет технических возможностей создавать listener-ы (происходит через прямую запись в БД, например в случае с NAVISION):*

```
"storeClass": "Incoming Buffer"
```

В случае ошибок при отправке от шины к получателю в отбивке будет добавлено поле details:

```
"errorId": "PubTask", -- ошибка, возникла на этапе рассылки (publish task).
```

```
"details": "HttpImpl(UniMinMsg) url: < http://msk10bpm1.simple.ru:888/Service.svc/ProcessMessage234 >, content-type: < ApplicationJson >, < 404 Not Found >"
```

Есть возможность получения ответов от шины на 2-а разных URL, например успешные приходят на URL #1, ошибки - #2

### 2.3.9 Асинхронный ответ от получателя

Для получателя все подтверждения должны отправляться на адреса шины:

- Тестовый контур:  
<http://msk10websvc3:8888/front/async/json/sub/confirm>
- Рабочий контур:  
<http://msk10websvc2:8888/front/async/json/sub/confirm>

#### Формат сообщения:

```
{
  "responseContentType": "ApplicationJson",
  "clazz": "StatusMsg",
  "juuid": "<guid-метка сообщения>",
  "senderId": "<код системы отправителя сообщения>",
  "eventId": "CONFIRM",
  "code": "<0,1,2,3>",
  "message": "<заполняется в случае Code= {0,1}; краткое сообщение, например, 'committed'>",
  "errorId": "<внутренняя или общая в рамках компании классификация ошибок (Code={2,3})>",
  "details": "<детализация ошибки>"
}
```

Поле "Code" – указывается всегда, в случае успеха/ошибки. Тип: text.	
Значения	Комментарии
0	Сообщение в очереди/не обработано (wait)
1	Успешно обработано, валидация прошла, создана запись (committed)
2	Внешняя ошибка – не прошло валидацию, ошибка в значениях полей (ext.err)
3	Внутренняя ошибка – внутренние причины: ошибка программной логики (inner err)

Поле "SenderId" – указывается всегда, обозначает код системы отправителя	
Значения	Комментарии
NAV	Navision
BTX	Bitrix
CRM	Terrasoft
ONE	1C

Остальные поля	
Message	Размерность не ограничена
ResponseContentType	В планах убрать из следующих реализаций
ErrorId	Тип: text. Коды внутренний или общей классификации ошибок.
Details	Размерность не ограничена

#### Примеры получения ответов от ВТХ:

##### Ошибок нет:

##### 1 вариант:

```
{
  "responseContentType": "ApplicationJson",
  "clazz": "StatusMsg",
  "juuid": "EC1FBF91-AB9A-46AD-9466-24F6DB41FA22",
  "senderId": "BTX",
  "eventId": "<id-события для ответа>", -- пример: BTX_CRM_CONFIRM_CUST
  "code": "1",
  "message": "<любое текстовое сообщение>",
}
```



```
"errorId": "",
"details": ""
}
```

2 вариант:

```
{
"responseContentType": "ApplicationJson",
"clazz": "StatusMsg",
"juuid": "EC1FBF91-AB9A-46AD-9466-24F6DB41FA22",
"senderId": "BTX",
"eventId": " BTX_CRM_CONFIRM_CUST ",
"code": "1",
"message": ""
}
```

#### Ошибки:

```
{
"responseContentType": "ApplicationJson",
"clazz": "StatusMsg",
"juuid": "EC1FBF91-AB9A-46AD-9466-24F6DB41FA22",
"senderId": "BTX",
"eventId": " BTX_CRM_CONFIRM_CUST ",
"code": "2",
"errorId": "any text",
"details": "any text"
}
```

### 2.3.10 Требования к содержимому body

- Для отправки JSON-содержимого в теге body, требуется экранировать через обратный слэш двойные кавычки внутри значения:

```
{
..
  "body": " {\"text \" text} "
..
}
```

- XML-отправка должна внутри тега выглядеть так:  
`"body"><![CDATA[<root><TypeofPayment>Поставщик</TypeofPayment><OriginalAmount>1000525.25</OriginalAmount><PaymentCurrency>RUR</PaymentCurrency><Contractor>П56770</Contractor><DDSCodeNormal>044.01.0200</DDSCodeNormal><GlobalDimension2Code>УП-01-100</GlobalDimension2Code><PaymentType>Безналичная</PaymentType><PayBeforeDate>20171010</PayBeforeDate><Description>Зарплата сентябрь 2017</Description></root>]]></body></Message>`

Если необходимо передавать в теге body любой формат, отличный от общепринятых XML, JSON, например бинарный код или plain-text, то достаточно позаботиться об экранировании двойных кавычек внутри значения, если они используются.

Это обеспечит валидность JSON-формата.

### 2.3.11 Конвертация данных payload

Функция конвертации данных нужна только в случае, если получателю нужен формат JSON или XML.

Например, отправитель высылает данные в теге body как XML, что в итоге конвертируется в JSON и передается получателю.

Во всех остальных случаях, функция конвертации не задействована.

- Стандартное экранирование  
Для отправки любого содержимого в теге body, требуется экранировать через обратный слэш двойные кавычки внутри значения:
- Специальное экранирование

Для некоторых отправителей (например, 1С) нет технической возможности формировать payload-сообщения в теге «BODY» как экранированный текст через обратный слэш.

В этом случае, можно использовать любой одиночный символ или комбинацию символов до 3-х знаков:

*“body”: “{ ‘поле’: ‘значение’ }”*

*“body”: “{ \$поле\$: \$значение\$ }”*

*“body”: “{ #\$\$поле#\$\$: #\$\$значение#\$\$ }”*

### 2.3.12 Ретрансляция http-ответов

Необходимо для получения данных из удаленного источника через шину.

Инициатор делает запрос с определенными параметрами к шине, которая в свою очередь переадресует запрос к источнику, принимает ответ и возвращает его в исходном виде инициатору запроса.

Настройки загружаются в кэш шины при ее запуске и хранятся там до следующего рестарта.

Для онлайн проверки кэша используются 2-а утилитных метода с следующими адресами:

<http://msk10websvc2:8888/front/util/cache/check/retranslate?key=<ключ>> - по ключу

<http://msk10websvc2:8888/front/util/cache/allRetranslates> - весь список

пример:

<http://msk10websvc2:8888/front/util/cache/check/retranslate?key=/sync/get/json/bpm/clientRecommendations>

### 2.3.13 Дополнительные рекомендации

Ограничивайте скорость отправок в случае однократных полных заливок (большой объем данных в ед. времени), ставьте sleep(50-100 мс.) между запросами.

Желательно при отправке вести какой-то лог с id-ми записей, в случае потерь пакетов можно было установить какие из них не были обработаны шиной.

# Клиентский интерфейс обмена данными с ESB

При обмене с шиной через клиентское СУБД (MS SQL, MySQL и тд.) есть ряд служебных таблиц и хранимых процедур на стороне клиента, выполняющих роль «буферной зоны».

Основная цель интерфейса состоит в приведении к единому формату обмена с ESB.

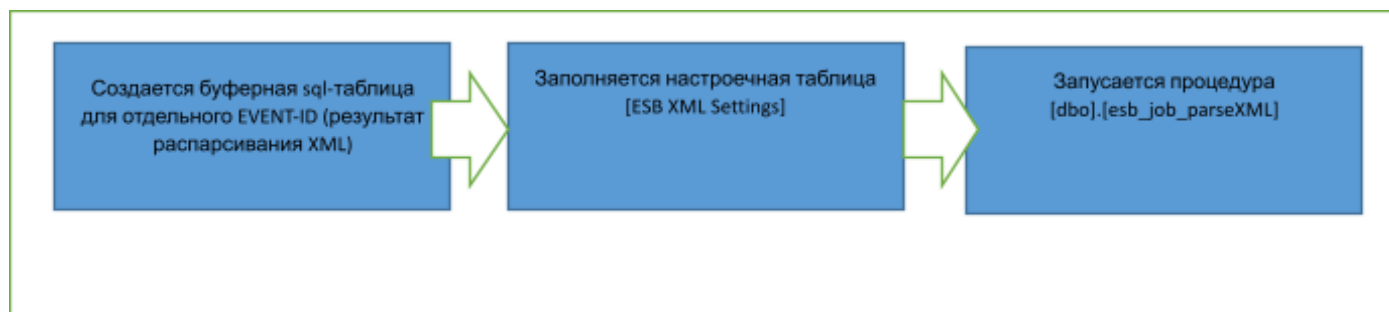
## Служебные таблицы и их описание

Название Таблицы	Описание
ESB Incoming Buffer	Буфер-таблица для входящих сообщений (XML, JSON)
ESB Outcoming Buffer	Буфер-таблица для исходящих сообщений (XML, JSON)
ESB Publish Log	Лог-таблица для входящих сообщений от шины об статусах отправок
ESB Subscribe Log	Лог-таблица для входящих сообщений от шины о статусе ответов (устарело)
ESB Event Settings	Настроечная таблица соответствия eventId и URL шины для отправки
ESB XML Settings	Настроечная таблица маппинга для входящих XML-сообщений

## Служебные хранимые процедуры и их описание

Название процедуры	Описание
[dbo].[esb_add2jn1]	Служит для заполнения таблицы [ESB Outcoming Buffer]
[dbo].[esb_job_Emit]	Служит для конечной отправки сообщения из [ESB Outcoming Buffer]
[dbo].[esb_job_parseXML]	Служит для распарсивания входящих сообщений из [ESB Incoming Buffer]

## Общая схема настройки для входящих сообщений



*Пример:*

1. Исходное сообщение находится в таблице [ESB Incoming Buffer]:

[ESB Incoming Buffer]					
entry_no	datetime	event_id	guid	body	answer
1	27.03.2019 13:52:43	NAV2ARES_EmplQual_INS	53658EA8-15BA-4398-A965-54934A841781	<employeequalification> <Person>ФЛ02195</Person> <QualificationType>1</QualificationType> <LineId>10000</LineId> <FromDate>1753-01-01</FromDate> <QualificationCode>ОХРАННИК</QualificationCode> <ToDate>1753-0101</ToDate> <Description>Охранник</Description> <DocumentNo>992867</DocumentNo> <DocumentSeries>А</DocumentSeries> <DocumentDate>2012-11-22</DocumentDate> <ExpireDocumentDate>2017-11-22</ExpireDocumentDate> > </employeequalification>	0

2. Создаем буферную SQL-таблицу для события NAV2ARES\_EmplQual\_INS

Есть правила при создании таблицы:

- все поля должны быть одним из 2х типов во избежание ошибок, связанных поддержкой нативных форматов клиентских серверов (дата время и тп): INT, NVARCHAR.
- всегда добавлять 2а служебных поля:  
[Entry No] [int] IDENTITY(1,1) NOT NULL  
[JUUID] [uniqueidentifier] NULL
- все поля должны быть NULL, кроме [Entry No]
- названия полей должны соответствовать названию полей XML-входящего сообщения, но между отдельными словами в названии должен стоять знак «пробел»

```
CREATE TABLE [dbo].[Buffer Employee Qualification](
    [Entry No] [int] IDENTITY(1,1) NOT NULL,
    [Person] [nvarchar](30) NULL,
    [Qualification Type] [nvarchar](30) NULL,
    [Line Id] [nvarchar](30) NULL,
    [Contract Id] [nvarchar](30) NULL,
    [From Date] [nvarchar](30) NULL,
    [Qualification Code] [nvarchar](30) NULL,
    [To Date] [nvarchar](50) NULL,
    [Description] [nvarchar](1024) NULL,
    [Document No] [nvarchar](30) NULL,
    [Document Series] [nvarchar](30) NULL,
    [Document Date] [nvarchar](50) NULL,
    [Expire Document Date] [nvarchar](50) NULL,
    [JUUID] [uniqueidentifier] NULL,
    CONSTRAINT [PK__Buffer B__3005472174940103] PRIMARY KEY CLUSTERED
)
    ([Entry No] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
    ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, FILLFACTOR = 85) ON [PRIMARY]
) ON [PRIMARY]
```

3. Для NAV2ARES-EmplQual\_INS настроенная таблица [ESB XML Settings] примет вид:

Results		Messages				
	id	event_id	parent_tag_path	field_tag	buffer_table_name	buffer_field_name
1	12	NAV2ARES_EmplQual_INS	employeequalification	Person	[dbo].[Buffer Employee Qualification]	[Person]
2	13	NAV2ARES_EmplQual_INS	employeequalification	QualificationType	[dbo].[Buffer Employee Qualification]	[Qualification Type]
3	14	NAV2ARES_EmplQual_INS	employeequalification	LineId	[dbo].[Buffer Employee Qualification]	[Line Id]
4	15	NAV2ARES_EmplQual_INS	employeequalification	FromDate	[dbo].[Buffer Employee Qualification]	[From Date]
5	16	NAV2ARES_EmplQual_INS	employeequalification	QualificationCode	[dbo].[Buffer Employee Qualification]	[Qualification Code]
6	17	NAV2ARES_EmplQual_INS	employeequalification	ToDate	[dbo].[Buffer Employee Qualification]	[To Date]
7	18	NAV2ARES_EmplQual_INS	employeequalification	Description	[dbo].[Buffer Employee Qualification]	[Description]
8	19	NAV2ARES_EmplQual_INS	employeequalification	DocumentNo	[dbo].[Buffer Employee Qualification]	[Document No]
9	20	NAV2ARES_EmplQual_INS	employeequalification	DocumentSeries	[dbo].[Buffer Employee Qualification]	[Document Series]
10	21	NAV2ARES_EmplQual_INS	employeequalification	DocumentDate	[dbo].[Buffer Employee Qualification]	[Document Date]
11	22	NAV2ARES_EmplQual_INS	employeequalification	ExpireDocumentDate	[dbo].[Buffer Employee Qualification]	[Expire Document Date]

4. Запуск esb\_job\_parseXML приведет к распарсиванию и созданию строки в таблице [PDC].[dbo].[Buffer Employee Qualification]

Если XML не плоский, то настройка происходит через указание полного пути к полю (FIELD\_TAG)

Пример входящего сообщения

```
<LinkedHashMap>
<sorder>
  <bitrixOrderId>142434</bitrixOrderId>
  <parentOrderId>142434</parentOrderId>
  <createdAt>2018-09-29T21:27:30+03:00</createdAt>
  <isLkOrder>true</isLkOrder>
  <contractId>AGS13-4344</contractId>
  <navClientId>K29877</navClientId>
  <email> human@gmail.com </email>
  <phone>12345678</phone>
  <customerFirstName>Александра</customerFirstName>
  <customerLastName>Карманова</customerLastName>
  <paymentType>14 ДНЕЙ</paymentType>
  <activityCompany />
  <deliveryType>ДОСТАВКА</deliveryType>
  <deliveryDate>2018-10-01T00:00:00+03:00</deliveryDate>
  <deliveryTimeFrom>2018-10-01T10:00:00+03:00</deliveryTimeFrom>
  <deliveryTimeTo>2018-10-01T15:00:00+03:00</deliveryTimeTo>
  <deliveryAddressId>2</deliveryAddressId>
  <targetSystem>nav</targetSystem>
  <managerId>OME</managerId>
  <clientComment />
  <sendInvoiceToEmail>false</sendInvoiceToEmail>
  <products>
    <article>111342</article>
    <price>1190</price>
    <quantity>3</quantity>
    <amount>3570</amount>
    <discount>20</discount>
  </products>
</sorder>
</LinkedHashMap>
```

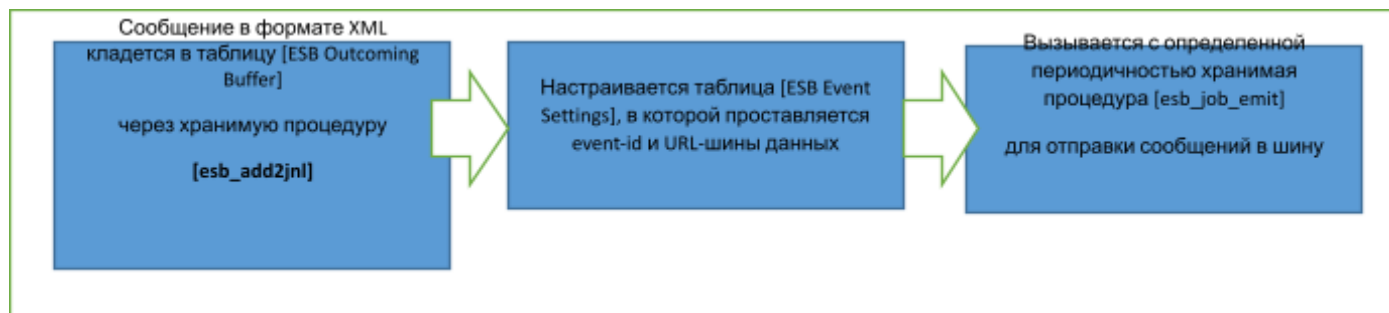
```

    <btxLineNo>1</btxLineNo>
  </products>
</products>
  <article>111017</article>
  <price>990</price>
  <quantity>2</quantity>
  <amount>1980</amount>
  <discount>35</discount>
  <btxLineNo>2</btxLineNo>
</products>
<codeEmployee />
<companyName>СИМПЛ</companyName>
<region />
<city />
<district />
<street />
<locality />
<house />
<building />
<flat />
<vladenie />
<corpus />
<office />
<additional />
</sorder>
</LinkedHashMap>

```

event	parent_tag_path	field_tag	buffer table name	buffer field name
CREATE_SORDER	LinkedHashMap/sorder	district	[Buffer Bitrix Queue]	[KLADR District]
CREATE_SORDER	LinkedHashMap/sorder	city	[Buffer Bitrix Queue]	[KLADR City]
CREATE_SORDER	...	...	...	...
	LinkedHashMap/sorder/products	btxLineNo	[Buffer Bitrix Queue]	[Btx Line No]
	LinkedHashMap/sorder/products	discount	[Buffer Bitrix Queue]	[Discount]
	...	...	...	...

## Общая схема настройки для исходящих сообщений



### 1. Пример настройки [ESB Event Settings]

*EVENT-ID определяет тип сообщения, служит для идентификации его сущности.*

*Об его названии договариваются между собой участники обмена: отправитель и конечный получатель:*

Results		Messages
event_id	esb_url	
1 NAV_to_IC_CUSTOMER	http://qvd.delta.local:8888/front/async/xml/pub/uni	
2 NAV_to_AresWeb_Qualification	http://qvd.delta.local:8888/front/async/xml/pub/uni	
3 NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/uni	
4 NAV2ARES_EmplQual_UPD	http://qvd.delta.local:8888/front/async/xml/pub/uni	
5 NAV2ARES_EmplQual_DEL	http://qvd.delta.local:8888/front/async/xml/pub/uni	

## 2. Примеры исходящих сообщений в таблице [ESB Outcoming Buffer]

Это очередь-таблица, которая обрабатывается хранимой процедурой [esb\_job\_emit]. Как только сообщение было отправлено, напротив этой строки выставляется признак is\_sent = 1.

90 %

Results Messages

	entry_no	datetime	event_id	url	guid	xml	is_sent
1	61	2019-03-27 13:48:25.590	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	53658EAB-15BA-4398-A965-54934A841781	<employeequalification><Person>ФП-02195</Person><...	1
2	62	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	3508396F-4550-4D59-BA85-9768EEC2C3F6	<employeequalification><Person></Person><Qualificatio...	1
3	63	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	FB2D731D-6250-4CFD-A457-3F90CEF31F94	<employeequalification><Person>ФП-00615</Person><...	1
4	64	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	2525A231-FF99-4CC1-B0C1-5A53987E1F3D	<employeequalification><Person>ФП-00663</Person><...	1
5	65	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	4371995E-634B-4714-B8E5-172663DBFE72	<employeequalification><Person>ФП-02033</Person><...	1
6	66	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	8539C0C7-9145-476A-961E-B8D5562EC733	<employeequalification><Person>ФП-02037</Person><...	1
7	67	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	61FBDD3F-1D95-42E5-AA51-5AD6B2E9A62D	<employeequalification><Person>ФП-02084</Person><...	1
8	68	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	8F9A35FD-EB6C-49F1-8E43-FA19D21FF1BD	<employeequalification><Person>ФП-02190</Person><...	1
9	69	2019-03-27 16:23:09.150	NAV2ARES_EmplQual_INS	http://qvd.delta.local:8888/front/async/xml/pub/...	80FD8CC9-40BE-44C3-B465-119B18D0CB93	<employeequalification><Person>ФП-02191</Person><...	1

Данные из буферной таблицы можно переотправлять, например вернув значение is\_sent = 0.

## Контроль отправки сообщений

После того как сообщение было отправлено в шину, ожидается ответ от шины в таблицу [ESB Publish Log] об статусе отправки: успешно/ошибка.

Пример сообщение не было отправлено конечному получателю из-за ошибке недоступности клиентского сервера (поля Error\_id и Details говорят об ошибке):

```
SELECT *
FROM [RESTORE].[dbo].[ESB Publish Log]
```

Results		Messages								
entry_no	datetime	event_id	guid	publisher_id	subscriber_id	subscriber_handler	subscriber_store_class	error_id	details	
1	151	21.03.2019 11:23:55	NAV2ARES_EmplQual_INS	55CEABED-060C-4189-94D0-C576352835B2	NAV	ONE	https://personal2.ares.ru/index4.php?language=ru...	com.sample.server.domain.contract.EnrPubMsg	Pub Task	HttpInpl, url: < 404 Not Found >, https://
2	152	21.03.2019 11:23:56	NAV2ARES_EmplQual_INS	CD2739E8-8627-4A06-8F5E-BEBA0ABD45DA	NAV	ONE	https://personal2.ares.ru/index4.php?language=ru...	com.sample.server.domain.contract.EnrPubMsg	Pub Task	HttpInpl, url: < 404 Not Found >, https://
3	153	21.03.2019 11:23:57	NAV2ARES_EmplQual_INS	D99011E4-1084-4A2E-85AA-935FD878875A	NAV	ONE	https://personal2.ares.ru/index4.php?language=ru...	com.sample.server.domain.contract.EnrPubMsg	Pub Task	HttpInpl, url: < 404 Not Found >, https://
4	154	21.03.2019 11:23:58	NAV2ARES_EmplQual_INS	90709842-2BB7-4C6A-8F15-58D814AB35B2	NAV	ONE	https://personal2.ares.ru/index4.php?language=ru...	com.sample.server.domain.contract.EnrPubMsg	Pub Task	HttpInpl, url: < 404 Not Found >, https://
5	155	21.03.2019 11:23:59	NAV2ARES_EmplQual_INS	8AF59F01-DC5F-4510-8755-312786EC2BA9	NAV	ONE	https://personal2.ares.ru/index4.php?language=ru...	com.sample.server.domain.contract.EnrPubMsg	Pub Task	HttpInpl, url: < 404 Not Found >, https://
6	156	21.03.2019 11:24:00	NAV2ARES_EmplQual_INS	F60D48F7-FD2B-4C5F-9818-42BF62629C05	NAV	ONE	https://personal2.ares.ru/index4.php?language=ru...	com.sample.server.domain.contract.EnrPubMsg	Pub Task	HttpInpl, url: < 404 Not Found >, https://

**ВАЖНО:** Отправляемые сообщения нужно сопоставлять по уникальной метке сообщения GUID, которая генерируется каждый раз при заполнении таблицы [ESB Outcoming Buffer] через хранимую процедуру [dbo].[esb\_add2jnl]

Можно делать новую переотправку заново сгенерированных сообщений, тогда уникальная метка GUID будет присвоена заново, при этом event-id и тело сообщения остается тем же, что и было.

Пример, что сообщение было отправлено успешно конечному получателю:

(ERROR\_ID и Details – пустые)

Results		Messages										
	entry no	datetime	event_id	guid	publisher_id	subscriber_id	subscriber_handler	subscriber_store_class	error_id	details		
34	184	26.03.2019 16:37:13	NAV2ARES_EmpQual_INS	7FA40632-2F7D-411D-B93E-94C05AA2162B	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
35	185	26.03.2019 16:37:14	NAV2ARES_EmpQual_INS	8764333F-270A-4212-87BF-22420808A22D	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
36	186	26.03.2019 16:37:15	NAV2ARES_EmpQual_INS	26808496-CCAF-47CD-AD99-09C7C52CA158	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
37	187	26.03.2019 16:37:16	NAV2ARES_EmpQual_INS	C4854E38-3CE1-4089-8E39-586217E8F403	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
38	188	26.03.2019 16:37:18	NAV2ARES_EmpQual_INS	E5863545-9216-476A-92AA-3868DC4D633C	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
39	189	26.03.2019 16:37:19	NAV2ARES_EmpQual_INS	6331DD24-19E0-4CE9-ADAA-F54AF69D05AA	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
40	190	26.03.2019 16:37:20	NAV2ARES_EmpQual_INS	096AC487-CDFF-4013-AA24-18DCA22E274F	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
41	191	26.03.2019 16:40:17	NAV2ARES_EmpQual_INS	59153002-4440-40DF-8702-36F868652710	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
42	192	26.03.2019 16:40:18	NAV2ARES_EmpQual_INS	389C196F-CEB6-493E-8CE0-48CE4158A02	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
43	193	26.03.2019 16:40:19	NAV2ARES_EmpQual_INS	37C088A7-D961-430F-886E-9CE1524F485A	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
44	194	26.03.2019 16:40:20	NAV2ARES_EmpQual_INS	2E6DCD8E-9C48-41A5-8D2C-A7F34736F35E	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
45	195	26.03.2019 16:40:21	NAV2ARES_EmpQual_INS	9E6F2147-907B-4C3A-B87B-6EC3E388CCBE	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
46	196	26.03.2019 16:40:22	NAV2ARES_EmpQual_INS	024F9227-7917-4941-A232-359313948898	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
47	197	26.03.2019 16:40:23	NAV2ARES_EmpQual_INS	F1301966-5686-4ADF-9205-39456DA8B60B	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		
48	198	26.03.2019 16:40:24	NAV2ARES_EmpQual_INS	7EBF8FE9-CFA3-42F1-91AF-05E75F99935B	NAV	ONE	https://personal2.ares.ru/index.php?language=ru...	NULL	NULL	NULL		

# Управление безопасностью ESB (#1.1)

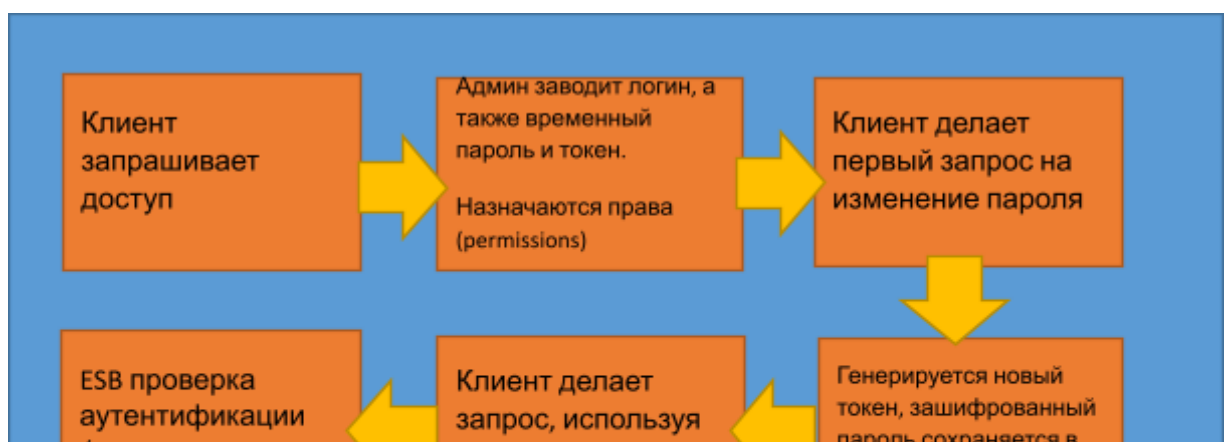
## Концепция

В основе управления безопасностью при эксплуатации корпоративной шины данных лежит концепция [JSON Web Token](#) (JWT).

Текущая реализация функционала с использованием JWT включает:

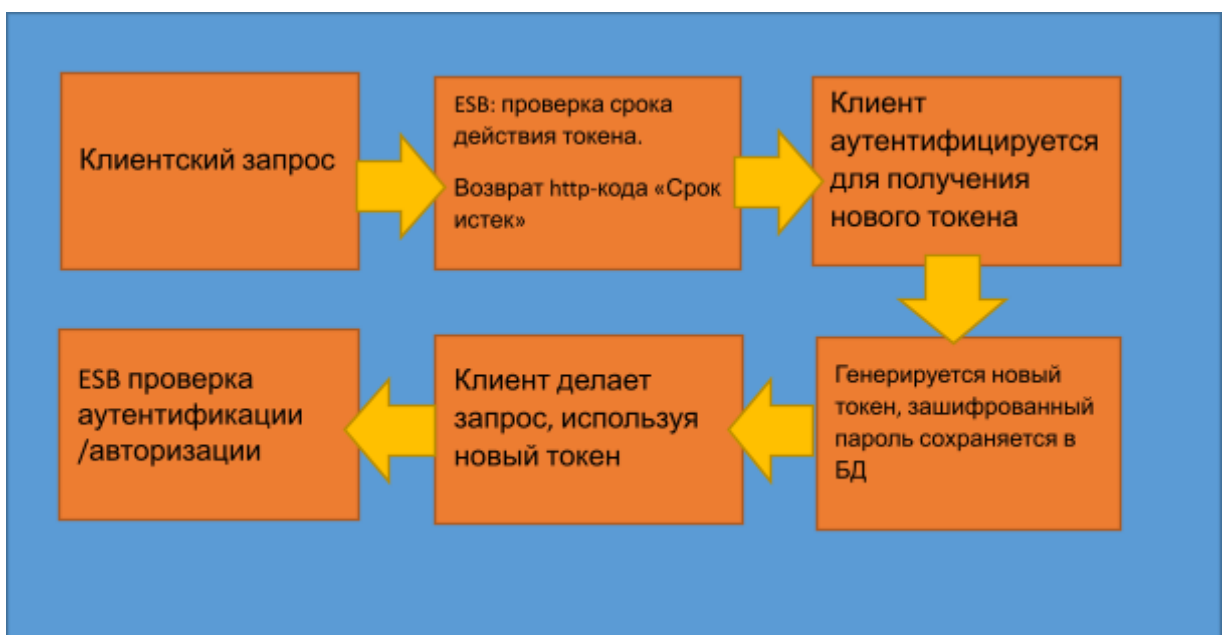
- использование https-протокола, защищающий от middleware-атаки
- вместо пароля доверенный пользователь отправляет токен, который расшифровать невозможно без закрытого ключа
- все пароли хранятся в зашифрованном виде в базе данных, поэтому даже при наличии root-прав (dbowner permissions) невозможно воспользоваться зашифрованной информацией в злостных целях.
- гибкое управление логинами, в т.ч. настраиваемый срок действия индивидуального токена.
- двузонное разделение API на защищенный/незащищенный через настройки базы данных
- дополнительная стойкая защита от случайных/злоумышленных действий root-пользователя базы данных

## Work-Flow «Первичная настройка»





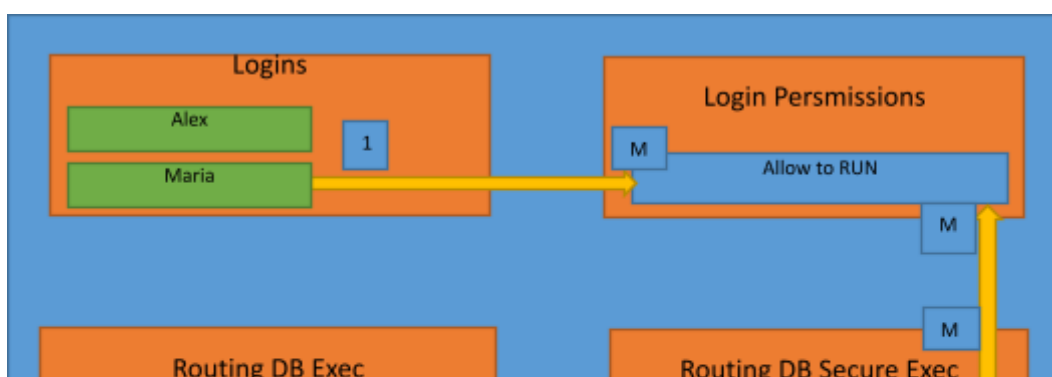
## Work-Flow «Истекий срок действия токена»



## Схема «Сущность-Связь»

База данных ESB включает несколько сущностей:

- **Logins** – таблица для логинов и паролей, а также срок действия токена
- **[Routing DB Secure Exec]** – содержит команду (инструкцию) запуска клиентской хранимой процедуры, также настройку соответствия (маппинг) ESB API-параметров и параметров хранимой процедуры.
- **[Logins Permissions]** – настройка разрешений для пользователя запускать хранимую процедуру через ESB API.
- **[Routing DB Exec]** – настройка вызова открытых клиентских процедур для общего доступа + маппинг параметров.



## Как это работает

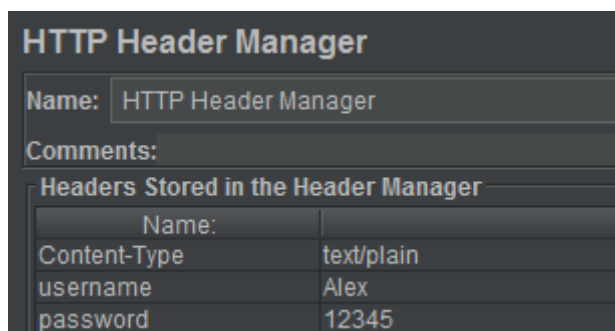
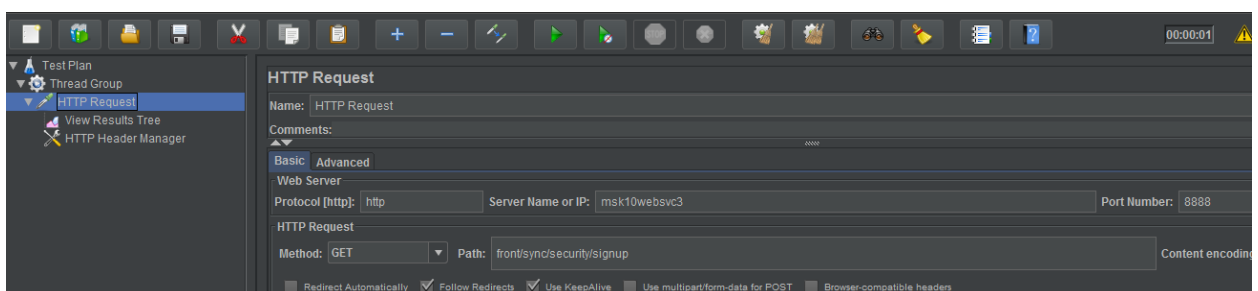
1. Клиентской системой (1C, NAV и др.) опубликована хранимая процедура/http api, доступ к которой должны иметь только доверенные пользователи ESB. Для этого в хранимую процедуру достаточно всего лишь добавить параметр @\_secured:

```
CREATE PROC [dbo].[web_getTestSecurity] @_customer_id NVARCHAR(20)= NULL,  
                                         @_secured INT = 0  
  
WITH EXECUTE AS OWNER  
AS BEGIN  
SET NOCOUNT ON;  
  
IF @_secured = 0 RETURN;  
  
SELECT 'well done' as status  
END
```

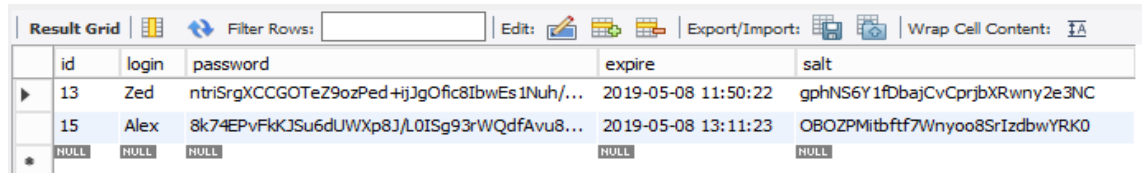
2. Администратор ESB заводит настройку в определённой таблице для хранимой процедуры с указанием всех необходимых параметров, но без указания параметра @\_secured:

id	endpoint_id	method_name	function_call_statement	params_web_func_map	result_handler	description
1	NAV	nav_get_customer	EXECUTE [dbo].[esb_GetCustomerXML]	custNo:@CustNo;companyName:@CompanyName	flatXML	NULL
2	NAV	nav_test_secure	EXEC [dbo].[web_getTestSecurity]	custNo:@_customer_id	flatJSON	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL

3. Получатель данных регистрируется на сервере ESB посредством запроса через указание логина/пароля в заголовке:

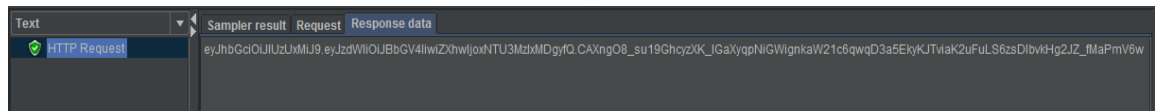


4. Пароль пользователя шифруется с использованием секретного ключа и сохраняется в БД ESB:

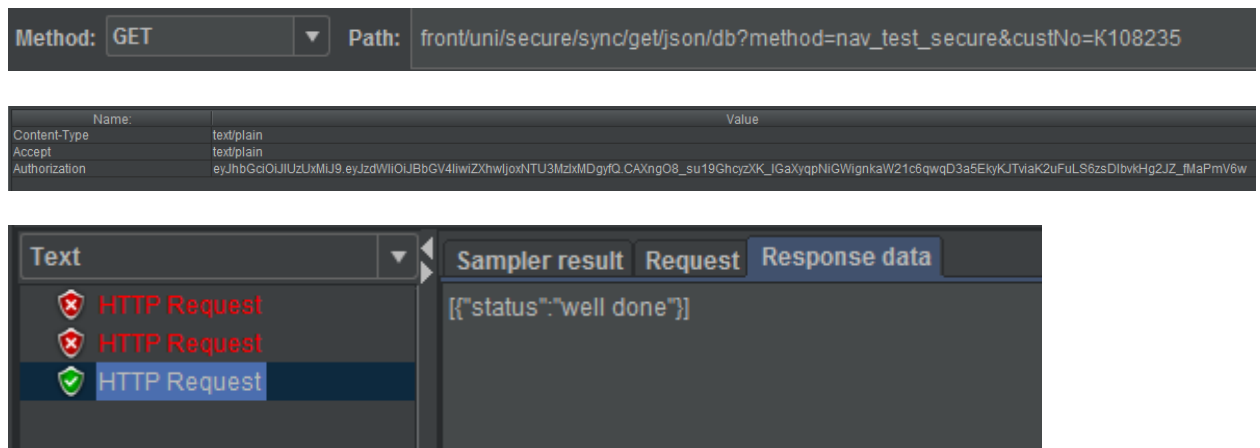


	id	login	password	expire	salt
▶	13	Zed	ntriSrgXCCGOTeZ9ozPed+ijJgOfic8IbwEs1Nuh/...	2019-05-08 11:50:22	gphNS6Y1fDbajCvCprjbXRwny2e3NC
	15	Alex	8k74EPvFkKJSu6dUWXp8J/L0ISg93rWQdfAvu8...	2019-05-08 13:11:23	OBOZPMitbftf7Wnyoo8SrIzdbwYRK0
*		NULL	NULL	NULL	NULL

5. Пользователь получает токен, который сохраняет у себя:



6. Далее пользователь использует выданный токен в заголовке запроса для получения данных:



## Вопросы по безопасности

- Почему нельзя перехватить запрос и воспользоваться перехваченным токеном?

*Обмен между доверенным клиентом и ESB происходит по защищенному каналу HTTPS с использованием шифрованных транспортных механизмов TLS/SSL. Даже при перехвате токена пользоваться им можно лишь ограниченное время и зависит от установки настроек в БД ESB.*

- Как часто пользователь должен аутентифицироваться на сервере ESB?

*Как только заканчивается период действия токена (Expire Date), пользователю необходимо снова залогиниться на сервере используя старый и новый пароль.*

- Какие шаги при заведении нового пользователя?

*Новому пользователю выдает логин и временный токен, созданный администратором ESB.*

*В последующих запросах токен используется в заголовке Authorization.*

*По окончании срока действия токена пользователь будет извещен специальным HTTP-кодом в ответе.*

- Могут ли перехватить пароль внутри домена ESB?  
*Когда пароль попадает во FRONT-сервис, он сразу же шифруется и далее в зашифрованном виде сохраняется в БД ESB. Секретный ключ неизвестен никому, в т.ч. администратору ESB.*

- Зачем нужен @\_secured в параметрах хранимых процедур/API-клиента?

*Данный параметр является признаком, посылаемый ESB после проверки аутентификации пользователя. Таким образом, даже если случайно переместить вызов хранимой процедуры в зону свободного доступа, вызова не произойдет.*

# Мониторинг логов корпоративной шины данных

## Поддерживаемый контрольные функции

Для мониторинга состояния корпоративной шины данных и сообщений, участвующих в обмене, предусмотрено 5 способов контроля:

- общая статистика – представляет собой отдельный веб-сервер, имеющий доступ к текстовым лог-файлам ESB.
- записи в таблицах базы данных ESB (jdb)
- записи в текстовых лог-файлах, создаваемые в процессе работы ESB
- возвращаемые статусы об отправке данных в систему-отправитель (таблица [ESB Publish Log])
- непосредственный вызов утилитных веб-методов из браузерной строки, предоставляющих детальную информацию об ошибках внутри ESB.

Все способы за исключением веб-сервера общей статистики подходят для детального анализа каждого сообщения.

## Детальный анализ логов

Используется в случаях, когда необходимо узнать детальную информацию по определенному сообщению.

Каждое сообщение имеет уникальную метку (GUID), генерируемую системой-отправителем. Значение этой метки активно логируется наряду с другими данными шиной данных и используется для быстрого поиска в лог-таблицах и лог-файлах.

Сообщение можно контролировать на уровне:

- поступило ли сообщение в шину (+ на какую ноду, если кластерное решение)
- было ли отправлено сообщение от одного микросервиса в другой внутри шины
- поступило ли сообщение в нужный Task (существуют задачи по отправке данных через http-протокол, а также другие задачи, отвечающие за запись в базу данных и т.д.)
- текущий вид трансформированного сообщения, перед отправкой к системе-приемщику
- статусы отправки: успешно, ошибки

## Общая статистика

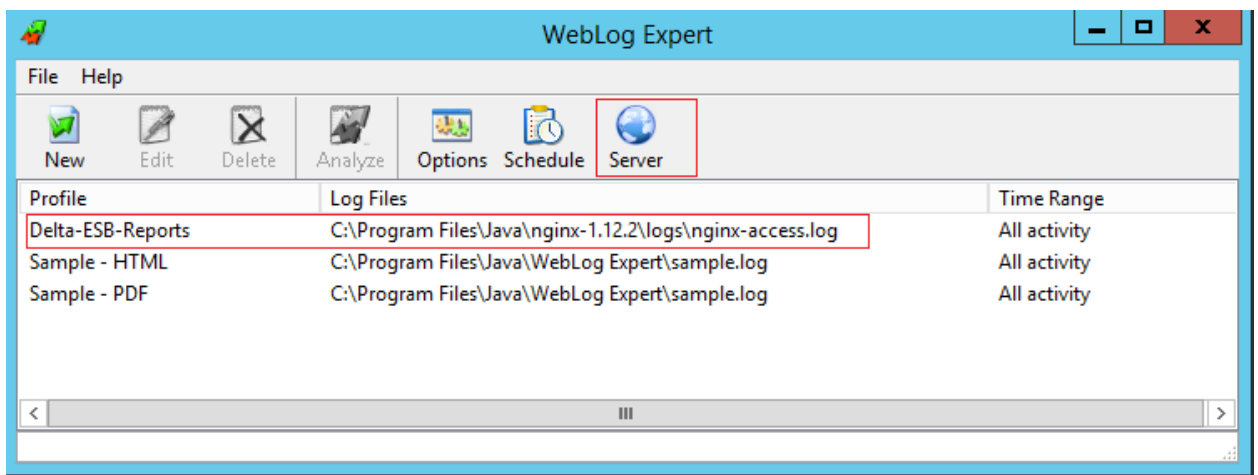
В процессе обменов важно руководствоваться агрегированными данными:

сколько всего сообщений было отправлено на тот или иной адрес, пиковые нагрузки в течение дня (недели), общее количество ошибок в разрезе адресов (URLs).

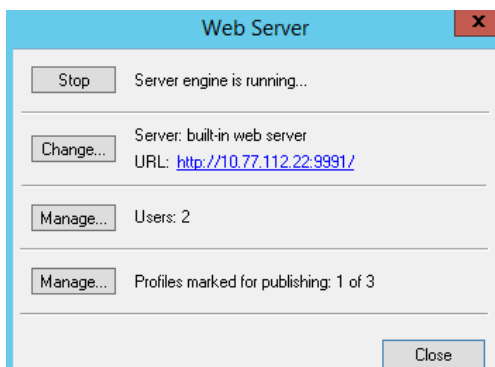
Веб-сервер отчетности (WebLog Expert) ежедневно собирает и анализирует текстовые лог-файлы ESB и формирует веб-отчеты, доступ к которым имеют авторизованные пользователи через браузер.

Настройка:

1. Для настройки веб-сервера запускаем приложение, затем добавляем новую строку (указываем источник данных NGNIX nginx-access.log).
2. Нажимаем кнопку Server:

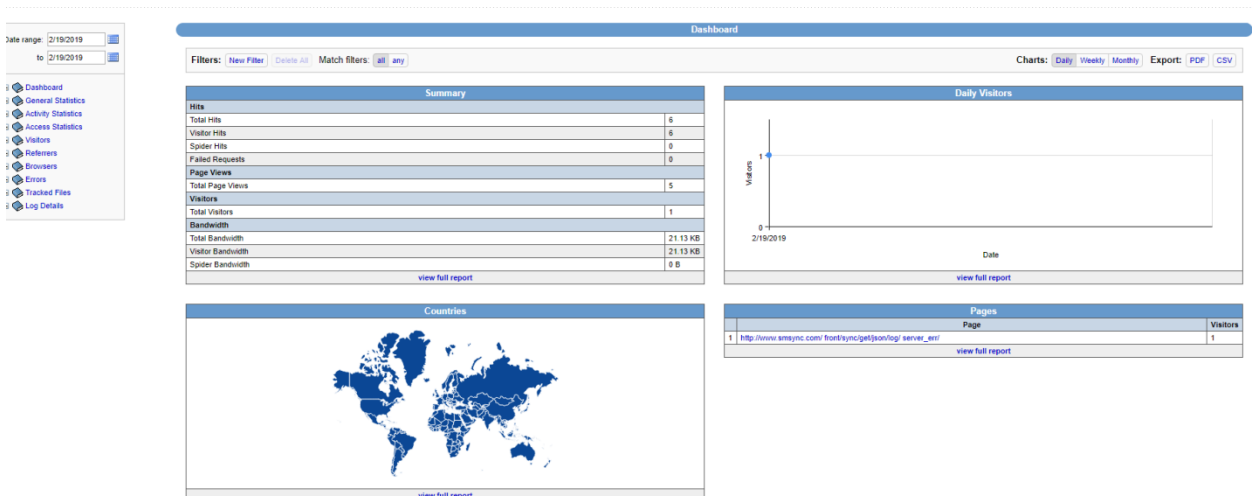


3. В открывшемся окне нажимаем Change, выбираем ip адрес вместо localhost.
4. Нажимаем Manage – добавляем локальных пользователей и запускаем сервер:



5. Через браузер вводим адрес сервера URL, который указали в настройках, вводим свои логин и пароль.

На новой странице кликаем на ссылку с названием отчета:



Утилитные веб-методы из браузерной строки

Позволяют быстро получить доступ к статусам отправок сообщений. Эти методы можно вызывать в сторонних сервисах, автоматизирующих процесс контроля и мониторинга за шиной.

Иногда достаточно указать адрес в браузерной строке с нужным параметром, чтобы вручную получить информацию о статусе сообщения.

Методы разделены по группам:

- вызов настроек и получения сведений конфигурации ESB
- данные по статусам сообщений

#### 1. Вызов настроек:

Веб-метод	Описание	Настроечная таблица ESB
<b><code>front/util/cache/check/retranslate?</code></b> <b><code>key=&lt;ключ из таблицы ESB&gt;</code></b> <b>GET-запрос</b> <b>пример:</b> <b><code>front/util/cache/check/retranslate?</code></b> <b><code>key=sorders_from_1c</code></b>	Вывести на экран значение URL-а ретрансляции, которое хранится в кэше ESB. Используется, если данные не возвращаются с этого URL-а.	таблица: <a href="#">[jdb].[router redirect]</a>
<b><code>front/util/cache/check/allRetranslates</code></b> <b>GET-запрос</b>	Вывести на экран список всех URL-а для ретрансляций, находящихся в кэше ESB.	
<b><code>front/util/cache/check/retranslates/refresh</code></b> <b>GET-запрос</b>	Обновить кэш значениями из базы данных ESB.	
<b><code>front/sync/get/json/log/server_err?</code></b> <b><code>eventId=&lt;ключ из таблицы ESB&gt;</code></b> <b>GET-запрос</b> <b>пример:</b> <b><code>front/sync/get/json/log/server_err?</code></b> <b><code>eventId=create_sorder</code></b>	Узнать о наличии серверных ошибок, связанных с отправками сообщений. Возвращает список неотправленных сообщений.	таблица: <a href="#">[jdb].[log pub]</a>  хранимая процедура: <a href="#">[dbo].[get_publog_serverErr]</a>
<b><code>front/util/cache/allTimeouts</code></b> <b>GET-запрос</b>	Вывести на экран список значений политики таймаутов, находящихся в кэше ESB.	таблица: <a href="#">[jdb].[timeout policies]</a>
<b><code>front/util/cache/timeouts/refresh</code></b> <b>GET-запрос</b>	Обновить кэш значениями из базы данных ESB.	

#### 2. Данные по статусам сообщений:

Веб-метод	Описание	Настроечная таблица ESB
<b><code>front/util/log/success?</code></b> <b><code>eventId=&lt;ключ события&gt;</code></b>  <b><code>front/util/log/success?</code></b> <b><code>juuid=&lt;уникальная метка сообщения&gt;</code></b>  <b>GET-запрос</b> <b>примеры:</b> <b><code>front/util/log/success?</code></b> <b><code>eventId=sorders_from_1c</code></b>  <b><code>front/util/log/success?</code></b> <b><code>juuid=B36B560F-607C-41E3-BB83-3D9A2F5984F6</code></b>	Необходимо для получения статуса, было ли отправлено сообщение из шины к конечной системе.	таблица: <a href="#">[jdb].[log pub success]</a>

<code>front/util/log/err?</code> <code>eventId=&lt;ключ события&gt;</code>  <code>front/util/log/err?</code> <code>juuid=&lt;уникальная метка сообщения&gt;</code>  GET-запрос	Необходимо для получения статуса, возникла ли ошибка по данному сообщению во время отправки из шины.	таблица: <a href="#">[ldb].[log pub err]</a>
<code>front/util/hot</code> <code>eventId=&lt;ключ события&gt;</code>  <code>front/util/hot?</code> <code>juuid=&lt;уникальная метка сообщения&gt;</code>  GET-запрос	Получение статус, поступило ли сообщение на бэкэнд-шины (bridge-service).	таблица: <a href="#">[ldb].[hot pub]</a>

## Данные из лог-таблиц базы данных ESB

Все сообщения (POST-запросы) поступают на front-service ESB, затем обрабатываются и передаются на bridge-service ESB, который занимается конечной отправкой сообщений. Все это время происходит заполнение лог-таблиц.

Общая схема логирования сообщения:

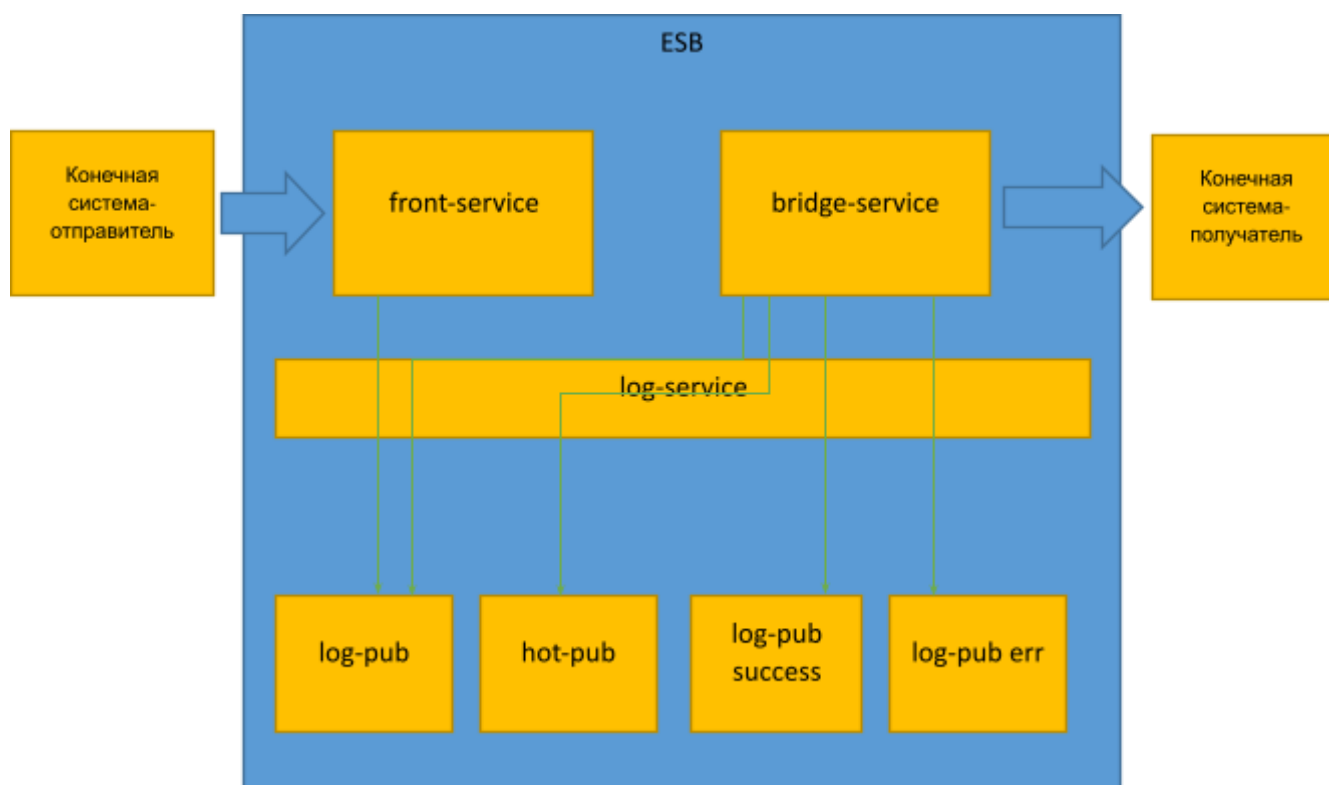


Таблица [log-pub]:



Название поля	Описание	Кем создается значение
[publisher_id]	Идентификатор отправителя, передаваемый вместе с сообщением	Конечная система-отправитель
[subscriber_handler]	URL конечной системы-получателя, куда было отправлено сообщение	Шина данных
[uuid]	Уникальная метка сообщения, генерируемая системой-отправителем	Конечная система-отправитель
[event_id]	Тип события, передаваемая отправителем	Конечная система-отправитель
[header_value]	Обозначает класс-сообщения (структура)	Шина данных
[service_id_from]	Идентификатор инстанса шины данных, от кого ушло сообщение	Шина данных
[service_role_from]	Идентификатор роли шины данных, от кого ушло сообщение: «Front» или «Bridge».	Шина данных
[service_role_to]	Устарело (будет удалено)	Шина данных
[service_role_to]	Устарело (будет удалено)	Шина данных
[outcoming_datetime]	Время отправки из сервиса	Шина данных
[incoming_datetime]	Время поступления в сервис	Шина данных
[log_datetime]	Время момента записи в лог-таблицу	Шина данных
[logger_id]	Идентификатор лог-сервиса	Шина данных
[method_handler]	Идентификатор веб-метода из front-сервиса, куда впервые поступило сообщение	Шина данных

Таблица [hot-pub]:

Название поля	Описание	Откуда берется значение
[sender_id]	Идентификатор отправителя	Из сообщения конечной системой-отправителя
[event_id]	Тип события	Из сообщения конечной системой-отправителя
[uuid]	Уникальная метка сообщения	Из сообщения конечной системой-отправителя
[log_datetime]	Время логирования	Шина данных
[msg]	Тело сообщения	Из сообщения конечной системой-отправителя

Таблица [log-pub-success]:

Название поля	Описание	Откуда берется значение
[sender_id]	Идентификатор отправителя	Из сообщения конечной системой-отправителя
[subscriber_id]	Идентификатор подписчика	Шина данных
[event_id]	Тип события	Из сообщения конечной системой-отправителя
[uuid]	Уникальная метка сообщения	Из сообщения конечной системой-отправителя
[log_datetime]	Время логирования	Шина данных

Таблица [log-pub-err]:

Название поля	Описание	Откуда берется значение
[sender_id]	Идентификатор отправителя	Из сообщения конечной системой-отправителя
[subscriber_id]	Идентификатор подписчика	Шина данных
[event_id]	Тип события	Из сообщения конечной системой-отправителя
[uuid]	Уникальная метка сообщения	Из сообщения конечной системой-отправителя
[log_datetime]	Время логирования	Шина данных

[error_id]	Код ошибки	Шина данных
[details]	Описание ошибки	Шина данных
[is_external_error]	Признак ошибки: ошибка по причине, возникшей на стороне конечной-системы получателя, ошибка внутри ESB	Шина данных

## Записи в текстовых лог-файлах в процессе работы ESB

Все файлы, формируемые на стороне Apache Tomcat, создаются в директории logs (*пример, Apache Software Foundation\Tomcat 8.5\logs*).

При первом запуске/рестарте Apache Tomcat сервера в пустой директории logs будут созданы следующие файлы приложения ESB:

- *front-log4.txt*
- *bridge-log4.txt*
- *log-erros.txt*

### Стандартные лог-файлы Apache Tomcat

*tomcat8-stdout.<текущая дата>.txt*

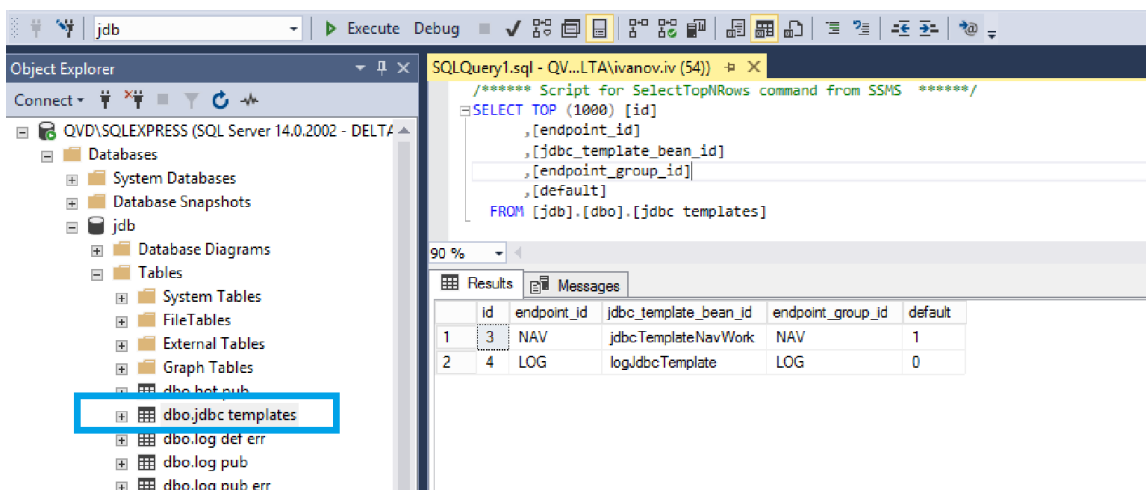
**Показывает «готовность» внутренних служб шины к работе.**

В этом файле указывается кол-во считанных записей из конфигурационных таблиц БД ESB:

- bridge:::sessionFactories: 2 – указывает, что было считано 2е настроечные строки из таблицы [session factories]
- bridge:::jdbcTemplates: 2 – указывает, что было считано 2е настроечные строки из таблицы [jdbc templates]

Также простая детализация этих строк:

- bridge:::sessionFactories: NAV
- bridge:::sessionFactories: LOG
- bridge:::jdbcTemplates: NAV
- bridge:::jdbcTemplates: LOG

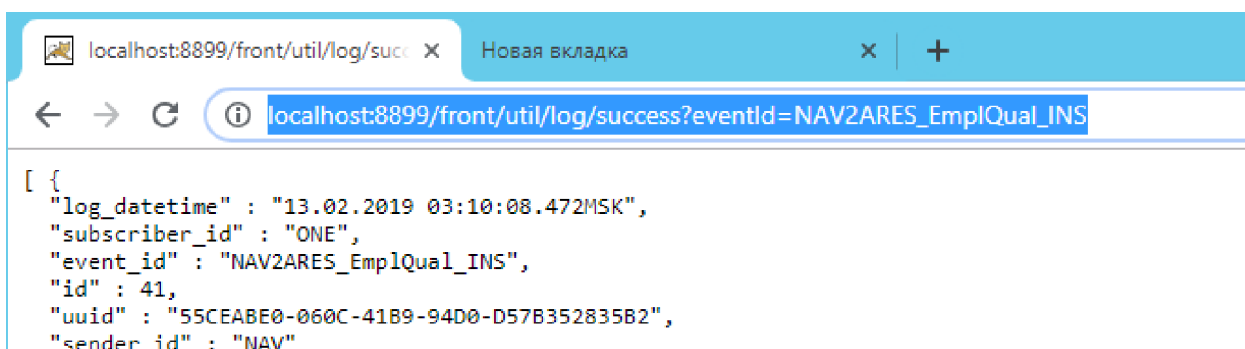


Наряду с этим отображаются обработчики, загруженные в память:

- com.simple.server.tasks.DispatcherTask@5f8525ea
- com.simple.server.tasks.DispatcherTask@5d9cc1f4
- com.simple.server.tasks.ClientSenderTask@bb68912
- com.simple.server.tasks.ClientSenderTask@6344f373
- com.simple.server.tasks.LogSenderTask@6431982d
- [com.simple.server.tasks.LogSenderTask@69000601](#)

*localhost\_access\_log.<дата>.txt*

Создает запись при каждом внешнем обращении к шине (POST-,GET-), пример:



будет создана строка:

0:0:0:0:0:0:1 - - [21/Mar/2019:10:06:22 +0300] "GET /front/util/log/success?eventId=NAV2ARES\_EmplQual\_INS HTTP/1.1" 200 6422

## Лог-файлы ESB

*bridge-log4.txt*

Логируются данные при отправке POST-запросов и при записи в БД ESB JDB.

При отправке сообщения создаются следующие записи:

1. Перед отправкой –
  - время
  - адрес доставки
  - тип запроса – «GET\POST»
  - тело конечного сообщения

- уникальная метка сообщения
- идентификатор потока, отвечающий за отправку

**2019-02-20 13:23:56** DEBUG HttpImpl:100 - [Http] [PRE POST] juuid:[ F60048F7-FD2B-4C5F-9B18-42BFB2629C05 ], [ 20.02.2019 13:23:56 ], body:[  
{"clazz":"com.simple.server.domain.contract.UniMinMsg","juuid":"F60048F7-FD2B-4C5F-9B18-42BFB2629C05","eventId":"NAV2ARES\_EmplQual\_INS","body":{"employeequalification":{"DocumentSeries":"B","DocumentNo":"017210","Description":"Охранник","ExpireDocumentDate":"2022-11-03","QualificationType":1,"QualificationCode":"ОХРАНИК","FromDate":"1753-01-01","ToDate":"1753-01-01","Person":"ФЛ-02260","DocumentDate":"2017-11-03","LineNo":10000}}}, url:[  
[https://personal2.ares.ru/index.php?language=ru&page=nav\\_exchange](https://personal2.ares.ru/index.php?language=ru&page=nav_exchange) ], thread id: [ 510 ], thread name: [ pool-28-thread-1 ]

## 2. После отправки –

- время
- адрес доставки
- уникальная метка сообщения
- идентификатор потока, отвечающий за отправку
- статус ошибки (описание) или успеха

*в случае успешной отправки:*

**2019-02-12 23:41:47** DEBUG HttpImpl:106 - [Http] [POST ERR] juuid:[ F60048F7-FD2B-4C5F-9B18-42BFB2629C05 ], [ 20.02.2019 13:23:56 ], url:[ [https://personal2.ares.ru2/index.php?language=ru&page=nav\\_exchange1](https://personal2.ares.ru2/index.php?language=ru&page=nav_exchange1) ] I/O error on POST request for "https://personal2.ares.ru2/index.php": personal2.ares.ru2; nested exception is java.net.UnknownHostException: personal2.ares.ru2, thread id: 33, thread name: pool-6-thread-1

*в случае ошибки:*

**2019-02-20 13:23:56** DEBUG HttpImpl:127 - [Http] [POST OK] juuid:[ F60048F7-FD2B-4C5F-9B18-42BFB2629C05 ], [ 20.02.2019 13:23:56 ], url:[ [https://personal2.ares.ru/index.php?language=ru&page=nav\\_exchange](https://personal2.ares.ru/index.php?language=ru&page=nav_exchange) ], thread id: 510, thread name: pool-28-thread-1

## 3. Соответствующая запись в лог-таблицу базы данных ESB:

*в случае успешной отправки:*

**2019-02-20 13:23:56** INFO PubTask:634 - [PubTask] [log pub success] - juuid:[ F60048F7-FD2B-4C5F-9B18-42BFB2629C05 ] [ 20.02.2019 13:23:56 ], eventId:[ NAV2ARES\_EmplQual\_INS ], sender:[ NAV ], subscriber:[ ONE ], auth:[ false ], sub handler:[ [https://personal2.ares.ru/index.php?language=ru&page=nav\\_exchange](https://personal2.ares.ru/index.php?language=ru&page=nav_exchange) ], thread id:[ 510 ], thread name:[ pool-28-thread-1 ]

*в случае ошибки:*

**2019-02-12 23:41:47** INFO PubTask:532 - [PubTask] [log pub err] - 12.02.2019 23:41:47 F60048F7-FD2B-4C5F-9B18-42BFB2629C05, event: NAV\_to\_AresWeb\_EmployeeQualification\_INS, sender: NAV, subscriber: ONE, auth: false, sub handler: null, thread id: 33, thread name: pool-6-thread-1

# USE-CASES

## 1. Проверка отправки POST-запроса

```
-- Переотправляем данные в шину
USE [RESTORE]
GO
-- Подчистим логи в наве
DELETE
FROM [RESTORE].[dbo].[ESB Publish Log]

UPDATE [ESB Outcoming Buffer]
```

```

SET [is_sent] = 0          -- меняем статусы на готовность к отправке
WHERE [event_id] LIKE 'NAV2ARES_EmplQual_INS'

-- отправляем
EXEC [dbo].[esb_job_Emit]

```

```

-- Проверка работоспособности сервисов:
-- должно быть по 2-е записи на одно сообщение
USE [JDB]
GO
SELECT *
FROM [jdb].[dbo].[log pub]
ORDER BY 1 DESC

```

id	publisher_id	subscriber_handler	uuid	event_id	header_value	service_id_from	service_id_to	service_role_from	service_role_to	outcoming_datetime
1	277	NAV	https://personal2.ares.ru/index.php?language=ru&...	D84645C0-010D-4194-9CAC-7F29C8923BC5	NAV2ARES_EmplQual_INS	BusPubMsg	bridge_msk10websvcDEV	NULL	BRIDGE-SERVICE	NULL
2	276	NAV	NULL	D84645C0-010D-4194-9CAC-7F29C8923BC5	NAV2ARES_EmplQual_INS	UniMsg	front_msk10websvc2777	NULL	FRONT-SERVICE	NULL

```

-- Проверка успешно ли произошла отправка
SELECT *
FROM [jdb].[dbo].[log pub success]
ORDER BY 1 DESC

```

90 %

Results

Messages

	id	sender_id	subscriber_id	event_id	uuid	log_datetime
1	80	NAV	ONE	NAV2ARES_EmplQual_INS	D84645C0-010D-4194-9CAC-7F29C8923BC5	21.03.2019 10:27:52.861MSK
2	79	NAV	ONE	NAV2ARES_EmplQual_INS	08429F31-E84D-46C8-A4A0-A371AB68F135	21.03.2019 10:27:52.861MSK
3	78	NAV	ONE	NAV2ARES_EmplQual_INS	A174790F-8AD4-4183-930D-33A11691CD24	21.03.2019 10:27:46.757MSK
4	77	NAV	ONE	NAV2ARES_EmplQual_INS	BF5E03B4-DE4D-499B-98FA-773C0CA5DACC	21.03.2019 10:27:46.757MSK
5	76	NAV	ONE	NAV2ARES_EmplQual_INS	F60048F7-FD2B-4C5F-9B18-42BFB2629C05	21.03.2019 10:27:46.757MSK
6	75	NAV	ONE	NAV2ARES_EmplQual_INS	8AF59F01-DC5F-4510-B755-312786EC2BA9	21.03.2019 10:27:46.757MSK
7	74	NAV	ONE	NAV2ARES_EmplQual_INS	90709B42-2BB7-4C6A-BF15-58D814AB35B2	21.03.2019 10:27:46.757MSK
8	73	NAV	ONE	NAV2ARES_EmplQual_INS	D99D11E4-1084-4A2E-85AA-935FD879875A	21.03.2019 10:27:46.757MSK

```

-- Проверка не было ли ошибок
SELECT *
FROM [jdb].[dbo].[log pub err]
ORDER BY 1 DESC

```

-- Проверка через браузерную строку

[http://localhost:8899/front/util/log/success?eventId=NAV2ARES\\_EmplQual\\_INS](http://localhost:8899/front/util/log/success?eventId=NAV2ARES_EmplQual_INS)

[http://localhost:8899/front/util/log/err?eventId=NAV2ARES\\_EmplQual\\_INS](http://localhost:8899/front/util/log/err?eventId=NAV2ARES_EmplQual_INS)

-- Проверка через лог-файл bridge-log4.txt

```

2019-03-21 10:27:46 DEBUG HttpImpl:101 - [Http] [PRE POST] juuid:[ D84645C0-010D-4194-9CAC-7F29C8923BC5 ], body:[
{"clazz":"com.simple.server.domain.contract.UniMinMsg","juuid":"D84645C0-010D-4194-9CAC-7F29C8923BC5","eventId":"NAV2ARES_EmplQual_INS","body":{"employeequalification":{"DocumentSeries":{"A"},"DocumentNo":"993454","Description":{"Охранник"},"ExpireDocumentDate":{"2018-02-11"},"QualificationType":1,"QualificationCode":{"ОПАННИК"},"FromDate":{"1753-01-01"},"ToDate":{"1753-01-01"},"Person":{"ФЛ-02272"},"DocumentDate":{"2013-02-11"},"LineNo":"10000"},"url":"https://personal2.ares.ru/index.php?language=ru&page=nav_exchange","contentType":"ApplicationJson"}}]

```

```

2019-03-21 10:27:46 DEBUG HttpImpl:127 - [Http] [POST OK] juuid:[ D84645C0-010D-4194-9CAC-7F29C8923BC5 ], [ 21.03.2019 10:27:46 ], url:[ https://personal2.ares.ru/index.php?language=ru&page=nav_exchange ], thread id: 32, thread name: pool-6-thread-1

```

```

2019-03-21 10:27:46 INFO PubTask:634 - [PubTask] [log pub success] - juuid:[ D84645C0-010D-4194-9CAC-7F29C8923BC5 ] [ 21.03.2019 10:27:46 ], eventId:[ NAV2ARES_EmplQual_INS ], sender:[ NAV ], subscriber:[ ONE ], auth:[ false ], sub handler:[ https://personal2.ares.ru/index.php?language=ru&page=nav_exchange ], thread id:[ 32 ], thread name:[ pool-6-thread-1 ]

```

-- Проверка из БД системы-отправителя

```
SELECT *
FROM [RESTORE].[dbo].[ESB Publish Log]
ORDER BY 1 DESC
```

Results		Messages									
	entry no	datetime	event_id	guid	publisher_id	subscriber_id	subscriber_handler	subscriber_store_class	error_id	details	active
1	120	21.03.2019 10:27:46	NAVZARES_EmpQual_INS	D84645C0-010D-4194-9CAC-7F29C8923BC5	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
2	119	21.03.2019 10:27:45	NAVZARES_EmpQual_INS	08429F31-E84D-46C8-A4A0-A371AB88F135	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
3	118	21.03.2019 10:27:44	NAVZARES_EmpQual_INS	A174790F-9AD4-41B3-930D-33A11691CD24	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
4	117	21.03.2019 10:27:43	NAVZARES_EmpQual_INS	BF5E0384-D1E4D-499B-98FA-773C0CA5DACC	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
5	116	21.03.2019 10:27:42	NAVZARES_EmpQual_INS	F60048F7-FD2B-4C5F-9B18-42BF2623C05	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
6	115	21.03.2019 10:27:41	NAVZARES_EmpQual_INS	8AF59F01-DC3F-4510-8755-312786EC2BA9	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
7	114	21.03.2019 10:27:40	NAVZARES_EmpQual_INS	90709842-2B87-4C6A-BF15-58D814AB35B2	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
8	113	21.03.2019 10:27:39	NAVZARES_EmpQual_INS	D99D11E4-1084-4A2E-85AA-935FD879875A	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
9	112	21.03.2019 10:27:38	NAVZARES_EmpQual_INS	CD2739E3-8627-4A06-8F5E-BEBA0BD45DA	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0
10	111	21.03.2019 10:27:38	NAVZARES_EmpQual_INS	55CEABE0-060C-41B9-94D0-D57B352835B2	NAV	ONE	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NULL	NULL	NULL	0

## 2. Проверка неудачной отправки POST-запроса

```
-- Отправляем данные в шину
USE [RESTORE]
GO
EXEC [dbo].[esb_job_Emit]
```

```
-- Проверка работоспособности сервисов:
-- должно быть по 2-е записи на одно сообщение
USE [JDB]
GO
SELECT *
FROM [jdb].[dbo].[log pub]
ORDER BY 1 DESC
```

id	publisher_id	subscriber_handler	uuid	event_id	header_value	service_id_from	service_id_to	service_role_from	service_role_to	outcoming_datetime
1	277	NAV	https://personal2.ares.ru/index.php?language=ru&page=nav_exchange8888	NAVZARES_EmpQual_INS	BusPubMsg	bridge_msk10websevcDEV	NULL	BRIDGE-SERVICE	NULL	NULL
2	276	NAV	NULL	NAVZARES_EmpQual_INS	UniMsg	front_msk10websevc2777	NULL	FRONT-SERVICE	NULL	NULL

```
-- Проверка не было ли ошибок
SELECT *
FROM [jdb].[dbo].[log pub err]
ORDER BY 1 DESC
```

Название поля	Значение	Комментарий
sender_id	NAV	Идентификатор отправителя
subscriber_id	ONE	Идентификатор подписчика
event_id	NAVZARES_EmpQual_INS	Идентификатор события
uuid	55CEABE0-060C-41B9-94D0-D57B352835B2	Уникальная метка сообщения
error_id	PubTask	Код ошибки - (для локализации) ошибка при рассылке (publishing task)
details	HttpImpl, url: < 404 Not Found >, https://personal2.ares.ru/index4.php?language=ru&page=nav_exchange8888	Текст ошибки
log_datetime	13.02.2019 03:09:07.429MSK	
is_external_error	1	Ошибка не по причине ESB - 1

```
-- Проверка через лог-файл bridge-log4.txt
```

```
2019-03-21 11:15:08 DEBUG HttpImpl:101 - [Http] [PRE POST] juuid:[ D84645C0-010D-4194-9CAC-7F29C8923BC5 ], body:[ {"clazz":"com.simple.server.domain.contract.UniMinMsg","juuid":"D84645C0-010D-4194-9CAC-7F29C8923BC5","eventid":"NAVZARES_EmpQual_INS","body":{"employeequalification":{"DocumentSeries":"A"},"DocumentNo":"993454","Description":"Охранник"},"ExpireDocumentDate":"2018-02-11"},"QualificationType":1,"QualificationCode":"ОХРАНИК"},"FromDate":"1753-01-01"},"ToDate":"1753-01-01"},"Person":"ФЛ-02272"},"DocumentDate":"2013-02-11"},"LineNo":"10000"},"url":"https://personal2.ares.ru/index4.php?language=ru&page=nav_exchange8888","contentType":"ApplicationJson"} ]
```

```
2019-03-21 11:15:08 DEBUG HttpImpl:122 - [Http] [ERR POST] juuid:[ D84645C0-010D-4194-9CAC-7F29C8923BC5 ], [ 21.03.2019 11:15:08 ], url:[ https://personal2.ares.ru/index4.php?language=ru&page=nav_exchange8888 ], err:[ 404 Not Found ],thread id: 148 , thread name: pool-24-thread-1
```

2019-03-21 11:15:08 INFO PubTask:559 - [PubTask] [log pub err] - juuid:[ D84645C0-010D-4194-9CAC-7F29C8923BC5 ] [ 21.03.2019 11:15:08 ],  
eventId:[ NAV2ARES\_EmplQual\_INS ], sender:[ NAV ], subscriber:[ ONE ], auth:[ false ], sub handler:[  
https://personal2.ares.ru/index4.php?language=ru&page=nav\_exchange8888 ], thread id:[ 148 ], thread name:[ pool-24-thread-1 ]

-- Проверка через WEBLOG EXPERT:

Last Hits													
Filters: <a href="#">New Filter</a> <a href="#">Delete All</a> Match filters: <a href="#">all</a> <a href="#">any</a> Export: <a href="#">PDF</a> <a href="#">CSV</a>													
	Date & Time	Host	Visitor ID	Method	URL	Status Code	Referrer	Browser	Operating System	Spider	Country	User Name	Virtual Domain
1	3/21/2019 11:25:21	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
2	3/21/2019 11:25:20	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
3	3/21/2019 11:25:19	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
4	3/21/2019 11:25:18	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
5	3/21/2019 11:25:17	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
6	3/21/2019 11:25:16	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
7	3/21/2019 11:25:15	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
8	3/21/2019 11:25:14	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
9	3/21/2019 11:25:13	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-
10	3/21/2019 11:25:12	127.0.0.1	17976	POST	/frontasyenc/mlpub/uni/	200	-	-	-	-	Unknown	-	-

