

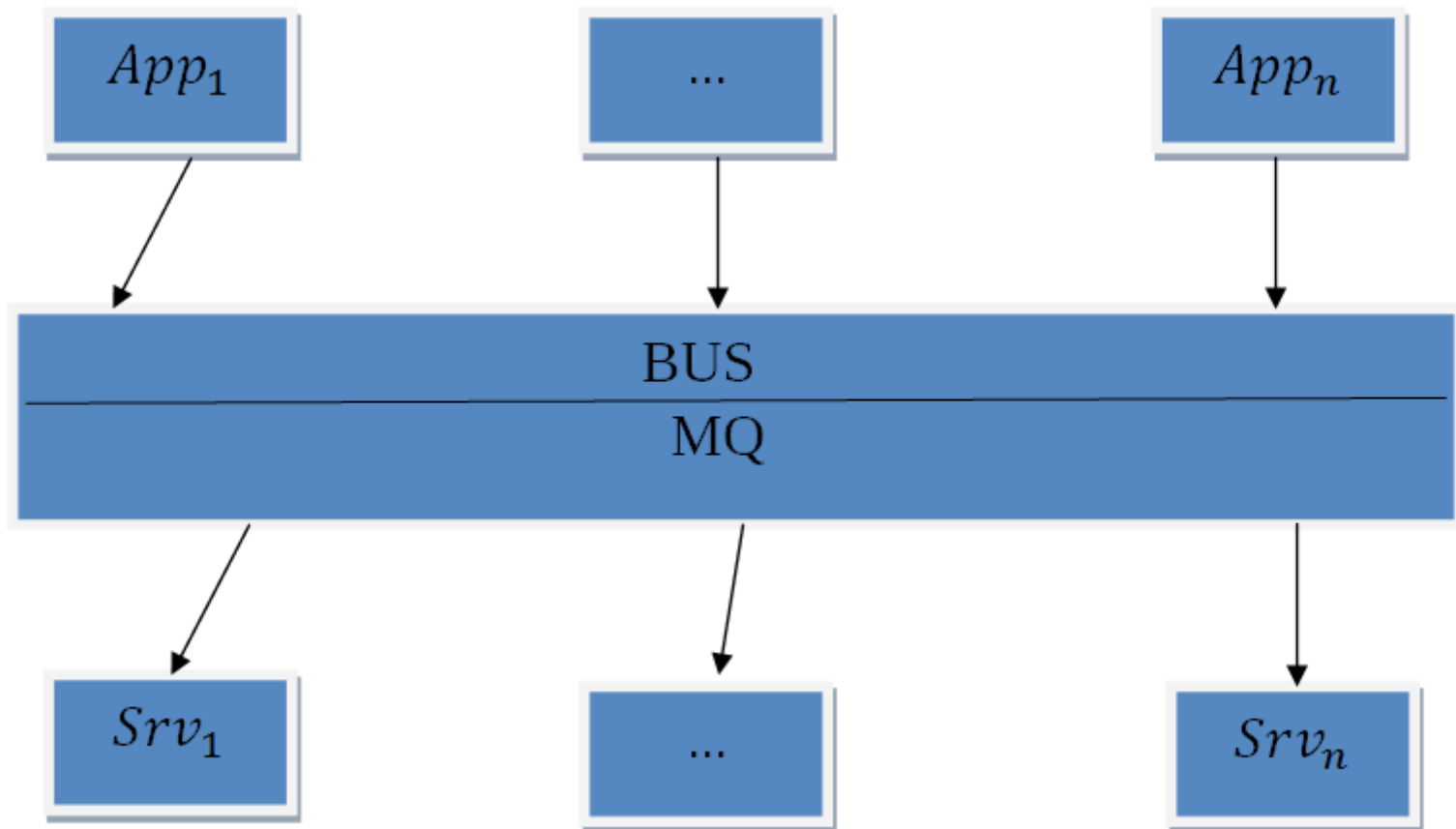
Немного о IBM Integration BUS
(ex WebSphere Business
Integration)

Передача данных

Задача.

- Асинхронно передать данные с хоста А к хосту В, транспорт выделить как промежуточный слой
- Возможность взаимодействия различных приложений
- Cross-platform

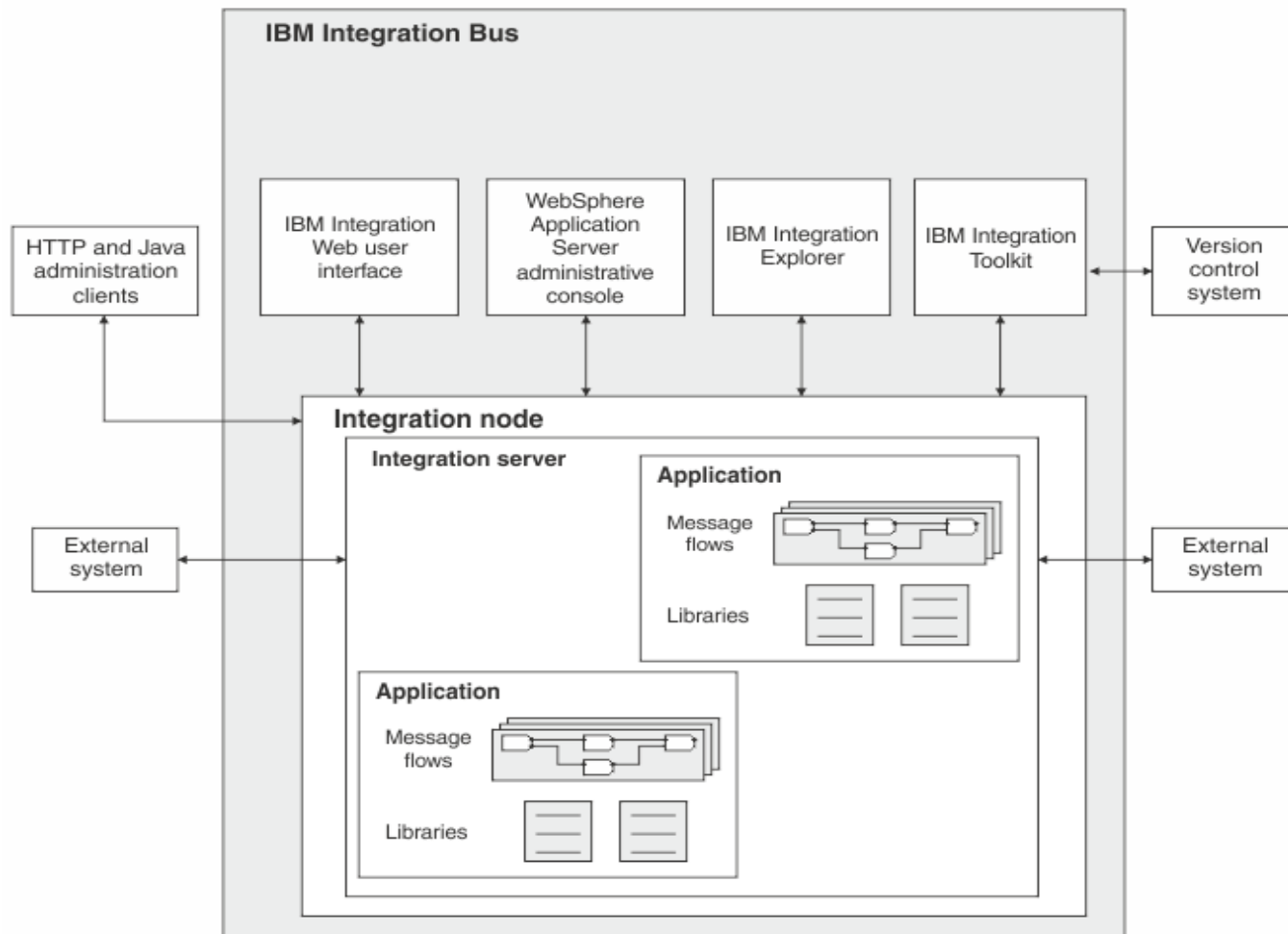
Передача данных



Передача данных

- Сетевое взаимодействие: TLI, маршрутизация
- Protocol(messages)
- Место хранения(queue)
- Целостность данных, гарантия доставки
- Производительность
- Распределение нагрузки
- Масштабируемость
- Безопасность
- Мониторинг
- Передача различных форматов данных

IBM Integration BUS



Компоненты: Integration Node

- **Integration Node** – управляет набором Execution Groups
- **Execution Group(EG или Integration Server)** – именованный контейнер для потоков сообщений, которые были назначены брокеру. Исполняются как процессы в ОС, а message flow – как потоки внутри этих процессов. В одной EG Message Flows запускаются в разных Thread pool'ах.
- **Message Flow** – поток обработки сообщения(1 msg – 1 поток, обработка данных в рамках единой транзакции, subflow)
- **Message Nodes** – узлы обработки, из которых состоит Message Flow

IBM WebSphere MQ

Структура сообщения:

- MQ Message Descriptor (MQMD): Version, MsgID, CorrelID (ID request - reply message), MsgType (datagram, request, reply, report), Persistence (flag), MsgPriority, PutTime (UTC), Format, Expiry, Encoding [1]
- Message properties [optional]: определяемый пользователем набор свойств, не является частью MQMD и payload
- Message data

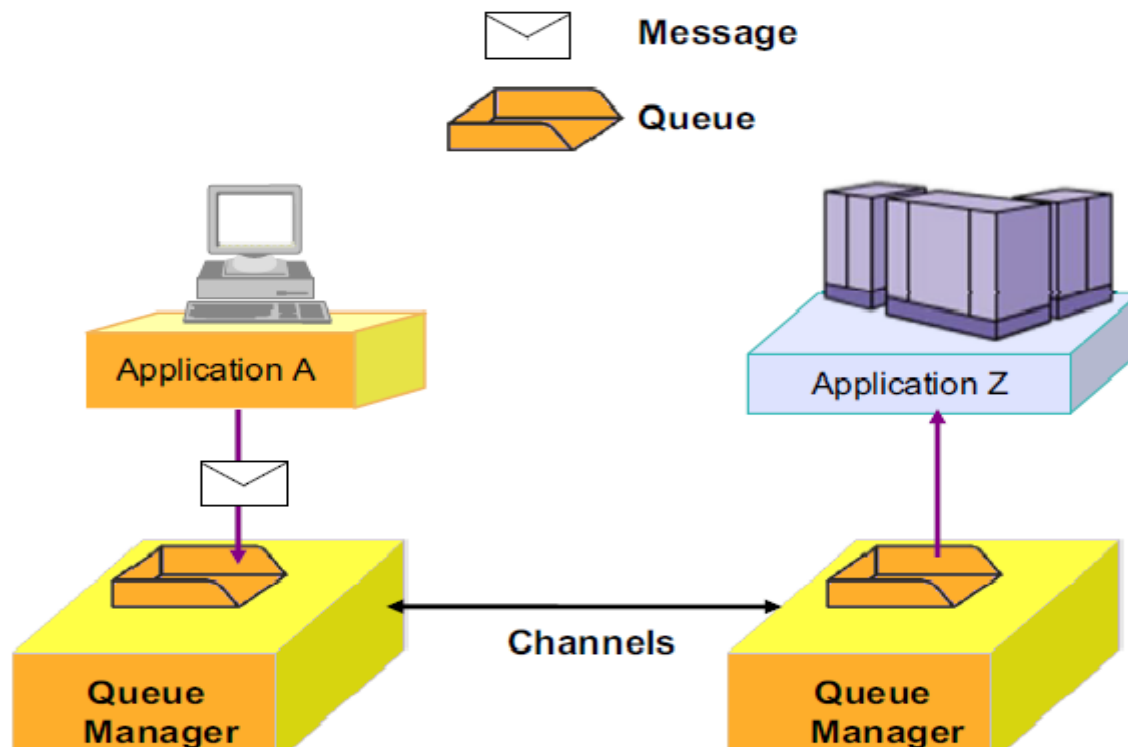
1 – https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_9.0.0/com.ibm.mq.ref.dev.doc/q097410_.htm

IBM WebSphere MQ

Queue Manager(QM) – управление очередями, обработка сообщений

QMs подключаются между собой каналами(channels).

- Каналы однонаправленные
- Triggering (автостарт каналов между QM)



Demo

- Web-service входную xml преобразует в CEF-строку, кладет в очередь
- Service читает из очереди – отправляет логи в Arcsight

Demo

The screenshot displays a web browser window with the address bar showing `http://localhost:7080/WSTest`. The browser is split into two panes. The left pane, titled "Request 1", shows the raw XML of an HTTP request. The right pane shows the raw XML of the HTTP response.

Request XML:

```
<?xml version='1.0' encoding='utf-8'>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsc="http://WSTest10.EQ.CS.ws.alfabank.ru">
  <soapenv:Header/>
  <soapenv:Body>
    <wsc:WSTestInParam>
      <Version>0</Version>
      <DeviceVendor>WSArc</DeviceVendor>
      <DeviceProduct>TestSrv3</DeviceProduct>
      <DeviceVersion>a6acfd8db47a8bc4c6a64e2a9036d017b6a9c6c</DeviceVersion>
      <DeviceEventClassID>bd9ce70-4125-11e7-b9a9-2dc253df02e3</DeviceEventClassID>
      <Name>PCMB3</Name>
      <Severity>3</Severity>
      <!--Optional-->
      <act>logout</act>
      <!--Optional-->
      <app>HTTPS</app>
      <!--Optional-->
      <msg>Message3:\nSend message. Act logout</msg>
      <!--Optional-->
      <deviceExternalId>414d51206d71202020202012476</deviceExternalId>
      <shost>A104994</shost>
      <src>172.25.21.84</src>
      <dhost>A104993</dhost>
      <duser>u_021nt@alfabank.ru</duser>
      <!--Optional-->
      <dmac>00:0D:60:AF:1B:61</dmac>
      <!--Optional-->
      <dvc>172.25.20.157</dvc>
      <!--Optional-->
      <dvchost>172.25.20.157</dvchost>
      <!--Optional-->
      <suid>2</suid>
      <!--Optional-->
      <suser>u_m0tft</suser>
    </wsc:WSTestInParam>
  </soapenv:Body>
</soapenv:Envelope>
```

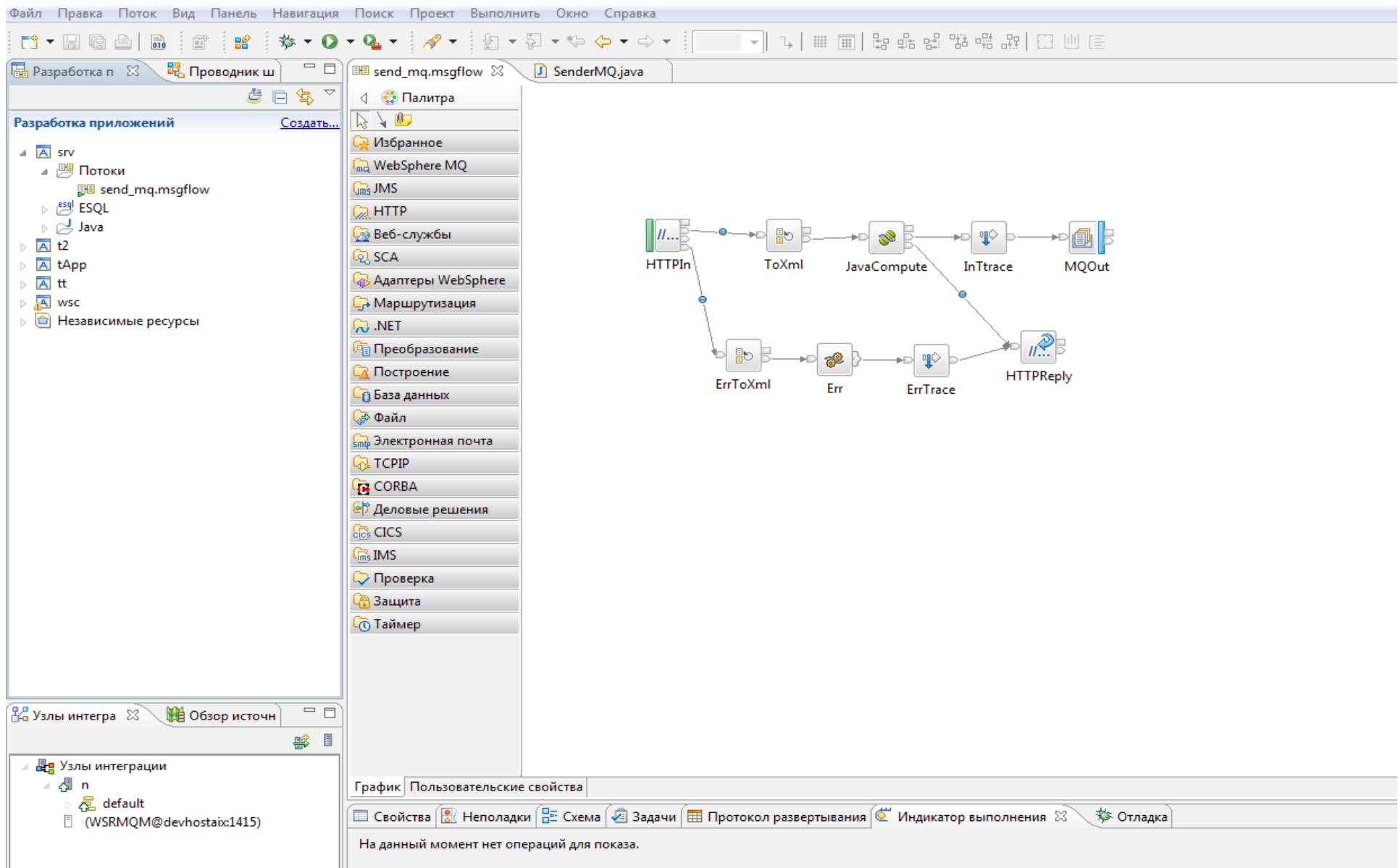
Response XML:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: text/xml; charset=utf-8
Content-Length: 273
Date: Fri, 12 Jan 2018 10:53:18 GMT

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsm="http://WSTest10.EQ.CS.ws.alfabank.ru"> <sc
```

Demo

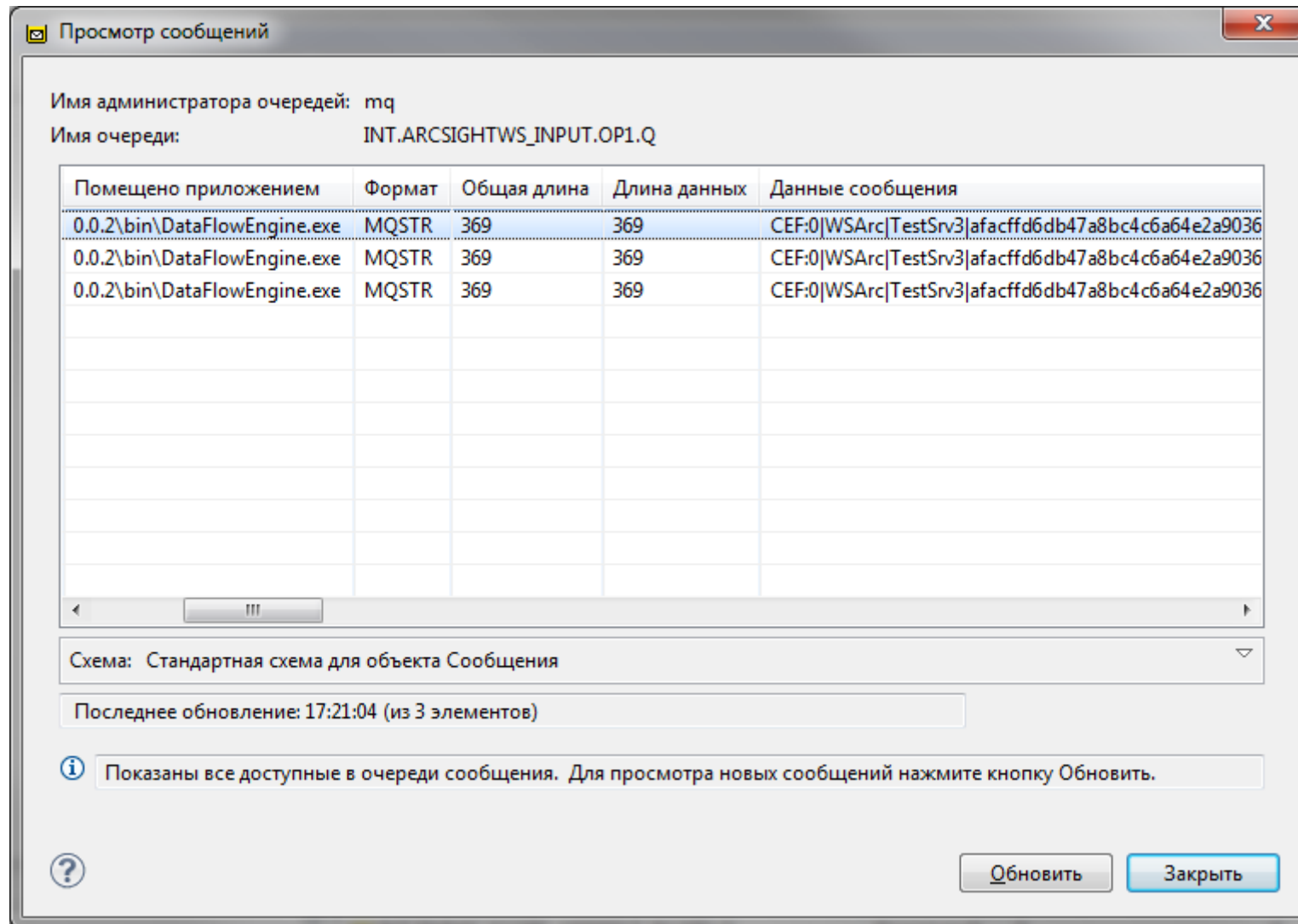
- Web-service



Demo

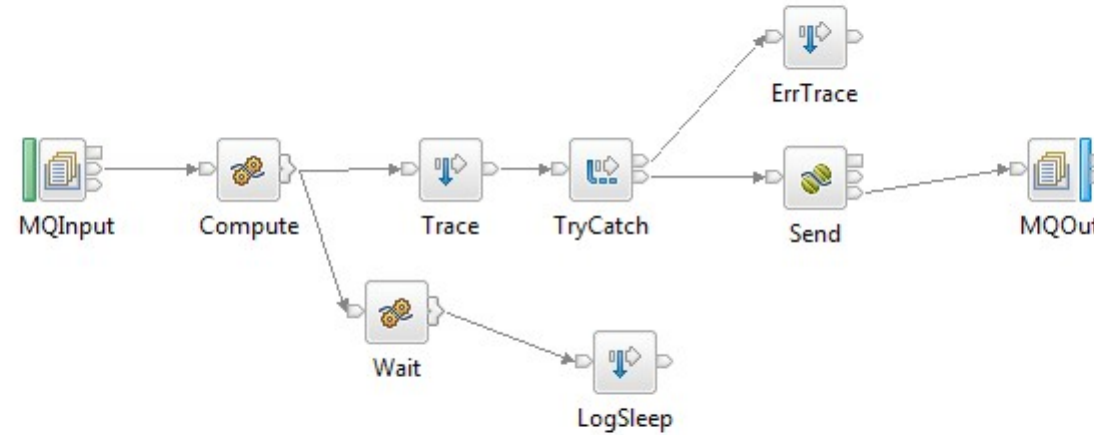
```
138
139 public void evaluate(MbMessageAssembly inAssembly) throws MbException {
140     MbOutputTerminal out = getOutputTerminal("out");
141     MbMessage inMessage = inAssembly.getMessage();
142
143     // create new empty message
144     MbMessage outMessage = new MbMessage();
145     MbMessageAssembly outAssembly = new MbMessageAssembly(inAssembly, outMessage);
146     try {
147         // Add user code below
148
149         MbElement inRoot = inMessage.getRootElement();
150         MbElement body = inRoot.getLastChild().getFirstChild().getLastChild();
151         MbElement cur = body.getFirstChild();
152         String str = cefString(cur);
153         // fill message
154         MbElement outRoot = outMessage.getRootElement();
155         MbElement mqmd = outRoot.createElementAsLastChild(MbMQMD.PARSER_NAME);
156         mqmd.createElementAsFirstChild(MbElement.TYPE_NAME_VALUE, "Version", VERSION_2);
157         mqmd.createElementAsLastChild(MbElement.TYPE_NAME_VALUE, "Format", FORMAT_MQSTR);
158         mqmd.createElementAsLastChild(MbElement.TYPE_NAME_VALUE, "CodedCharSetId", CODEDCHARSET_UTF8);
159
160         MbElement mqrh2 = mqmd.createElementAsLastChild(FORMAT_RFH2);
161         mqrh2.createElementAsFirstChild(MbElement.TYPE_NAME_VALUE, "Version", VERSION_2);
162         mqrh2.createElementAsLastChild(MbElement.TYPE_NAME_VALUE, "Format", FORMAT_MQSTR);
163         mqrh2.createElementAsLastChild(MbElement.TYPE_NAME_VALUE, "Encoding", ENCODING);
164         mqrh2.createElementAsLastChild(MbElement.TYPE_NAME_VALUE, "CodedCharSetId", CODEDCHARSET_UTF8);
165         mqrh2.createElementAsLastChild(MbElement.TYPE_NAME_VALUE, "usr", String.valueOf(appName));
166
167         MbElement outBlob = outRoot.createElementAsLastChild(MbBLOB.PARSER_NAME);
168         outBlob.createElementAsLastChild(MbElement.TYPE_NAME_VALUE, MbBLOB.ROOT_ELEMENT_NAME, str.getBytes(StandardCharsets.UTF_8));
169
170         out.propagate(outAssembly);
171
172         createResponse(inAssembly);
173
174         // End of user code
175     } catch (MbException e) {
176         throw e;
177     }
```

Demo



Demo

- Service отправки в Arcsight (msgflow)



Demo

Service отправки в Arcsight использован:

- Syslog Java Client [1] – библиотека для отправки сообщений на Syslog Server (Arcsight):
- TcpSyslogMessageSender [RFC 3164 - The BSD syslog Protocol]

Спасибо