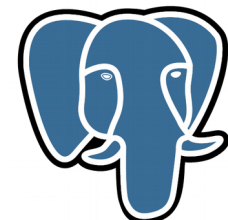
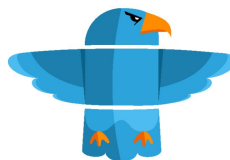
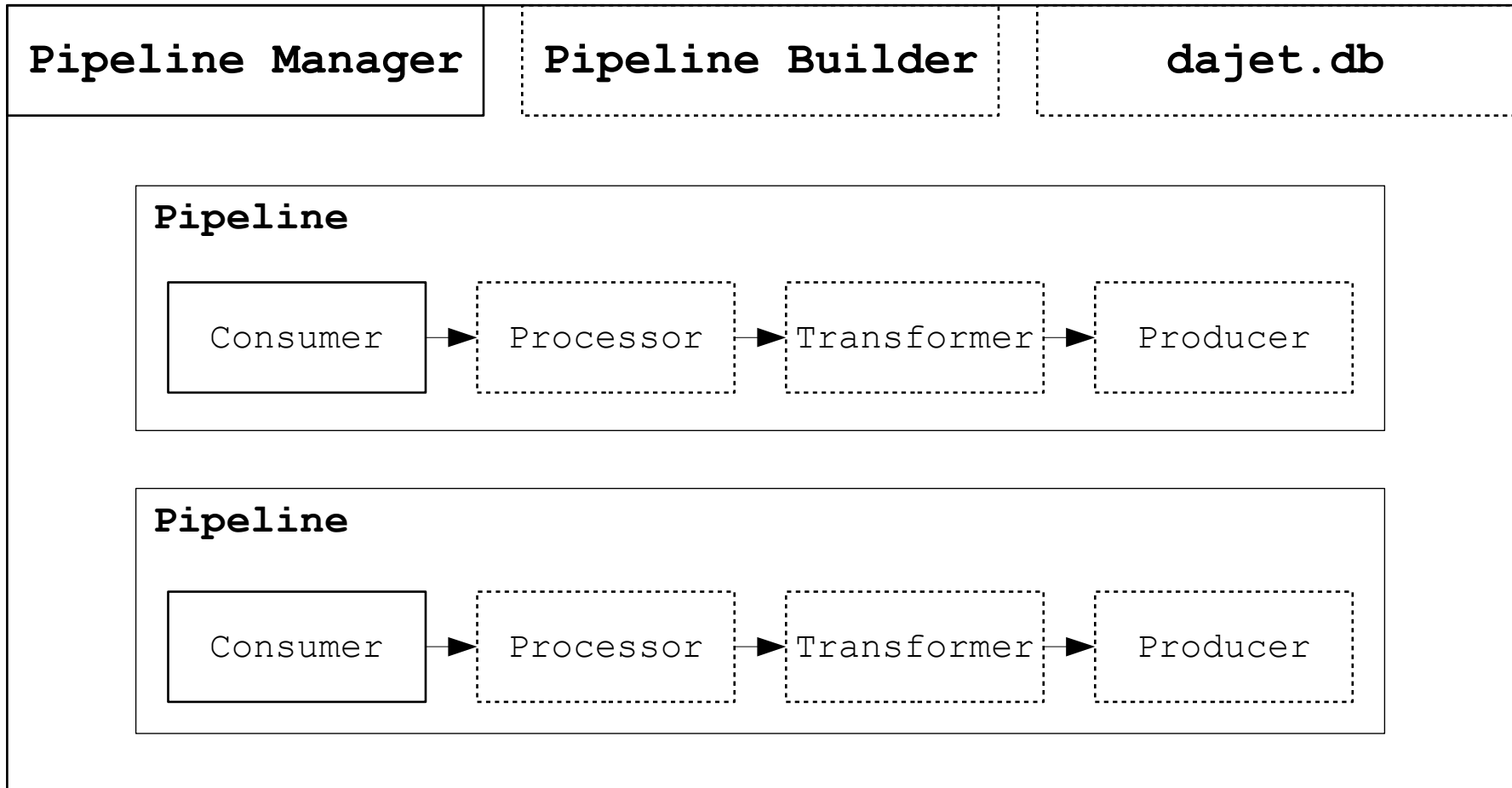
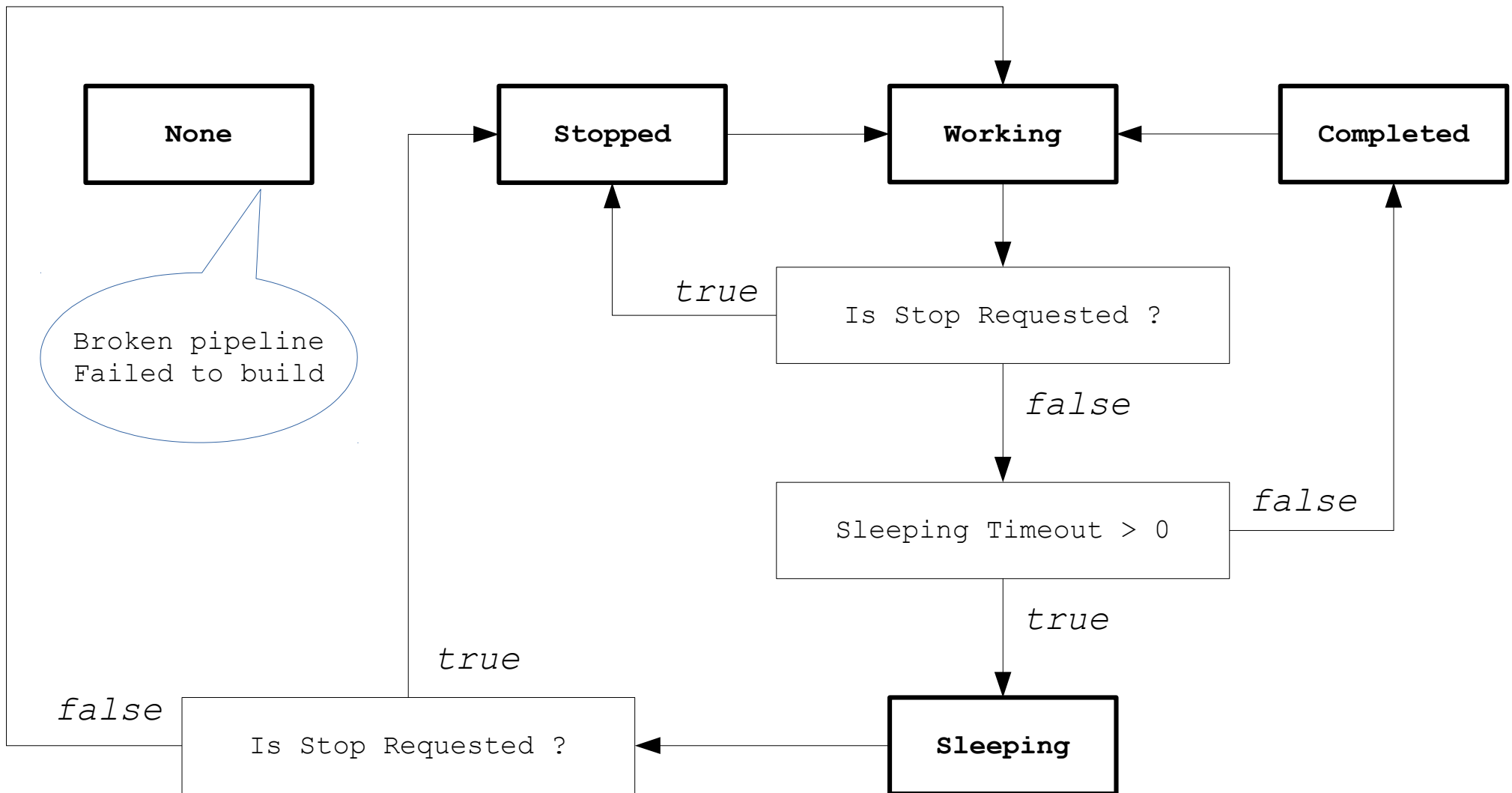


DaJet Flow Pipelines

(конвейеры обработки данных)



Pipeline state machine

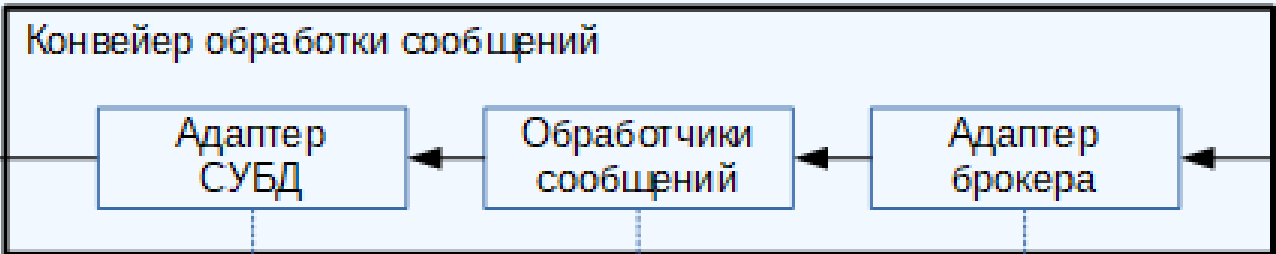
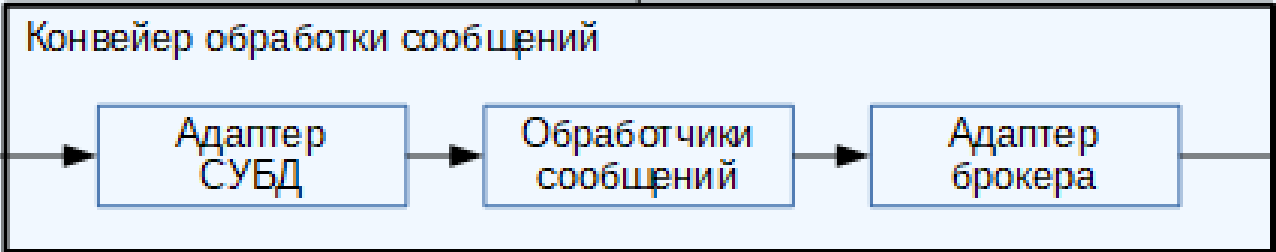


DaJet

Настройки конвейеров

Сборщик конвейеров

Доставка и обработка сообщений из источника в приёмник выполняется конвейерами обработки сообщений



СУБД

Брокер сообщений

Выполняет подключение к СУБД, отвечает за чтение и запись сообщений

Выполняет трансформацию сообщений из одного формата в другой

Выполняет подключение к брокеру, отвечает за чтение и запись сообщений

DaJet Flow Pipelines

<https://github.com/zhichkin/dajet>

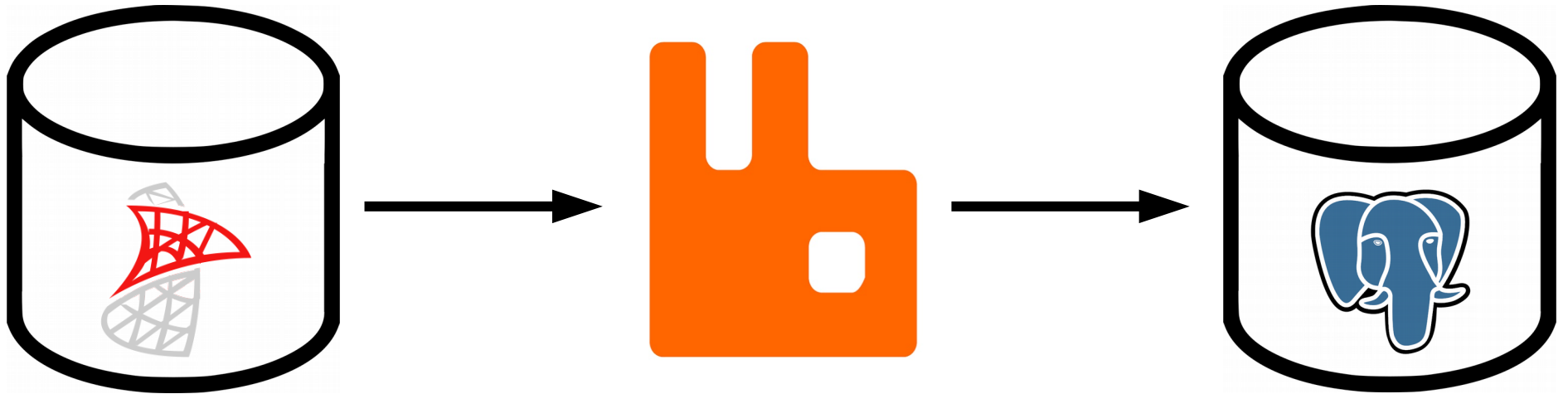
Интеграция Rabbit MQ



<https://zhichkin.github.io>

План интеграции

SQL Server → RabbitMQ → PostgreSQL



https://t.me/dajet_studio

План конвейеров

Pipeline: mssql > rabbit



IDataRecord → Payload → Message



QuerySource
SqlServer

Transformer
IDataRecord
Payload

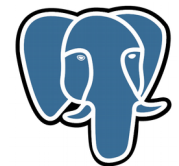
Transformer
Payload
Message

Producer
RabbitMQ

Pipeline: rabbit > pgsql



Message → IDataRecord



Consumer
RabbitMQ

Transformer
Message
IDataRecord

Producer
PostgreSql

SQL Server

```
CREATE TABLE Номенклатура
```

```
(  
    Ссылка          uniqueidentifier NOT NULL,  
    Код             nvarchar(12)      NOT NULL,  
    Наименование   nvarchar(100)     NOT NULL,  
    CONSTRAINT PK_Номенклатура PRIMARY KEY CLUSTERED (Ссылка ASC)  
)
```

PostgreSQL

```
CREATE TABLE public."Номенклатура"
```

```
(  
    "Ссылка"        uuid              NOT NULL,  
    "Код"           character varying(12) NOT NULL,  
    "Наименование" character varying(100) NOT NULL,  
    CONSTRAINT "Номенклатура_pkey" PRIMARY KEY ("Ссылка")  
)  
WITH (OIDS = FALSE);
```

SQL Server (исходящие данные)

```
SELECT Ссылка, Код, Наименование FROM Номенклатура
```

RabbitMQ (входящее сообщение)

▼ Add a new exchange

Virtual host:

test ▼

Name:

from-mssql

*

Type:

topic ▼

Durability:

Durable ▼

RabbitMQ (исходящее сообщение)

▼ Add a new queue

Virtual host:

Type:

Name: *

Durability:

Auto delete:

From	Routing key	Arguments	
(Default exchange binding)			
<input type="text" value="from-mssql"/>	#		<input type="button" value="Unbind"/>

PostgreSQL (входящие данные)

```
INSERT INTO Номенклатура (Ссылка, Код, Наименование)  
VALUES (@Ссылка, @Код, @Наименование)
```

Pipeline: mssql > rabbit



IDataRecord



Payload



Message



Pipeline

mssql > rabbit

Activation

Manual

▼ Pipeline Options

▼ DaJet.Flow.SqlServer.QuerySource < System.Data.IDataRecord >

▼ DaJet.Flow.RecordToPayloadTransformer < DaJet.Flow.Payload >

▼ DaJet.Flow.RabbitMQ.PayloadToMessageTransformer < DaJet.Flow.RabbitMQ.Message >

▼ DaJet.Flow.RabbitMQ.Producer < DaJet.Flow.RabbitMQ.Message >

Exchange

from-mssql

Routing Key

Redelivered

Properties

o

app_id: mssql

type: Справочник.Номенклатура

priority: 0

delivery_mode: 2

content_encoding: UTF-8

content_type: application/json

Payload

151 bytes

Encoding: string

{

"Ссылка": "059ebf57-e4b1-43fb-b89c-484409ce5b74",

"Код": "ЦУ-00000039",

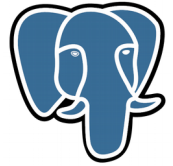
"Наименование": "1С:Упрощенка 8"

}

Pipeline: rabbit > pgsql



Message▶ IDataRecord



Pipeline
rabbit > pgsql







Activation
Manual

▼ Pipeline Options

▼ **DaJet.Flow.RabbitMQ.Consumer < DaJet.Flow.RabbitMQ.Message >**

▼ **DaJet.Flow.RabbitMQ.MessageToRecordTransformer < System.Data.IDataRecord >**

▼ **DaJet.Flow.PostgreSql.Producer < System.Data.IDataRecord >**

 	Name	Activation	State	Start	Finish	Status
 	mssql > rabbit	Manual	Idle	28-04-2023 15:10:24	28-04-2023 15:10:24	Processed 3 records
 	rabbit > pgsql	Manual	Idle	28-04-2023 15:08:36	28-04-2023 15:08:49	Consumed 3 messages in 10 seconds.

SELECT Ссылка, Код, Наименование **FROM** Номенклатура

```
{  
  "Ссылка": "059ebf57-e4b1-43fb-b89c-484409ce5b74",  
  "Код": "ЦУ-00000039",  
  "Наименование": "1С:Упрощенка 8"  
}
```

INSERT INTO Номенклатура (Ссылка, Код, Наименование)
VALUES (@Ссылка, @Код, @Наименование)