

Technical Overview

Challenge #1:

JSON definition:

We have created the json structure adding new array fields to enrich the reporting part.

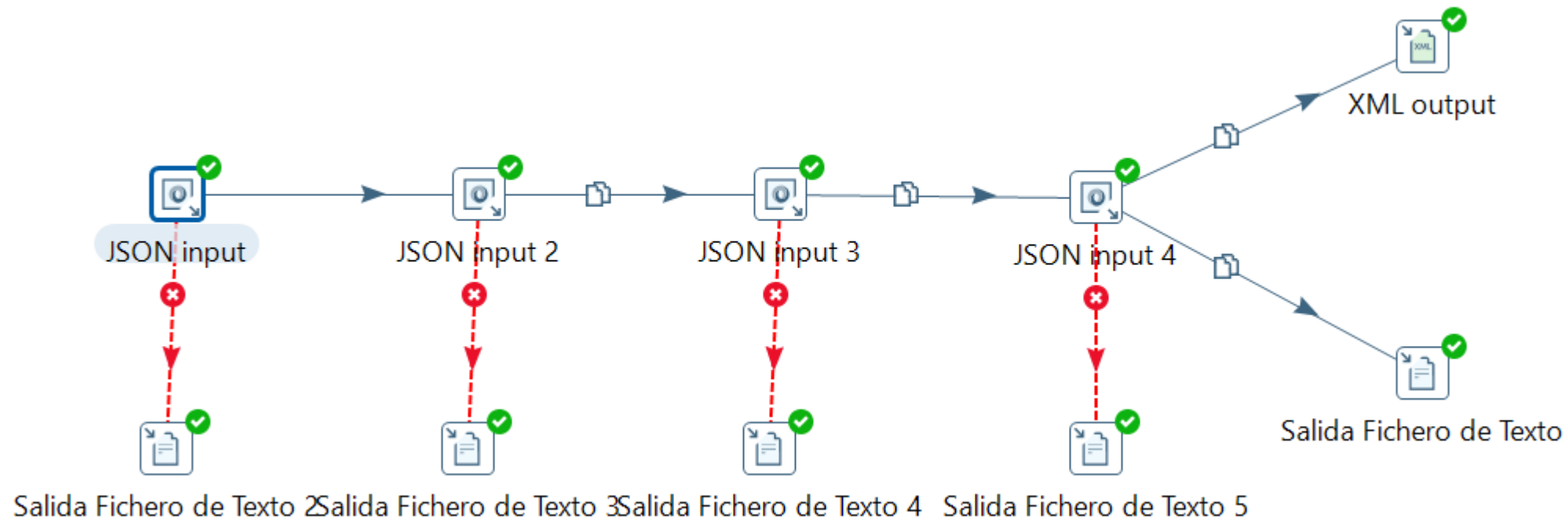
```
1.  {
2.    "userid": "5ccefaba7231cea552605682",
3.    "user_name": "taylorkline@kidstock.com",
4.    "event": "product_view",
5.    "product": [
6.      {
7.        "category": "Tennis",
8.        "product_model": 753,
9.        "product_sex": "Women"
10.     }
11.   ],
12.   "timestamp": "2018-06-01 03:01:42",
13.   "device_info": [
14.     {
15.       "vendor": "Other",
16.       "device_model": 312
17.     }
18.   ],
19.   "gps": [
20.     {
21.       "latitude": -67.525498,
22.       "longitude": 92.252578
23.     }
24.   ],
25.   "applicattion": "Adidas Fitness"
26. }
```

Technical Overview

Challenge #1:

Normalization

Now, we have to normalize the json and generate de xml file. We Will generate data automatically using json-generator



Technical Overview

Challenge #1:

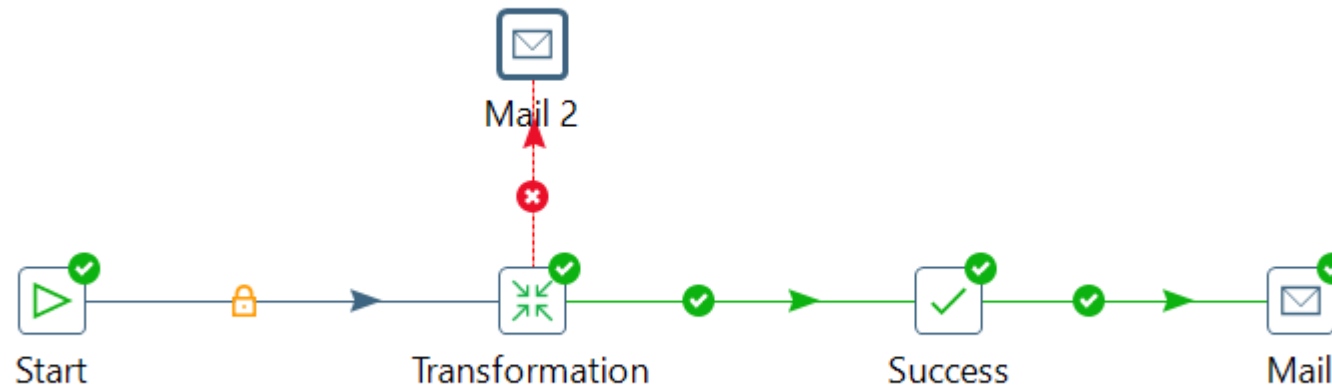
Batch Process:

This transformation has been created for testing purposes.

For production environment you will have to implement a batch process thinking in incremental data. You have to define the time you will execute your process and how add this information to the model. For example we can define a time directory folder per days, and execute the process from the day before and add this information as incremental data to the csv.

RollBack:

For rollingback, you have all the errors saved into the errors folder. So you can try to re-apply the files in the error folders.



Technical Overview

Challenge #1:

Reporting

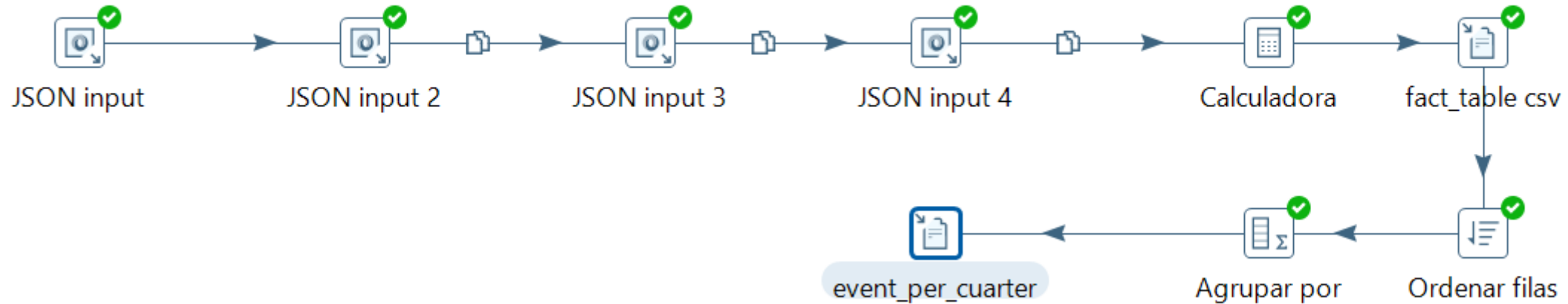
Now we have to use normalized data in a datawarehouse csv files.

First we will Split our csv timestamp in YEAR/MONTH/DAY/Quarter in our fact table definition.

With this information we can analyze our business in time line.

This data Will be our fact_table.

We can define new kpis to ease our analysis. For example we Will define a KPI like events per quarter.



Technical Overview

Challenge #1:

Reporting

I think the best option is to generate the csv with the fact_table and use a database for generating the kpis. You will have best performance. If you decide to use this option you have to define a PK in your table. You can use the id (userid) and the timestamp. Using these columns you will prevent loading duplicate rows. Related to this, it would be a good idea to create indexes and partitions by range in your reporting table schema.

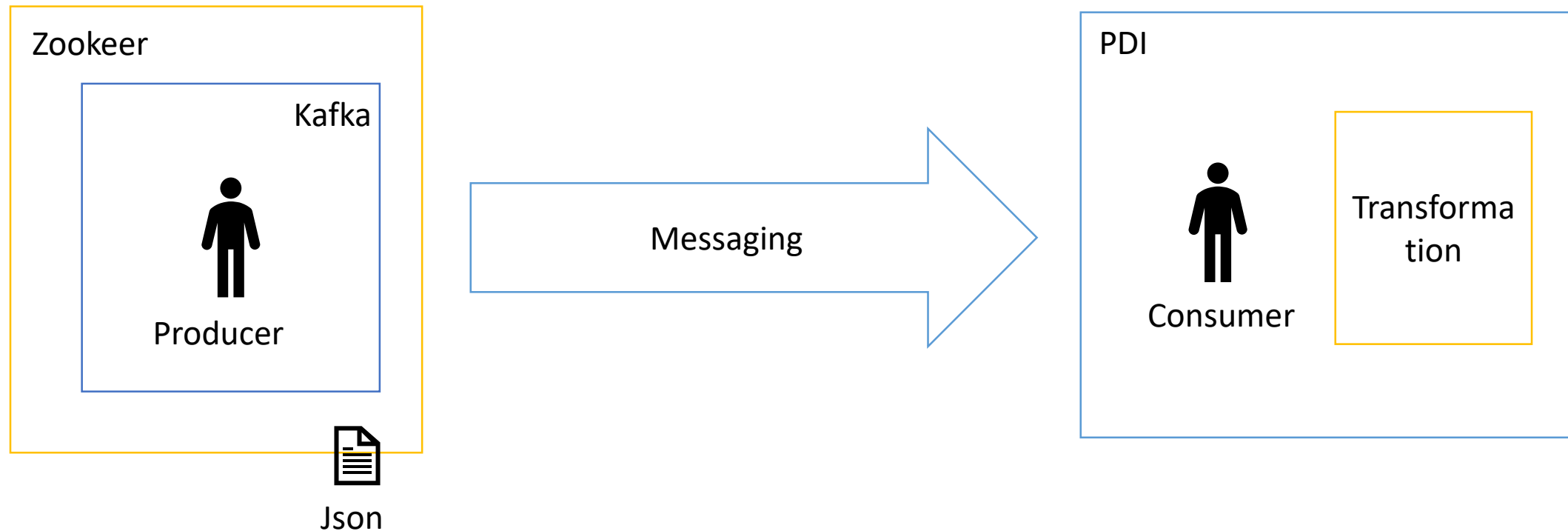
Technical Overview

Challenge #2:

API

We Will use Apache Kafka for creating and streaming API.

We need to install Zookeeper and Kafka to create the producer. Then when the producer are working we can start our consumers from PDI and receiving the data for transforming.



Technical Overview

Challenge #2:

API

We Will use Apache Kafka for creating and streaming API.

We need to install Zookeeper and Kafka to create the producer. Then when the producer are working we can start our consumers from PDI and receiving the data for transforming.

