Data streams and fraud detection

Maciej Próchniak

Data streams and fraud detection

The telecom business is facing a serious hazard growing as fast as the industry itself. Communications Fraud Control Association (CFCA) stated that telecom fraud represented nearly $30 billion globally in 2017. Another problem is to stay on top of the game with effective anti-fraud technologies.

Types of fraud

There are several types of fraud to be aware of:

- **Premium usage fraud.** In this case, the goal is to detect excessive premium usage coming from one phone, potentially stolen or bought using false documents. This can be done by calculating the total amount of charges from this account during the previous few hours.
- **SMS spam.** Sometimes regular phone accounts are misused to send spam messages, which stands against the service contract. The visible symptom of such fraud is a suspicious amount of messages coming from one account, e.g. 1000 in 4 hours. This case is a little more complex because of the risk of many false positives. A different approach would be useful in this case – calculating the unique recipients of such messages, not the total number of messages.
- **SIM cloning.** It's possible to hack SIM cards and have many physical cards connected to the network as one number. In this case, one of the ways of detecting such fraud is to track the location of the caller. If the location data results show that the same caller moved from Berlin to Paris in 10 minutes, this clearly indicates fraudulent behaviour.
- **Termination frauds of various kinds.** One of the most common is SIMBox, in which case the fraudster illegally sets up a kind of VoIP to bypass high interconnect rates. This type of fraud can be rather hard to detect. The key is to separate normal and suspicious usage and relate it to a specific location of installed appliances.

In each case, it's crucial to detect suspicious activity as fast as possible and decide on appropriate actions.

When dealing with certain types of frauds, such as SMS flooding or premium services abuses, each minute of unstopped operation can cause a significant loss of money. In such circumstances, proactive automated actions such as blocking the service might be necessary. In other cases, lowering the credit limit or alerting the revenue assurance team can be enough to tackle the problem.

In general, mobile operators have two ways of dealing with fraud:

- **Simple credit limits in their billing systems.** These are enforced immediately in real time. However, billing systems are not easily changed.
- **More elaborate models and rules performed on data warehouses.** Here, rules can be much more flexible, but they usually operate with a significant delay – an hour can be considered fast in this context.

Both methods are not sufficient. Revenue assurance teams monitor the suspicious behaviour of customers continuously and want to be able to prototype and check various types of fraud detection algorithms.

For example, while standard limits for SMS amounts may be 100 per hour, an analyst detects that many frauds take place in a particular country. They want to lower the limit of messages in this particular area for a week to see if it helps.

This type of change should not require development – analysts should be able to configure the system themselves, deploy new rules and monitor their performance. Professional work also requires testing the set rules beforehand since blocking regular clients should be avoided at all costs.

Is it possible to combine all the requirements above? Let us assure you – it is.

TouK has successfully delivered modern fraud detection frameworks for one of the largest Polish mobile operators, consisting of two components:

- **Apache Flink** – leading stream processing engine
- **TouK Nussknacker** – a unique GUI for analysts and business people to design, test and monitor processes deployed on Flink

Let’s take a closer look at the components and find out how they work.

Apache Flink

Apache Flink is one of the leading stream processing platforms. Its main features are:

- **Scalability and throughput** – it can process 10k events per second per one core, which means that even a small cluster (2-3 machines) can handle 100k of events per second
- **Very low latency** – our experiences show that typical event processing latency is less

than 100ms

- **Ability to easily define and efficiently process various aggregations** – e.g. time windows.

While not as widely known as Apache Spark, Flink is by no means unforeseeable or immature. It has many complex deployments worldwide, including such industry giants as Alibaba, Netflix or Uber.

Flink is also used in the financial sector with ING and Capital One are widely recognisable examples, and in telecom companies, such as by Bouygues Telecom, Play and Ericsson. You can read about other use cases on the Apache’s Flink product page.

Nussknacker

One of the most significant challenges in introducing stream processing platforms is giving analysts and business people (e.g. Revenue Assurance team) the ability to change the logic, test new behaviour, monitor and analyse performance.

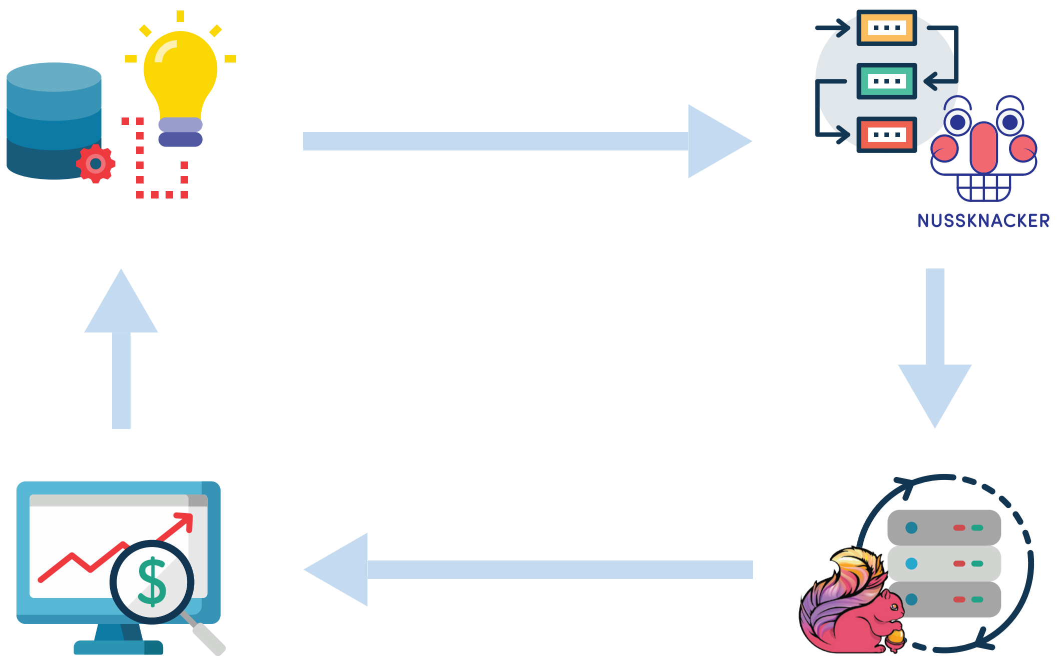
These features are especially important for companies that do not have an internal software development team. In such cases, each change that requires development from an external vendor consumes time and increases costs.

TouK Nussknacker is a unique tool that gives analysts access to the power of streaming with Apache Flink. It has a user-friendly interface built with the following general principles:

- **Business power users can author and configure processes themselves** - with no programming skills required – minimal knowledge about SQL will be enough
- **Testing and prototyping are easy** – the users have to feel empowered to try out their ideas, and they have to be able to deploy and control the process in the production environment.

Please note that **some coding and configuration may still be needed to develop specific models and integrations** unique to a particular deployment.

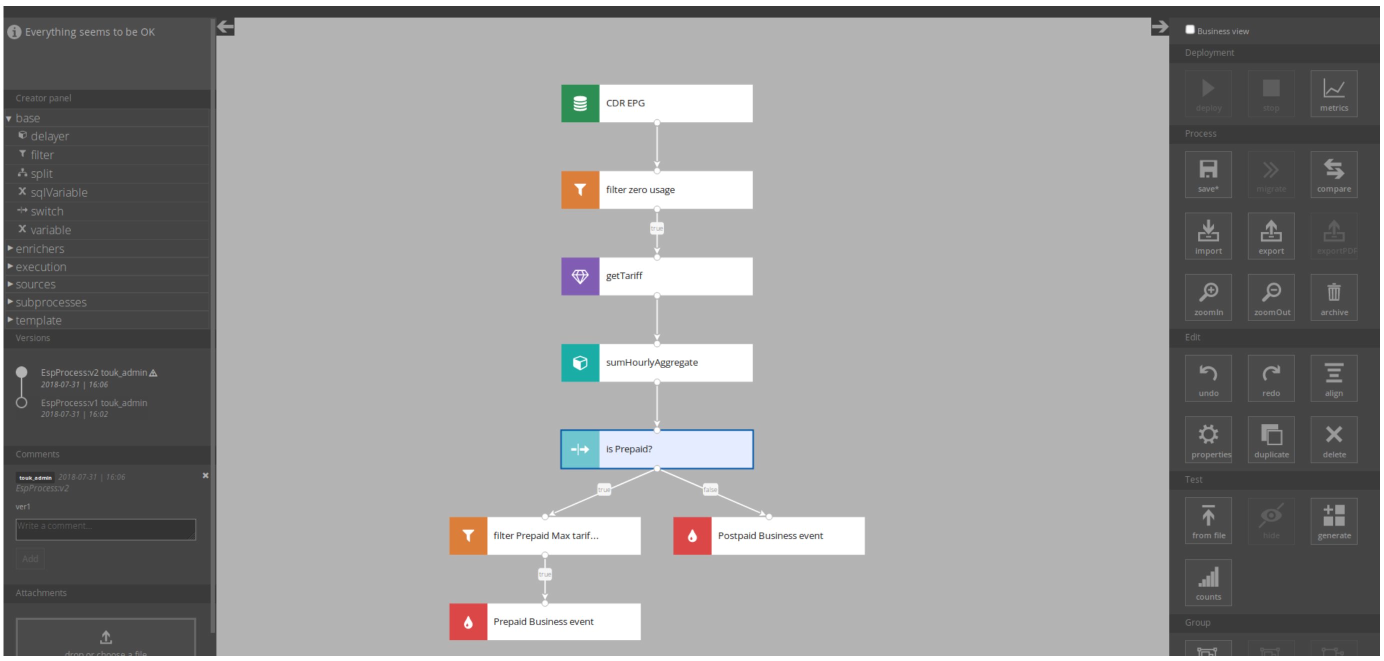
What we want to enable is closing the feedback loop shown on the diagram below.



We want to enable analysts to perform all steps:

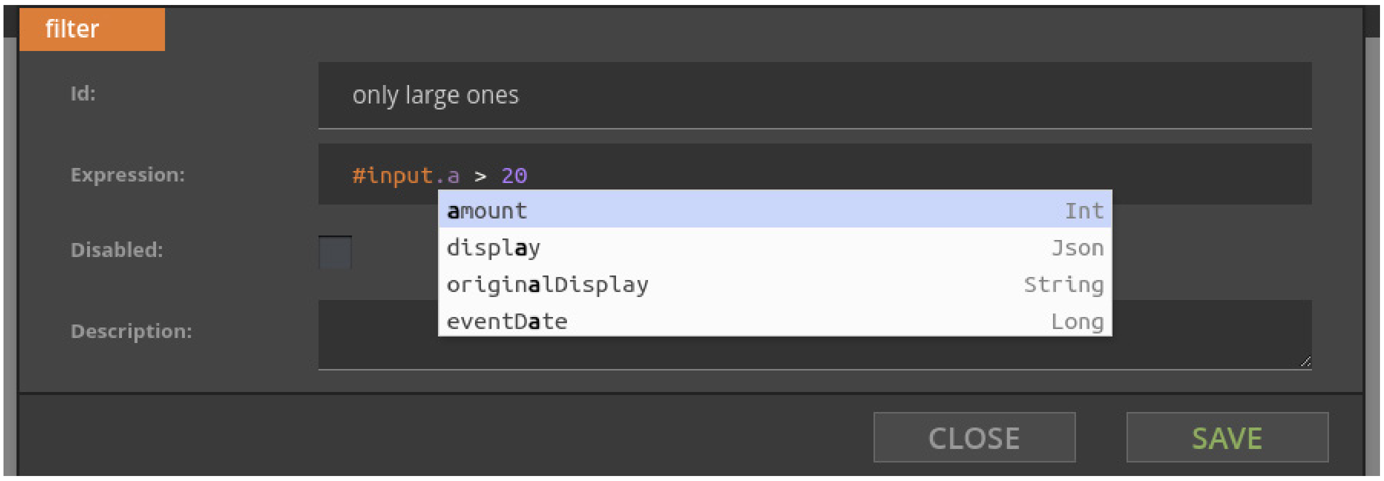
1. Design required events
2. Prototype and test the logic in Nussknacker sandbox
3. Run on real data on a high-performance Flink cluster
4. Analyse outcomes

Below you can see a sample process in Nussknacker:



The diagram defining data flow consists of some generic parts, like filters, variable definitions, aggregate definitions, and those specific for a given use case or deployment, e.g. enriching events with additional customer data or defining specific actions. In anti-fraud scenarios, these would be violation alerts or blocking accounts.

Let’s have a look at a sample filter and aggregate definitions:


filter

Id: only large ones

Expression: `#input.a > 20`

amount	Int
display	Json
originalDisplay	String
eventDate	Long

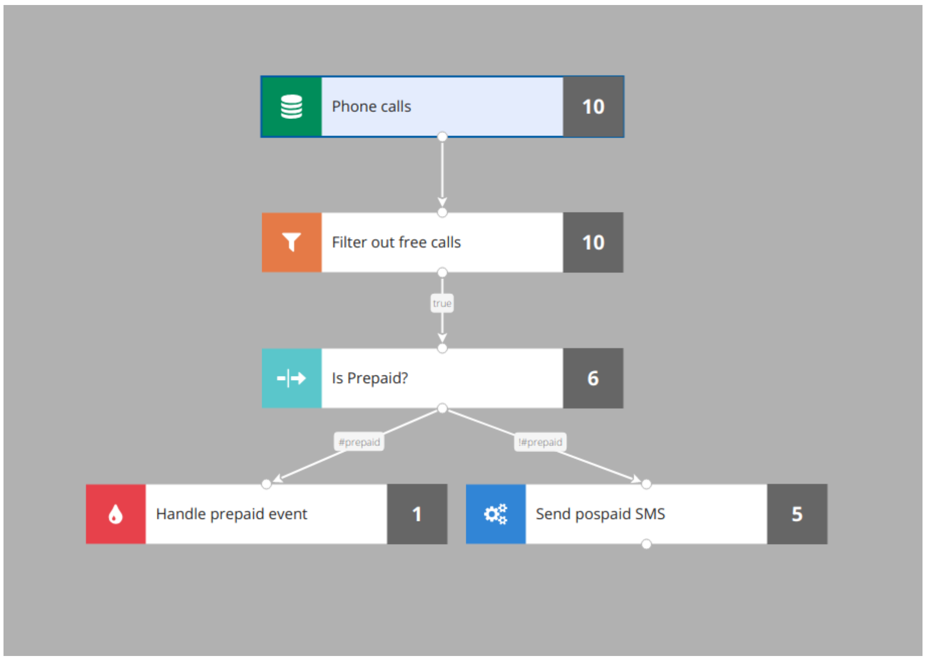
Disabled:

Description:

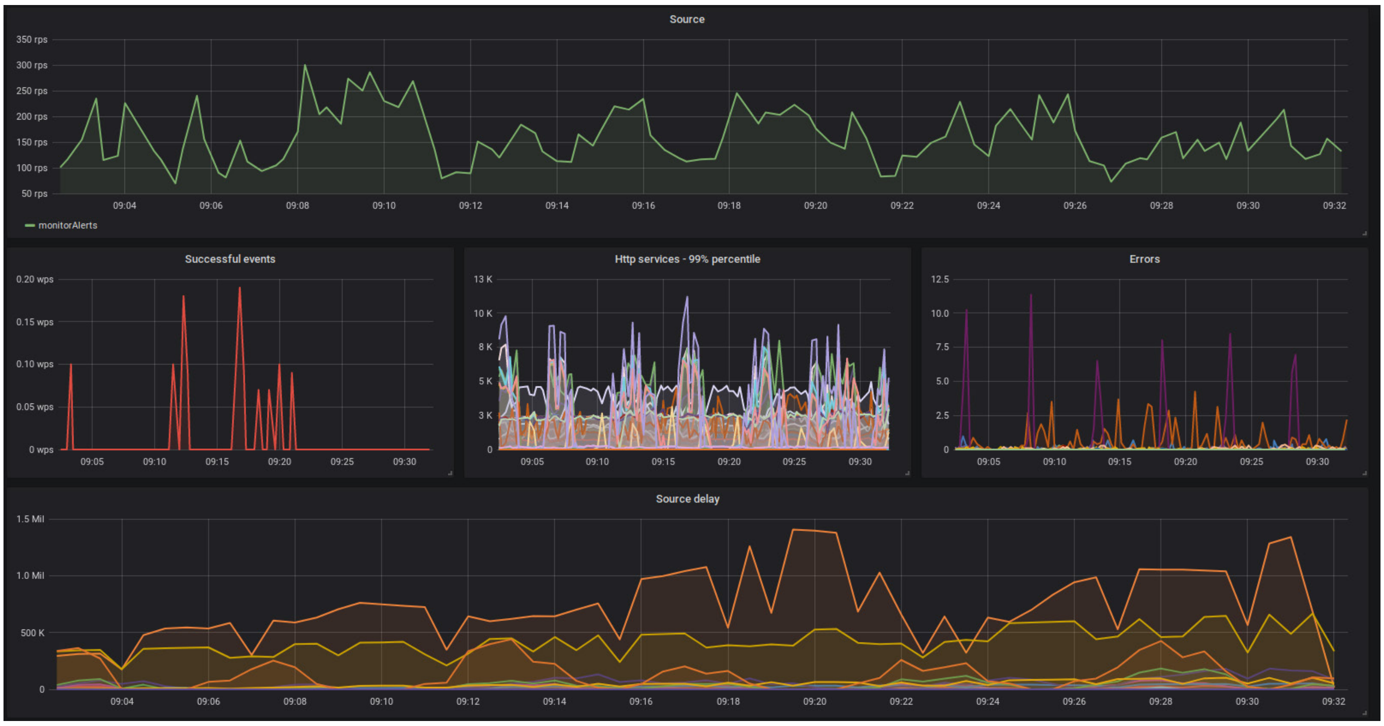
The rules are defined with a simple expression language, in many ways similar to SQL. It’s expressive enough to handle most cases with no coding skills required.

Designing a diagram is just the beginning. It’s even more important to test it properly. Nussknacker can help with it in two ways:

- **Sandbox testing**. With easy-to-prepare test cases based on real data, the user can run a process in a sandbox environment and observe how it behaves.
- **Acceptance testing**. It’s done in a dedicated environment, which has all relevant data but does not trigger real actions. The architecture of the platform (especially Apache Kafka features) simplifies cloning production data while Nussknacker monitoring solutions help to determine whether the process is ready for production. Then, migration is just one click away.



After deploying the process to the production environment, monitoring is crucial. We observe how the process behaves in two ways. The first one is monitoring based on metrics. Each process is displayed on an individual dashboard:



where you can see:

- **Throughput** - how many events per second are processed
- **Latency** - how long does it take from event to processing
- **Errors**, detected frauds and so on.

This dashboard gives a general view whether the process performs well and indicates if any problems occurred.

To go deeper into details, ElasticSearch and Kibana come in handy. We analyze interesting events and perform basic analysis and search to find out if all conditions are fine and check if the analysed case was real fraud only false positives.

Banking

The use case described above comes from the telecom industry but similar setup can be used in finance and other areas. Further development is also possible, such as machine learning models to filter more cases. Nussknacker is a perfect match for use cases which involve business rules that change dynamically, like fraud detection with Flink at

ING Bank: https://berlin-2017.flink-forward.org/kb_sessions/fast-data-at-ing-building-a-streaming-data-platform-with-flink-and-kafka/.

Other possible use cases

The capabilities of Flink and Nussknacker can be also applied for other purposes, like real-time Quality of Service monitoring. Bouygues Telecom uses Flink to detect connection and quality of service problems:

https://2016.flink-forward.org/kb_sessions/a-brief-history-of-time-with-apache-flink-real-time-monitoring-and-analysis-with-flink-kafka-hb/.

This form of use does not limit to telecom businesses – VoD services like Netflix are also an area worth investigating: https://www.datanami.com/2018/04/30/how-netflix-optimized-flink-for-massive-scale-on-aws/).

Conclusion

Stream processing provides excellent possibilities in the field of fraud detection. Apache Flink, a technology acknowledged by many companies and various industries, is ideal for this purpose. Nussknacker transforms the power of Flink into a handy tool that minimizes the time and effort of the IT team and empowers the business team in implementing and testing new business logic.

If you are looking for a technological solution to support your fraud detection systems and facilitate development and business processes, contact us. We'll be happy to show you how Flink and Nussknacker can help you achieve these goals.

Learn more:

https://touk.pl/esp/

https://touk.pl/portfolio

https://github.com/TouK/nussknacker/

https://touk.github.io/nussknacker/

TouK
make IT simple