



PAYSTRAX success story

Mikas Šimoliūnas | BI Developer



About me

Mikas Šimoliūnas,
BI Developer at PAYSTRAX

Avid gamer

- Europa Universalis IV
- Path of Exile II
- Stoneshard
- Barotrauma

Boardgame enthusiast

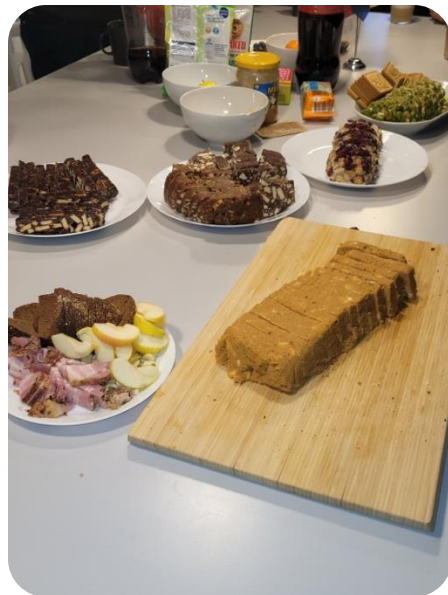
- Dune Imperium (with Rise of IX expansion)
- Root (with expansions)
- Harmonies
- Brass: Lancashire (should really get Brass: Birmingham)

Baking beginner

- Out of this world lazy cake (tinginys)
- Curonian roll (Kuršėnų vyniotinis)
- Cottage cheese cake (varškės pyragas)

Mini figure painter

- Mainly from purchased board games
- Would love to try painting Warhammer figurines





PAYSTRAX



About PAYSTRAX

PAYSTRAX is a **payment services provider** with Payment Institution licenses in Lithuania (issued by the Bank of Lithuania) and UK (FCA license). We help businesses get payments as quickly, securely, and cost-efficiently as possible. We provide **Visa and Mastercard** transaction services for POS-terminals, online, mobile or any other electronic payment systems.

100+

Employees (over
100% growth
compared to 2022)

~40%

growth of merchants*
since 2023

490 mln.

Acquiring
volume in 2024

*Merchants - Businesses or individuals that sell goods or services, typically in exchange for payment.



Our values



Ambition

We reach further by:

- Doing the impossible
- Advancing courageously
- Striving for excellence
- Learning continuously



Professionalism

We create trust by:

- Appearing and communicating professionally
- Committing fully
- Fostering mutual respect



Fun

We make it worthwhile by:

- Getting a laughter going
- Embracing a stress-free environment
- Building teams
- Celebrating achievements and making memories





Choosing the tooling



We picked...



What we were looking for



- Since we started from scratch, we looked into innovative and cutting-edge technology because no tech stack was predefined, we were looking for a dedicated central data warehousing solution.
- Avoiding on-prem servers with high up-front costs.
- Flexible scalability.
- Low to none maintenance on computational hardware (servers, VM's).
- Avoiding a dedicated DBA role.

WHY SNOWFLAKE ?



What Snowflake delivered



Target ➡

- Extensive and easy to follow documentation along with excellent support was a breeze to get the systems up and running.
- Snowflake cloud focused storage and processing allows to completely forget about hardware maintenance and dedicated maintenance personnel.
- Flexible resource allocation with an extremely precise way of calculating cost.
- Scaling up (computation) and wide (parallelisation) without upfront investment.

Extra ➡

- Automatic encryption of data at rest and in transit.
- Automatic partitioning (Snowflakes micro-partitioning architecture). You can forget about partitioning and order keys.
- Support for structured and unstructured data formats (csv, tsv, json, avro, parquet).
- Granular access controls with users and roles.
- Object tags and columnar/row level masking policies (Amazing for data governance).
- Amazing disaster recovery of data with time travel and fail-safe.

WHY SNOWFLAKE ?



Places where Snowflake shined for us

Environments

Easy to setup multiple accounts.

Accounts can be set up on: GCP, AWS and Azure. We use Azure and AWS.

Comfortable segmentation of development environments using separate accounts (ex. dev > uat > prod)

Replication

Straightforward and easy to use for databases and shares.

Schedule or run manually on demand. Scheduled executions do not clash, and a queue mechanism manages every execution.

Flexibility

Distribute workloads over multiple accounts and/or warehouse units.

Possibility to automatically allocate more resources for demanding tasks (multi-cluster warehouses, QAS)

Mix and match Snowflake account types with specific workloads (storing and processing in standard, masking in Enterprise etc.)

Data sharing

Supports csv, json, xml and other formats.

Smooth integration via data shares with any service provider who also uses Snowflake for seamless integration and sharing.

Excellent functionality for mirroring prod data in dev.

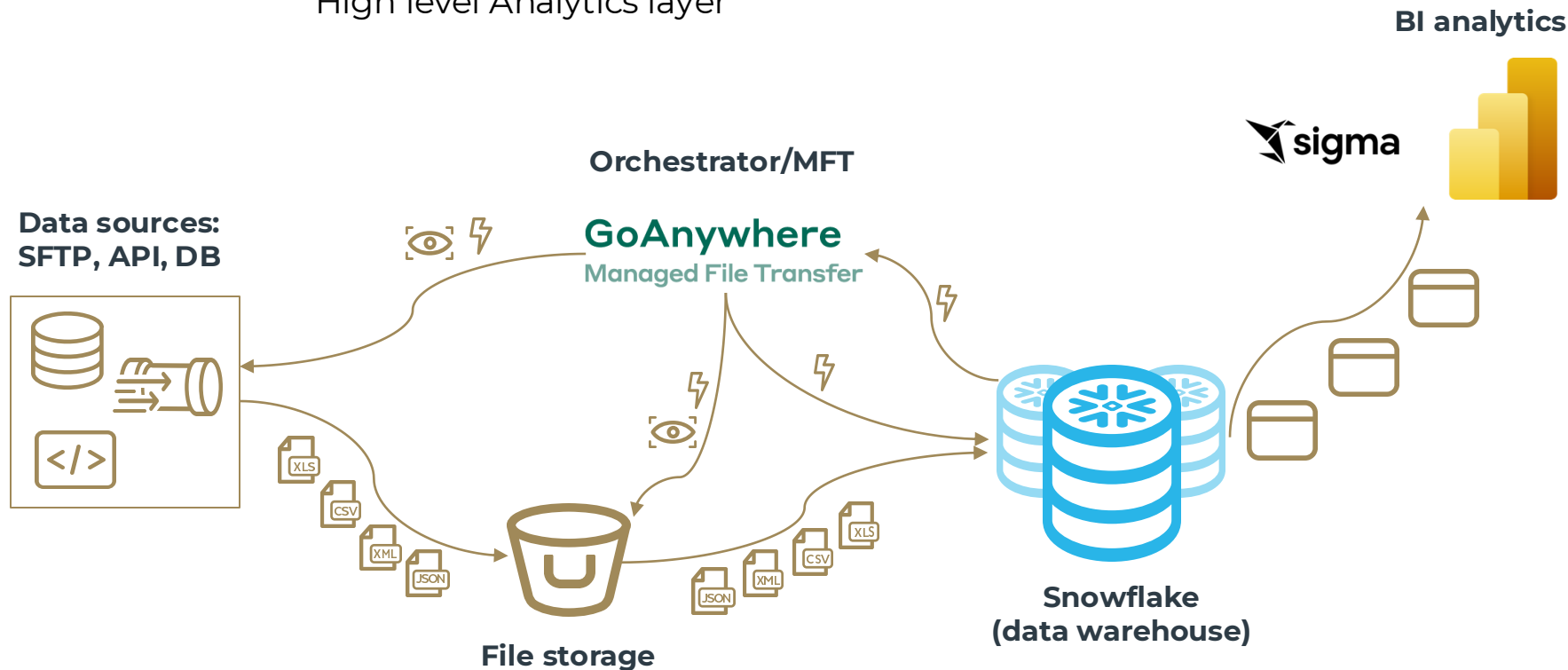
File ingestion and dispatch using AWS S3, Azure blob storage etc.





Architecture

High level Analytics layer



What we achieved using Snowflake



01.

Centralised data storage and avoided data silos. Prod environment contains the golden copy of data and using shares we can compare, reload and “test in production” through development environment.

02.

Logging framework integrated with orchestration tool. Monitoring data health and report availability.

03.

1000+ of data objects and transformation processes for self-service analytics made possible via Power BI & Sigma (& Excel.). Actively supplying over 100+ analytic solutions.

04.

Data quality framework. Defined business quality rules for validation, system comparison and pruning.

05.

Data statement delivery service. Clients receive detailed information via email about their business performance.



Lessons learned



1. Look into logging frameworks or tools and get them up and running ASAP. Having eyes on your data pipelines and jobs will reduce maintenance overhead if something goes unplanned (and it will eventually happen).
2. If you have time sensitive data, but are not storing historical snapshots. Start today. Your business users and analysts will thank you for historical data.
3. User and role management. Dedicate time and resources to map and configure roles for any use cases you might have. Streamlined roles will save you weeks of work.
4. Read the documentation. Read the documentation. Read the documentation. It contains a plethora of useful information and functionality no other tool can offer.
5. Do not be afraid of failure. Mistakes happen and projects have rough starts. Developers around the globe have experienced such hardships and they share their stories. Read forums and information sharing stacks. Take the time to reflect on your own work too.
6. Time travel and fail-safe are lifesavers. You can never be too prepared for disasters and there is no better feeling than knowing you are protected and taken care of (We have experienced such disasters a couple of times and time travel saved us).



What's next



Streams

Start using Snowflake native streams for logging DML operations to improve overall process visibility and health.



DMF's

DMF's (Data monitoring functions) are a native approach to monitor the state and integrity of data. Enhances visibility significantly when used alongside streams.



Data governance

Masking policies (column-level or row access) in tandem with tags. Creating a governed and managed data environment.

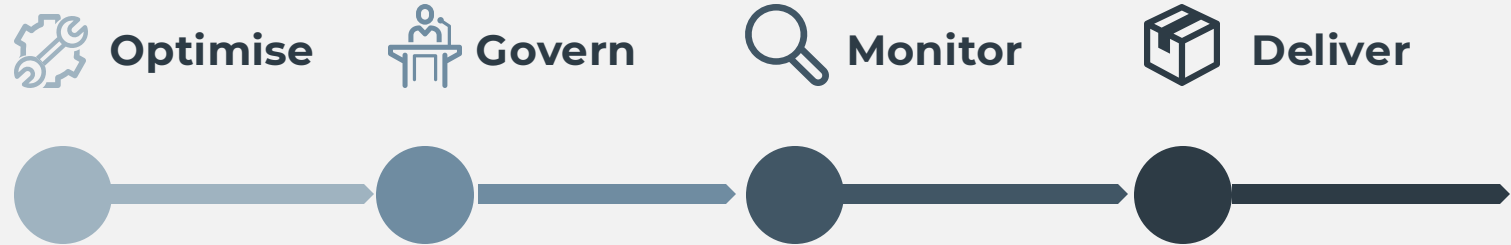


AI, ML & Cortex

Explore and look into Snowflake ML and Cortex models and functions.



Road of success



Thank you!