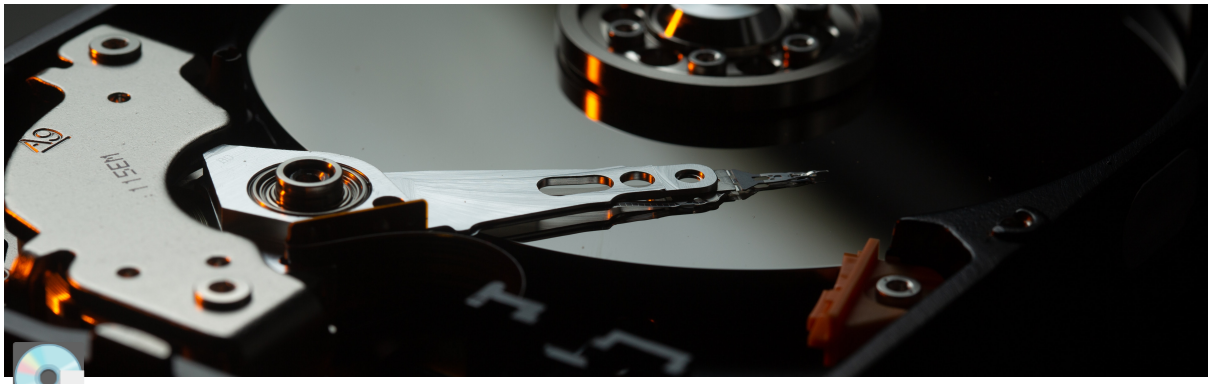
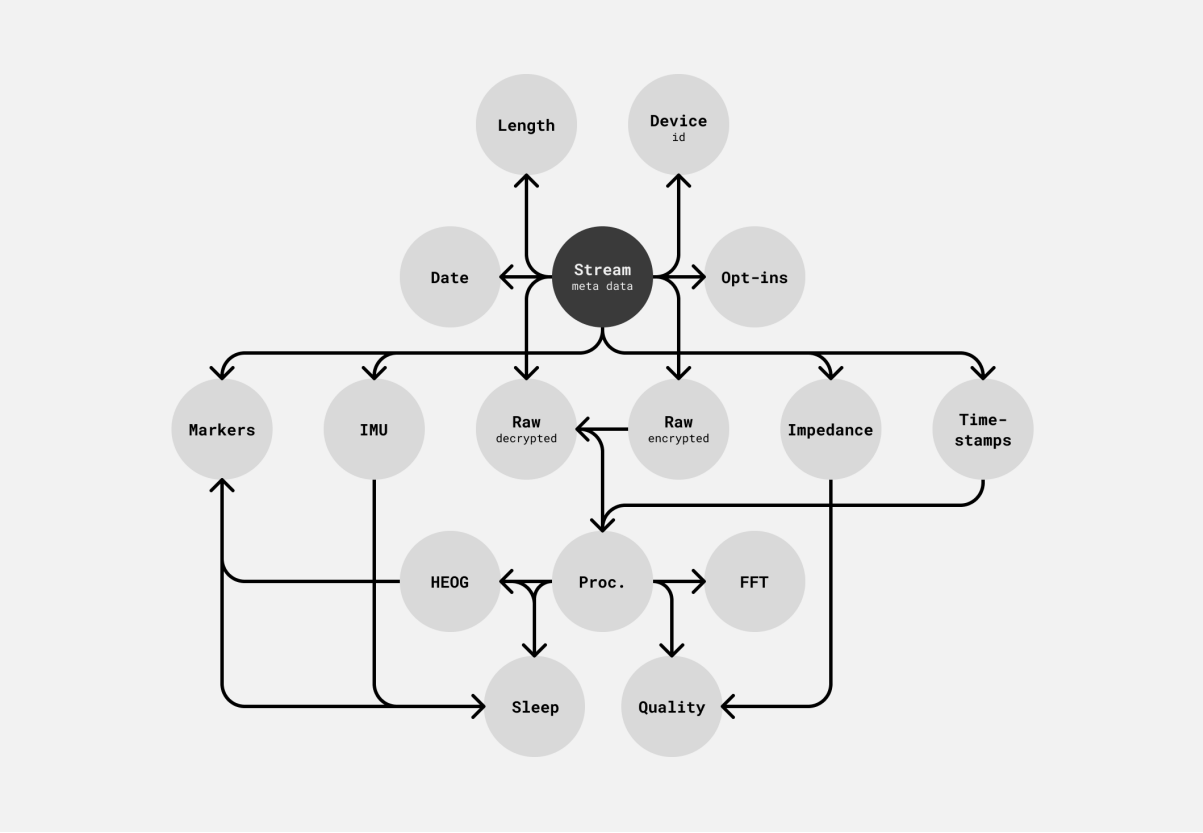


Data Model

This page describes the data mode of the data collected by IDUN's hardware. We initially store the data as CSV files on S3. The decision for that is to have an interchangeable file format since we offer to download these files or to convert them to other file types. The aspect of querying the data also doesn't happen inside the time series aspect of the CSV files themselves, but on a graph-like way between collected meta data or associated other file types (i.e. transformed or processed version of the original CSV) which are all stored inside the database.



Example illustration of the graph-like data structure between files and their meta data (from Daniel's bachelor thesis).

Model

The architecture of IDUN's backend is event-driven by nature. The main events in our cloud will be streams. A stream is a collection of recorded data from the IDUN hardware. It includes most importantly the EEG data as well as other collected time series data (i.e. IMU). Every stream has certain files associated to it, e.g. the raw encrypted EEG data, the transformed and decrypted EEG version of that, associated markers etc. Per file type we might have some context information as well as meta data per stream. Therefore we proposed to use the following data model per stream:

id*	device*	firmware*	local start time*	cloud start time*	encrypted*	decrypted*	battery*
Unique ID per stream, overall unique within all devices	Device ID that has been used to record this stream	The firmware version of the recording device	Local timestamp	Cloud timestamp	Link to the encrypted raw data stream as collected from the device (includes raw EEG, encrypted IMU and markers)	Link to decrypted raw CSV file	Link to b informati the strea

* cannot be changed after initial creation

Inside the files

Inside time series-based files such as decrypted raw EEG we will have this model:

numbering	ch1 EEG	markers	battery
Unique numbering per sample (initially it will be counting up from 1 to n)	Raw EEG data from channel 1	Markers array since there can be multiple markers per sample: [”a”, “b”]	Battery information per sample

We have a unique numbering per sample so that if we transform the raw data into another time series file we can associate sample-by-sample: if sample `123` gets e.g. classified in the HEOG classifier and then tagged by a marker of e.g. `L` (meaning looking left classification), we can reconstruct a time series file just for HEOG classification also as a time series CSV and match the information per-sample with e.g. the raw EEG, IMU or others.

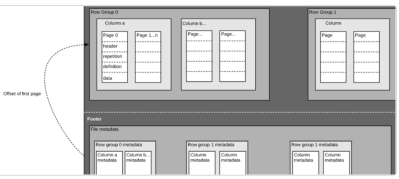
Future

We aim to store the CSV files in the Parquet file format to increase reading performance and save costs due to file format sizes. More information:

What is the Parquet File Format and Why You Should Use It | Upsolver

Trying to wrap your head around data lake concepts? We've written a practical handbook to help you with that. The ebook covers guiding principles for modern data lake architecture, storage best practices, ingestion pipelines, data processing, and much more. Get it for free here.

https://www.upsolver.com/blog/apache-parquet-why-use



Apache Parquet Explained in 5 minutes

Data #ApacheParquet #GCP #PySpark #DataprocWhat is Apache Parquet ? When it can be used ? and how to convert CSV to Parquet using PySpark job with cloud Dat...

https://www.youtube.com/watch?v=VZykcApkz_4