Goal

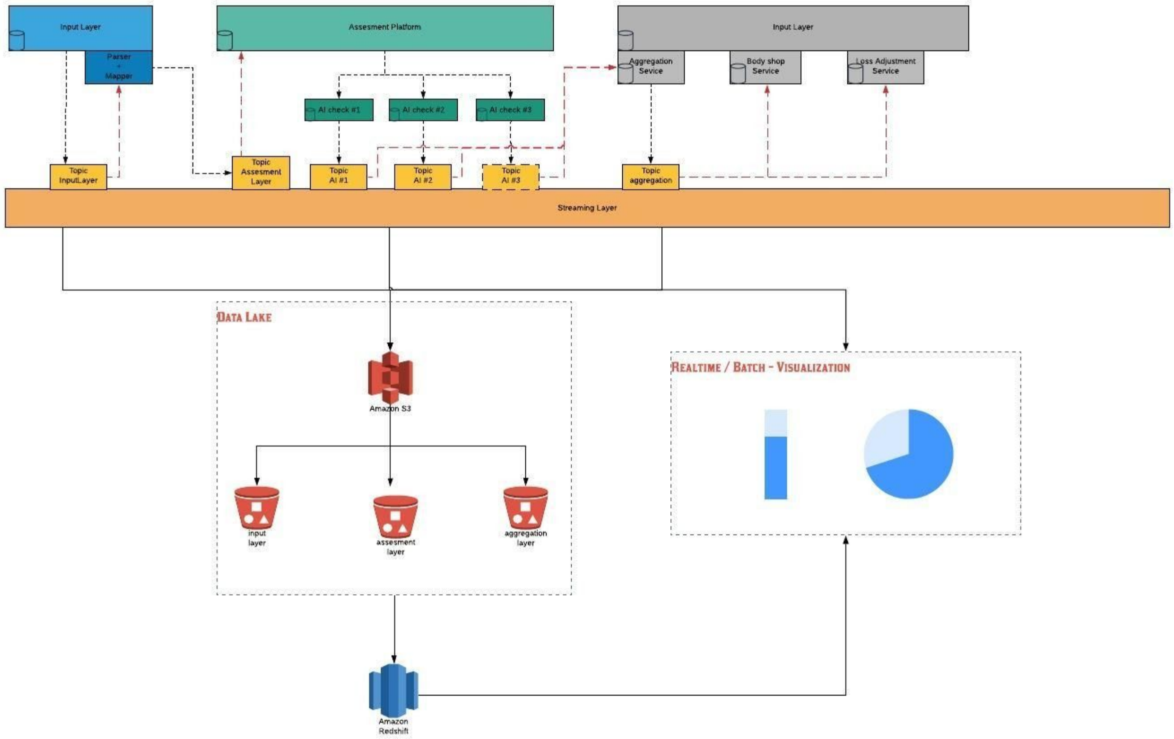
To come up with a new architecture that will be least intrusive to the existing workload & teams.

Assumption(s)

> Acme now wants to enable their analytics team to build a data analytics layer allowing the company to drive comprehensive insights from the data ingested by the product.

- Analytics layer: Infrastructure / Tools in place to answer metrics similar to the ones mentioned below
    - How many total claims have we done so far ?
    - What is the *min / max / average* amount claim made ?
    - How many positive feedbacks do we have in total ?
    - What are the feedback sentiments for claims marked as anomalies over the years across the world ?
    - How long does it take to solve an anomaly claim in each country ?
    - Etc...

Proposed Architecture



- The main goal here is to achieve Event Driven Architecture that is scalable and is highly flexible for future use cases.

- If we make changes to the existing system on how the input is received and where the outputs are stored, we can achieve pretty decent self sustaining micro services which purely focus on their functional operations and we can orchestrate the end to end process flows & retry logic on application level and the end to end orchestration can be solely left to stateful orchestrators.

- By making use of streaming services, we can easily collect the data into multiple services like a data lake, or visualisation tool, ad-hoc internal application, etc... without adding any extra stress onto OLTP systems.

- This data Lake will then be point of source for Batch ETL jobs which will then be transformed & stored into Data Warehouse

- The proposed solution does not include ETL jobs to fetch all the data from microservice databases into DWH / Data Lake. Happy to follow up on that in the future conversation.

Demerits

- **If streaming service is not chosen carefully, it will become a single point of failure.**