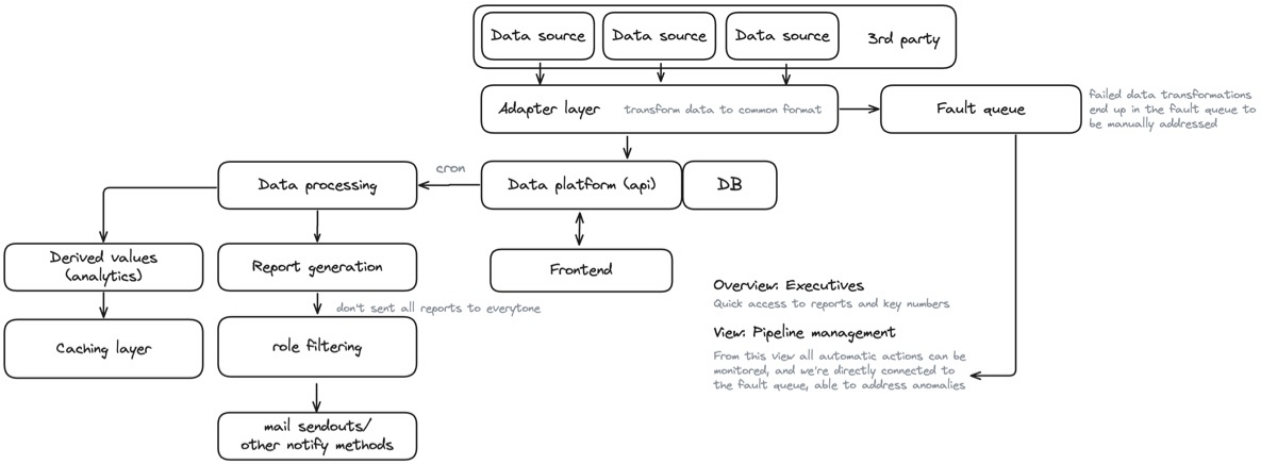
Sprints - Platform Engineer case study

Hampus Hallkvist

I always like to start on the other end when planning something like this, so the question is where are we in 90 days? So we're trying to answer the question, "What do we consider to be a successful outcome"?. My definition of a successful result is a platform that is accessible to everyone on the team, yet automates and processes data from different sources with minimal human overhead.

As stated by the case part 2 we're looking to accommodate a handful of stakeholders here, among the 4 the investment team and the finance team will both read/write data to the platform. They own their data, investor relations and executives only need access and overview in my understanding.

To reach the wanted state of having everything in one place, accessible by everyone, there has to be a standardized way to move all of the data through. Given that we have multiple data sources such as third-party tools or off-the-shelf solutions I'm imagining an adapter layer that everything is moved through, then storing everything in the wanted format (something I consider to be a first principle design goal). The obvious benefit of a unified data format is that all information can be used through the tools that are created, such as visualization- and management tools. In the image below the data sources are not necessarily owned and maintained by us, but we do own the adapter layer and the platform itself, allowing us to morph data as we desire.



As for the 90-day plan and execution thereof, the best thing in my opinion is always addressing the stakeholders, all agreements and disagreements have to be ironed out in order for everybody to share the same vision. Sharing the same vision is important as it aids the development process, instead of hindering it as it otherwise might do due to differing views and ideas. Initially I think it is important for each entity/ person to own their problem, the first cause of action would therefore be to have individual sit downs with all stakeholders separately, gather intelligence on desires, compile the needs, and them in a group setting discuss the direction we're taking. I'm envisioning that I'm quite hands-off initially, encouraging open-mindedness. As I then get a better understanding and the vision gets more clear I would also start to get a better understanding myself of the direction. While the goal is to find something that all stakeholders are satisfied with, there will have to be some priority in place as everything won't be able to come at once. My current take on this would be that prioritizing a framework to unify would be of very high priority, having this in place could then later allow for extended features, etc. Visualization and data display while being important features are not urgent to get done, they won't have an effect on anything else in the system. How the data is processed does, getting that right will be very important for the rest of the project. As a result of this, I therefore think it would be beneficial to listen extra carefully to the "Investment team" and the "Finance team" as they are "data owners".

While developing this system there will be challenges, I think the greatest threat to this project would be "scope creep" (the project dying from expanding the scope too much). It is for this reason that I think it is beneficial to have the initial sit down with everybody to align the expectations and the result, the system won't solve all the problems after the first release, and the MVP will have to be a stripped-down version of what everybody expects. However, when this MVP is operational, then, and only then can we build from it and start to satisfy all other non-mission-critical requirements (such as more ways to visualize data, different interactions with the data and so on).