### **Introduction**

Tommy Wu leads the Analytics Engineering team at [OneOncology](https://www.oneoncology.com/), a nationwide network of oncology providers. Within OneOncology, Tommy is part of the data team building an analytics platform that leverages clinical, operational, and financial data to enhance practice operations and enable physicians to provide the best care for their patients.

### **OneOncology’s Stack**

OneOncology uses a modern data stack primarily powered by Microsoft and Databricks/Spark:

* Data Warehouse & Orchestration: Azure Databricks – where the data from production is mirrored; there’s an additional proxy for analytical use.
* Transformation: DBT + native Databricks capabilities
* Visualization: PowerBI - used for production dashboards for decision makers as well as self-service tools for analysts. PowerBI’s native Excel/PowerQuery integration is especially powerful.

Within this stack, everything up to the data getting into the source schemas in Databricks, is the primary purview of another team, the data engineering team. Tommy will frequently collaborate with them to ensure that the right data is brought in, but his team specializes in what happens afterwards – curating & scaling business logic transformations to empower analysts.

### **Data Ingestion at OneOncology**

The data that is being ingested into the OneOncology data warehouse is broadly of two categories:

#### Operational data from OneOncology network practices and data partners

Among other things, this includes core claims and EMR data that is common to every healthcare organization. There’s a degree of variance in terms of how easy this data is to onboard. If the provider network is already using systems that are common & industry standard for community oncology, then the team is able to onboard data very efficiently– “it’s like turning on a switch”, Tommy says, because “we have developed great relationships with many of our data partners”.

But for more bespoke systems that OneOncology has yet to directly work with, data onboarding is a more time-intensive process. “What we ideally need then is a well-governed data dictionary. If one is not immediately available, we will consult with domain stakeholders + analyze the columns and try to identify & reverse engineer what business processes each field relates to and build our own dictionary.” By and large, though, the data team is able to use first principles of how healthcare data works, ex. “this is what a claim should look like. This is what a chart should look like”, to figure out a mapping of the columns.

#### Additional data from public & commercial sources

Like many other healthcare firms, OneOncology also ingests third-party data sources to enrich our datasets and enable analytics. These include public sources such as the CMS’s (Center for Medicare & Medicaid Services) data releases, as well as other commercial sources. For this part of the data ingestion, the OneOncology data team has built up a deep understanding of each domain and maintains an ontology of proprietary mappings between external & internal sources; for example, normalizing procedure codes and their meanings.

### **Data transformation at OneOncology**

OneOncology uses Databricks DeltaLake, which allows them to keep raw CSV or Parquet files (in this case, source files from data partners) – at the lowest level of the lakehouse. Any subsequent changes made to the data is able to be fully traced back to the source on Databricks.

What this means is that the source is always tied to every table downstream; conversely, it’s possible to trace every table to its upstream original source. “This is something I'm really proud of the engineers for building up before I even got there,” Tommy says.

After the data has arrived in Databricks, the data is further refined & aggregated within Databricks into core & domain specific tables using DBT.

### **Tommy’s lessons from working with healthcare data.**

In addition to his time at OneOncology, Tommy has spent close to a decade working as a data scientist in early stage healthcare startups. By and large he’s noticed that similar objectives, data workflows, and challenges exist within every healthcare data org. Below are some examples of common data challenges he’s encountered:

#### Data definitions disambiguation

In healthcare, even if two organizations use the same data product (such as Epic), the way they configure the product and input data can vary greatly. For example, one group of doctors might use the "location" column to represent the physical location of their offices, while another group might use it to categorize their physicians based on specialty (e.g., "radiation" or "surgery").

This can make the analytical aspect of data quite challenging, and it's important to be cautious when investigating data to ensure that the definitions are consistent/as expected.

Vendor Schema Changes

A common source of data quality issues in healthcare is vendor schema changes. This is when a data vendor changes the schema on their data, but doesn’t inform their customers in a timely fashion. When this happens, receiving data teams have to rapidly troubleshoot & manually remap columns.

One solution to this is to set up alerting & gating so that the data is stale rather than bad until manual corrections are made.

#### Staleness

In addition to pushing silent schema changes, vendors can also be late with their data, resulting in stale data for healthcare companies that are consuming this data.

One tip from Tommy is to take existing logging the engineering team has in place- for example many engineering teams already regularly log and review the max date of the data - and summarize & expose it to stakeholders in a more digestible, user-friendly UI.

“Purely by knowing that the bottleneck is not from errors in our codebase, but that it’s instead due to upstream data vendors not sending us the data yet -- that's saved a lot of energy and increased trust through transparency. We used to get Slack messages or emails all the time about “hey, why is this data missing?” Now non-technical stakeholders can look at regularly updated dashboards and realize that there’s an extra day lag for this data due to XYZ data source being delivered late,” he said.

#### Extreme outliers

Another common data quality problem in healthcare is extreme outliers in data – for instance, the data should be in the thousands but there’s one spike in the millions. This is generally an indication that somewhere upstream a data entry or transcription error occurred, but until the error is fixed, the outlier will likely heavily skew analytics.

A potential solution is automated exclusion flags that exclude any data points that are, say, more than three standard deviations from the norm. You can start out by doing these checks manually; eventually, you can automate them with tests in DBT/Databricks, or a continuous monitoring solution like Bigeye.

#### Business logic errors

While many of the data quality issues we’ve covered are happening on the source data ingestion side, it’s also possible to make mistakes in the business logic transformation code.

To mitigate the chance of this, Tommy recommends that data should be treated as code - every material SQL transformation written within an organization should be version-controlled and code reviewed.

### 