We’ve updated the logic for how events are pulled from the queue in each refresh cycle. The goal is to make processing more efficient by prioritizing larger events and combining them in a way that uses our capacity effectively.

Here’s the new approach:

**Event Categories (by size of rows in millions)**

* **XL (Extra Large)** → > 200M rows
* **L (Large)** → 100–200M rows
* **M (Medium)** → 50–100M rows
* **S (Small)** → < 50M rows

**Pulling Logic (per cycle)**

1. **If there’s an XL event:**
   1. Take **only 1 XL** event and process it by itself.
   2. This ensures the largest jobs don’t slow down smaller ones in the same cycle.
2. **If there are at least 2 L events:**
   1. Take **exactly 2 L** events and process them together.
3. **If there’s 1 L and some M events:**
   1. Take **1 L** plus **up to 5 M** events.
4. **If there are at least 10 M events:**
   1. Take **exactly 10 M** events.
5. **If there are fewer than 10 M events:**
   1. Take all the M events available, and then fill the batch with **S events** until you have 10 total.
6. **If only S events are left:**
   1. Take all remaining S events.

**Why this change?**

* **Larger events** are handled separately to avoid long refresh times holding up smaller jobs.
* **Medium events** (the majority) are processed in larger batches for speed.
* **Small events** fill the gaps so we make the most of each cycle.

This approach should give us better performance, fewer refresh cycles, and more consistent processing times.