Need Find Activity

Qitian Liao, Richard Lin

Process Guidelines 1. Before you conduct the video call, write parts 1-3 of the writeup (described below). This should help you think about what you want to get out of your need finding study. 2. Before you conduct the video call, write out your plan for what you'll do--questions you'll ask, activities you'll ask to observe, etc--during the call. Summarize this plan for part 4 of the writeup. 3. Both partners should be present for the video call. It's up to you how you split the work of conducting it, but one option is to have one person in charge of communicating with your participant while the other partner is jotting down notes about what's happening. 4. Immediately after the video call, and before talking to your partner, write down the key insights you took away from the call. Or, if you didn't come away with key insights, write down why you think it didn't produce insights. Was it the wrong interviewee? Did you ask the wrong questions? It's ok if you don't get any insights this time! Doing need finding well is tough, and it can take a while to develop this skill. (Summarize in writeup part 5.) 5. After you've had a while to process the call, compare notes. What insights did you both find? What insights did one partner come up with that the other didn't? Do any of your insights actually contradict each other? Take notes on any new insights that come from discussing the call together. (Summarize in writeup part 6.)

Data Engineer Role: https://www.wayfair.com/careers/job/data-engineer--m-f-x--/4762098002?gh_src=a5f36eaa2

# Writeup

**Part 1 < 0.25 pages Description of the programming domain you're exploring.**
The programming domain we're exploring is data analytics and warehouse data management. More specifically, this would be looking at how data is analyzed and served in a production environment.

**Part 2 < 0.25 pages Description of the full need finding study you'd conduct if you had bunches of time and resources.**
Find beginner, intermediate, and advanced users in the programming domain we're exploring and conduct contextual inquiries on tasks in that domain, preferably related to their work. Also ask them to accomplish several tasks in that area, both ones they have experience with and aren't so familiar with. From this, we could try to find similarities in pain points across users and how they might differ depending on the task, expertise level, and methods used. The insights could then be used to focus in on the specific needs and determine which would bring the greatest impact.

**Part 3 2-3 sentences Description of why you chose the particular person you chose for the video call. Job? Particular prior experience? Other?**
Yanran's been working as a data engineer at Wayfair for 2 years and has proficient knowledge in its day-to-day workflows. Therefore, we believe that choosing her as our interviewee will provide us with profound insights into the potential programming challenges she would face in her job.

**Part 4 < 0.25 pages The plan for the video call**.

We will ask the interviewee to show us what her typical workflow is like for analyzing data and implementing data architectures. We will try to observe the shortcomings in her workflow, ask questions about anything we don't understand, and follow up with questions on why she does something a certain way or if she's considered any alternative methods for her work.

Questions for during interview:

1. Describe the tasks you're trying to accomplish
2. How do you go about doing it?
3. What alternatives have you considered?
4. Confirm observations with interviewee
5. Follow-ups depending on what we observe

**Part 5 Up to 0.5 pages per partner, but less is fine Notes from each partner about their individual insights from the call.**

**Richard's Notes:**

Context: User needs to send weekly executive reports and has to check whether there are any issues

- Analyzes chart and see if there are abnormalities with the chart
    - If so, investigates whether problem is caused by some business problem or data processing
    - There are a good number of charts to go through, but doesn't take too much time

Problem diagnosis process:

- Go to the chart on the dashboard and look at underlying SQL queries
    - Dashboard is backed by multiple backend clusters and data is pipeline through the clusters.
- Check first cluster and run query on it
    - If query gives expected output, then move onto next step. Otherwise, problem is here.
- Check pipeline procedures that are executed
    - There is tool that records the backlog of procedures that are executed. User need to look through the backlog and find the corresponding procedures. Doesn't seem like a very smooth process and complains about the slow and limited functionality.
- Check query on other cluster
    - Look at specific table with data that is used
    - Inspect table. Needs to have a good amount of knowledge regarding the exact tables, comes with exposure and time of use.
    - Look at logs of query and procedures
- Reach out to team that is responsible for data source
    - This seems to be a very adhoc approach and can lead to some friction with turnaround time. The team can then help investigate issue
- Root cause is determined and addressed in the future (hopefully)

Needs that were inferred:

1. The company needs to stay up to date on changes to key metrics
    a. This is currently done with business intelligence tools
2. Yanran has to detect and investigate potential issues with data sourcing and processing

. Seems pretty dependent on experienced analysts, but there are many things that could be automated

3. There are a lot of tools that are used. Maybe there could be a more seamless way of identifying the problem

Follow-up questions:
If the company had a lot of resources, how do you think they would improve this process?
Hire more analysts to look at these issues. Automate some of the process.
Are there automatic detectors for data anomalies?
There are some automatic integrity checks. Identify these numbers and see if they go beyond the threshold we set. Sends an alert email if data exceeds a range. There are 30 metrics and many different dimensions, not easy to do integrity checks on each metric. Very strict checks for some metrics.
What do you spend the most time on?
Looking at charts takes a little bit of time, but mostly diving into what the root cause of an issue is. There are a lot of things to look at and we need to break the issue into many steps and eliminate possible causes.

Other:
- Some tools are in the work to automate parts of the process.
- Not easy to adopt external tools to internal practices. Build own tool and scale it.
- Needs to jump around to different platforms
- Some tools are very slow and limited.

**Qitian's Notes:**
User (Yanran) introduced us to her standard QA procedures. Namely, she told us how she would identify the causes of spikes in business charts, whether because of actual business problems or rather some data anomalies.

She showed us the two clusters her company uses: GCP (Google Cloud Platform) and Vertica, which are the two potential places where the error could have occurred.

First she checks whether the numbers in GCP are the same as the dashboard. If not, then the dashboard is not showing the right numbers this might be due to internet lags. So she would try refreshing the page and see if the numbers become coherent.

If the numbers are indeed the same, then she needs to check if there is anything wrong with Vertica. There could be problems with pipeline procedures on Vertica, so she would check with Kronos (a backlog of procedures) to see whether the procedures are executed as expected (there is also information about status, qid badges, ids of the procedures).

Then she goes to the source tables and checks if the table is updated as expected. She looks at the SQL queries themselves. The final block of the script is the one that inserts the most recent data to the table. She tracks from downstream to upstream, and eventually she could locate the source table by investigating the SQL code. If the source table is not managed by her, then she would go to corresponding channels on Slack and find the team that has the ownership.

There are other people in Yanran's team who look at the data constantly. Stakeholders will also check for potential anomalies in the data. People do Q&A's on a very regular basis. Currently there are automatic integrity checks, which will send alert emails to users when the numbers exceed a certain criterion. The visualization tool she uses, Looker, also sends out alert emails in case of abnormalities in numbers. There are more than 30 metrics in total and even more

dimensions. Consequently, there are relatively strict checks for a couple metrics since it's hard to do integrity checks on each metric.

Yanran thinks that coordination between teams could be improved, since queries on slack are sometimes not answered promptly. Furthermore, she complains about how slow Kronos is, and there is a desperate need to speed it up. Lastly, she says that Looker is an external (third-party) tool which is not directly associated with her company, so it's hard to apply external tools on internal needs since she has to manually import code blocks to make them compatible with the software her company uses. She expresses the desire to make her own programming tools.

**Part 6 < 0.5 pages Notes from the post-call debrief, including any new insights that came out during discussion.**

There were many issues that came up within Yanran's workflow that included:

1. Having to use multiple tools because the data pipeline consisted of many different steps. This resulted in a less efficient process and having to duplicate a lot of the tasks across different platforms.
2. One surprising concern was actually coordination between teams. Many of the issues that pop up are due to lack of communication between teams about changes in data processing and formatting.
3. The requirement for significant backend work to visualize data. Yanran has to spend a lot of effort to visualize data and is frustrated by the tools she has to use. Being able to visualize the data in a simpler manner would help her with building these reports significantly.
4. The problem of low internet connectivity is surprising, and it's especially severe in times of working from home. She mentioned that some of the tools she uses daily run slowly and partly it's because people are using VPN instead of directly connecting to the company's wifi. This is a problem we didn't expect because most of the tools we're used to nowadays (google docs, android studio, etc) run in relatively high speeds.
5. Creating integrity checks on data is not easy because there are so many metrics and dimensions to keep track of. Automating the checking of data would reduce time spent on this.

Surprisingly, many of these needs are not related to technical tools she works with. We could potentially address issues in team coordination or company-wide communication. From just this call, we were able to detect many pain-points and think there is a lot of room for improvement.