

Queries on Queries

Questions to ask when migrating data

Who am I?

Twitter: @acrosman

D.o: acrosman

Web: spinningcode.org



Are your tools good enough?



Our migrations live and die by our tools.

Are your tools built for the scale of your project? Do they empower you to do your best work or impede rapid progress? Would a new tool serve you better now or in the future?

Do you make the data and processing atomic?



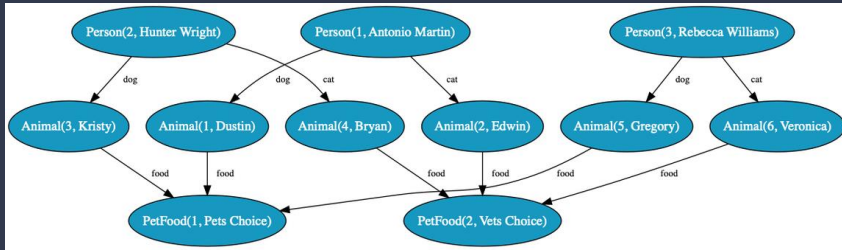
Smaller pieces of data are easier to track, manipulate, and test.

Do you divide the source data into its constituent parts? Can you process individual pieces of data easily and cleanly? Can you stop your process after each stage to validate the results?

Can you process samples of your data set?

When you have lots of data you need to test small parts to be sure your process works.

Do you know how to create and run small segments of your total input? Are your segments made up of complete and valid samples? Does your sample include all the errors and edge cases your data set will throw at your process?



Is your work repeatable?



You will need to do this more than once.

Is your process designed so you can run it over and over without error? Can you easily erase test attempts and start over from a clean slate? Do you have the capacity to do all the practice runs you need to complete your project successfully and on schedule?

Is your work measurable?



To know you moved all the data, you must know how much data is going in and how much should come out.

Can you accurately predict your output data volume based on the input size? Do you have valid estimates of the running time required for each stage based on the data volumes? Are the estimates of expected data set size from a reliable source?

Do you minimize migration data artifacts?



No two systems perfectly align, and no data manager wants to lose important data, but data for the sake of data is a distraction.

Do you only load data into the new system that you truly need? Can you easily spot the difference between a new and old record? Are there data points getting loaded that have no use case or maintenance plan in the target system?

Is your code reusable?



Migrations feel like one off processes, but teams that migrate once will migrate again.

Have you ensured that as much of your code as possible can be reused? Do you have a shared library of migration code that your whole team can access? When you write new functions are you thinking about ways to make them usable in your next project?

Is your migration testable?



All good processes are rigorously tested.

Do you have an automated testing solution that validates your process? Can you tell if the data migrated accurately after each test run? Do your tests cover the positive and negative cases?

Is your work fixable?



Migrated data often needs to be updated after the jobs have all run.

Do you have a plan to fix your data if errors are found post migration? Does your plan include ensuring you have external Ids, or other connections, to be able to update all records of every type? Have you validated this plan will work with actual sample jobs?

Did you know Attain Partners is hiring?



Come work with me.

Are you interested in working on large scale projects with excellent nonprofits and higher education institutions? Do you know Java, Apex, or any strongly typed c-like language? Can you learn new tools and platforms easily? Do you like working with smart and engaged colleagues?



I asked you my
questions.

Now you ask me yours.

