

dbt & python

better together



drew banin

co-founder, dbt Labs

@drewbanin

we're dbt Labs, we make dbt.



the company

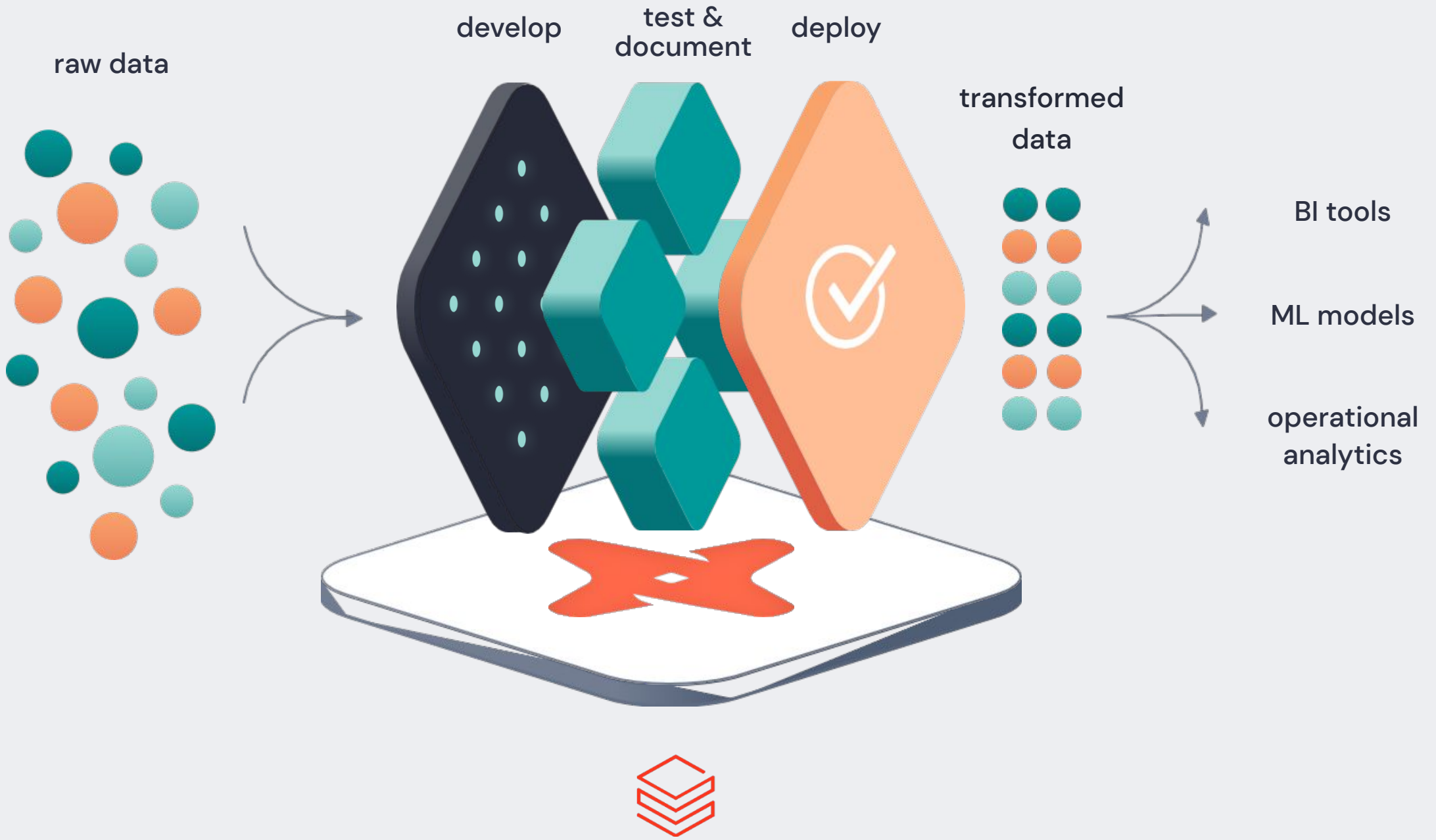


the product
(OSS + SaaS)

dbt is all up in your business (logic)

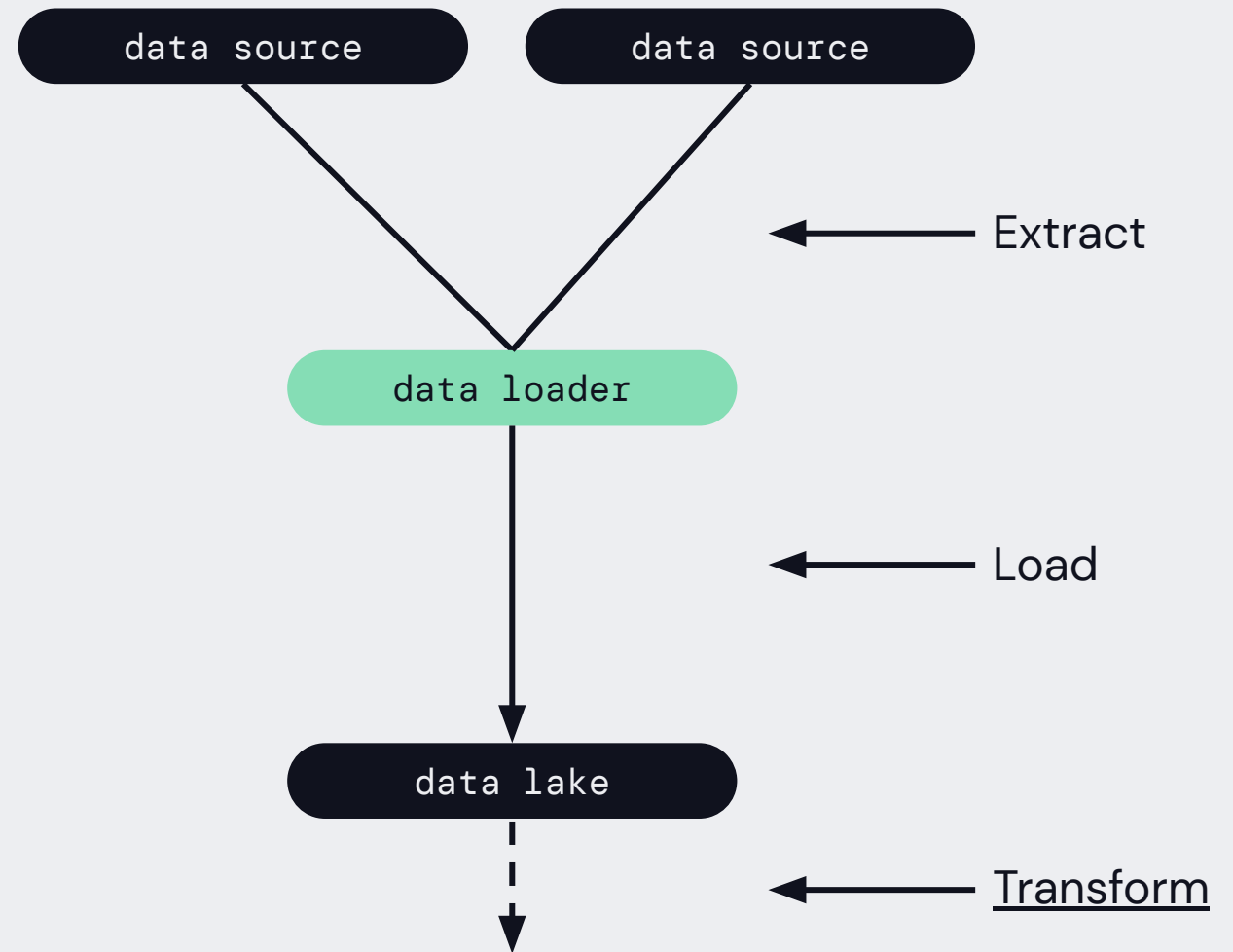
- “how many active customers did we have yesterday?”
 - “active?”
 - “customers?”
 - “yesterday?”

- In-warehouse data modeling (think: silver → gold)
 - **ELT**, not ETL
 - creates tables & views
 - version control them
 - test them
 - document them
 - deploy them



dbt is all up in your business (logic)

- **ELT**, not ETL
 - Extract
 - Load
 - Transform
- business logic changes
- storage is cheap
- flexibility is key



data sources



transformed tables



metrics

lever.application

stg_lever__applications

lever.archive_reason

stg_lever__archive_reasons

lever.opportunity

stg_lever__opportunities

lever.posting

stg_lever__postings

lever.stage

stg_lever__stages

lever.user

stg_lever__users

fct_hiring_opportunities

okr_3_3_1__headcount



transforming data

- historical focus on sql
 - CREATE TABLE AS (...)
 - accessible
 - powerful
 - scalable
- users write sql SELECT statements
- dbt issues DDL and DML to create objects
- data doesn't leave the data platform

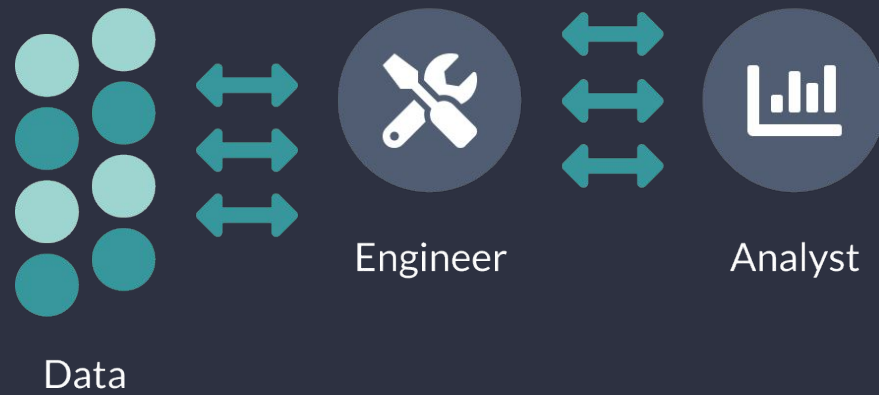
transforming teams

- business needs...
- technology processes...

- you need a bridge between these worlds!
 - *“the analytics engineer”*
 - gets the business, gets the tech
 - better work, done faster

empowerment, not bottlenecks

without dbt



with dbt



pardon my
language

did he say sql?

stg_github_stars.sql (source code)

```
1  select
2      user_id,
3      starred_at as first_starred_at,
4      user,
5      _sdc_repository as repo_name
6
7  from {{ source('github', 'stargazers') }}
8
9  --filtering on the dbt repository
10 where repo_name in ('dbt-labs/dbt', 'dbt-labs/dbt-core')
11      and star_idx = 1
```

did he say sql?

stg_github_stars.sql (source code)

```
1  select
2      user_id,
3      starred_at as first_starred_at,
4      user,
5      _sdc_repository as repo_name
6
7  from {{ source('github', 'stargazers') }}
8
9  --filtering on the dbt repository
10 where repo_name in ('dbt-labs/dbt', 'dbt-labs/dbt-core')
11      and star_idx = 1
```

did he say sql?

stg_github_stars.sql (compiled code)

```
1  select
2      user_id,
3      starred_at as first_starred_at,
4      user,
5      _sdc_repository as repo_name
6
7  from raw.github_stars
8
9  --filtering on the dbt repository
10 where repo_name in ('dbt-labs/dbt', 'dbt-labs/dbt-core')
11      and star_idx = 1
```

data sources



transformed tables



metrics

lever.application

stg_lever__applications

lever.archive_reason

stg_lever__archive_reasons

lever.opportunity

stg_lever__opportunities

lever.posting

stg_lever__postings

lever.stage

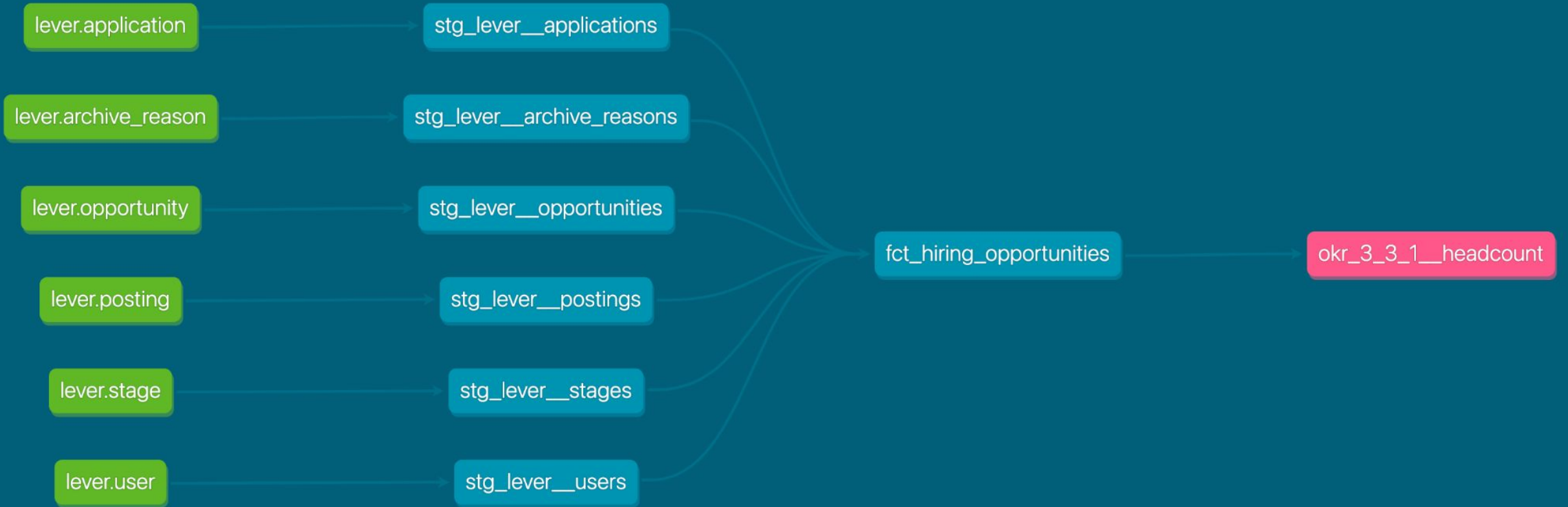
stg_lever__stages

lever.user

stg_lever__users

fct_hiring_opportunities

okr_3_3_1__headcount



the great divide: sql

- sql is great
- it's accessible & easy to learn
- it's powerful & capable
- it's been around for 40 years
- it's going to outlive us all

the great divide: python

- python is great
- it's accessible & easy to learn
- it's powerful & capable
- it's been around for 30 years
- it's going to outlive us all

...and yet

- it's two different worlds
 - with different users
 - and tools
 - and workflows
-
- data science & machine learning? **that's python**
 - business intelligence & analytics? **that's sql**

The screenshot shows a web browser window with the following elements:

- Browser tabs: One tab titled "Why doesn't dbt support Python" with a close button (X).
- Address bar: URL "getdbt.com/blog/why-doesn-t-dbt-support-python/" with navigation icons (back, forward, refresh) and utility icons (share, star, 9+ notifications, lock, puzzle, square, profile, menu).
- Page navigation: "« Back to blog" on the left and "Product News" on the right.
- Section Header: "Why doesn't dbt support Python?" in a large, bold, black font.
- Text: "dbt, our open source product for data modeling, is SQL-only. SQL can, by itself, accomplish a majority of analytical (non-data science) workloads."
- Author Info: A circular profile picture of Tristan Handy, followed by the name "Tristan Handy" and the date "4 Jul 2017".
- Text (partial): "dbt, our open source product for data modeling, is SQL-only. SQL can, by itself, accomplish a majority of analytical (non-data science) workloads. With dbt's built-in Jinja2 templating engine, including its ability to introspect the schema at"


Why doesn't dbt support Python

getdbt.com/blog/why-doesn-t-dbt-support-python/

« Back to blog Product News

Why doesn't dbt support Python?

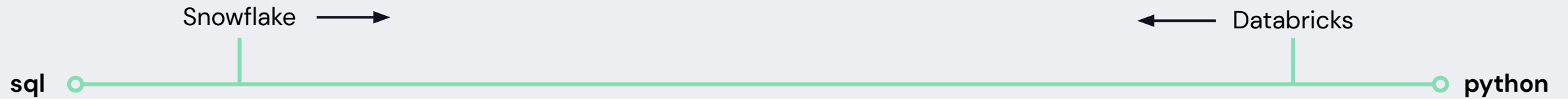
dbt, our open source product for data modeling, is SQL-only. SQL can, by itself, accomplish a majority of analytical (non-data science) workloads.

 Tristan Handy
4 Jul 2017

dbt, our open source product for data modeling, is SQL-only. SQL can, by itself, accomplish a majority of analytical (non-data science) workloads. With dbt's built-in Jinja2 templating engine, including its ability to introspect the schema at

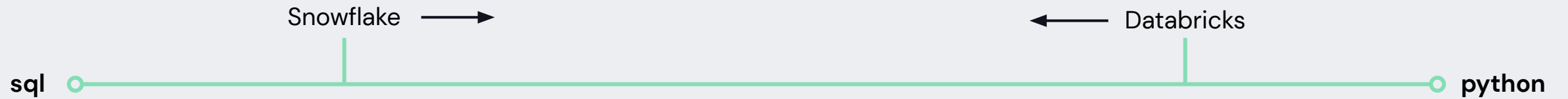
the great divide

*is being bridged**



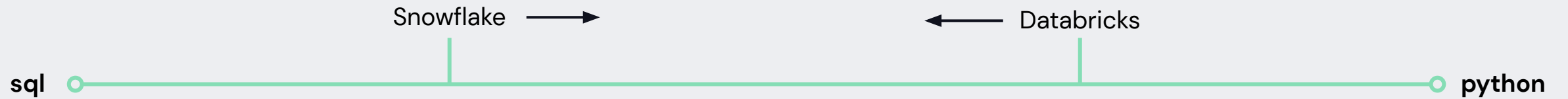
the great divide

*is being bridged**



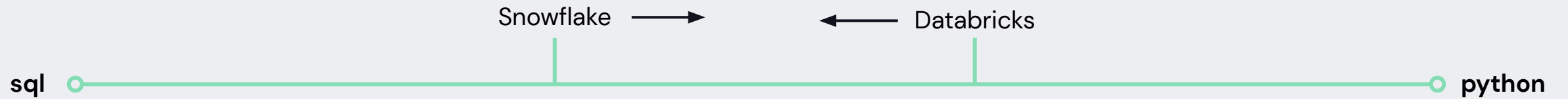
the great divide

*is being bridged**



the great divide

*is being bridged**



the great divide

*is being bridged**



the great divide

*is being bridged**



convergence is a good thing

- more
 - choice
 - flexibility
 - powerful tooling
- for
 - data analysts
 - data scientists
 - data engineers



convergence is a good thing

- for data tools
 - prepare for the polyglot future
 - eg. python models in dbt (new!)

- for data people
 - prepare for the polyglot future
 - data platforms will empower new personas to solve old problems
 - use this opportunity to move up the stack

convergence is a good thing

- historical context
 - data platforms improve →
 - you spend less time thinking about hdfs, zookeeper →
 - you spend more time on value-additive work
- let the technology automate the menial parts of your job
- focus on the value-add!
- move up the stack
 - and bring your colleagues with you :)

the big reveal

python models in dbt

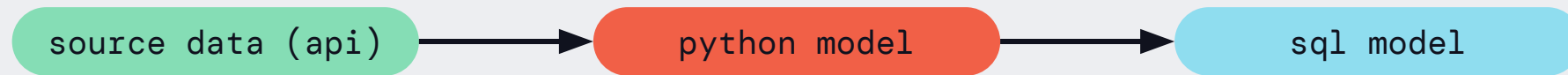
- we're adding support for python models to dbt
- rough timeline
 - beta: end of july
 - release: end of october (dbt v1.3)
 - want to help us beta test?
- different mechanisms for different data platforms
 - Databricks: PySpark
 - Snowflake: Snowpark
 - GCP: investigating...

python models in dbt: goals

- abstract environment setup and management from end-users
- all python code runs remotely
 - no local execution on client

- collaboration
- code reuse
- consistency
- unified dbt DAG

demo time



the road ahead

- first step in a multi-year arc
- we're most excited about accessibility, collaboration
 - running PySpark isn't new
 - a dbt DAG of sql and python transformations *is* new

the road ahead

- exciting ideas
 - call python udfs from sql?
 - document python models with docstrings?
 - data quality checks via unittest?
 - *The more pythonic, the better!*

-

- use spark connect

-

DATA+AI

SUMMIT 2022

thank you



drew banin

co-founder

@drewbanin