

データ変換ツール・サービス

dbtのご紹介



INSIGHT LAB株式会社

会社案内

会社名	INSIGHT LAB株式会社
設立日	2005年12月22日
資本金	1,000万円
市場	東京証券取引所 TOKYO PRO Market (証券コード 227A) 
所在地	東京都新宿区西新宿1-26-2 新宿野村ビル33階
事業内容	DXからデータ利活用、最先端技術までワンストップ でサポートするデータソリューション事業 DXコンサルティングサービス事業
拠点	札幌、宇都宮、新潟、上越、大阪、沖縄、佐渡
従業員数	約134名 ※ 2025年4月末現在
免許認証	プライバシーマーク（2008年8月取得） 労働者派遣事業（許可番号：派13-305281）
パートナー シップ	 snowflake  Qlik  dbt  sisense  vizlib an insightsoftware company

MISSION

ビッグデータを活用し、
より豊かな社会を創る

VISION

データを扱う技術者スペシャリスト集団として
顧客の心を深く理解することで企業へ
新しい価値観を提供する

PHILOSOPHY

ワクワクとふむふむを追求する集団
不可能を可能にする集団

アジェンダ

- 1 | 概要
- 2 | 機能紹介
- 3 | 運用例

アジェンダ

1 | 概要

2 | 機能紹介

3 | 運用例

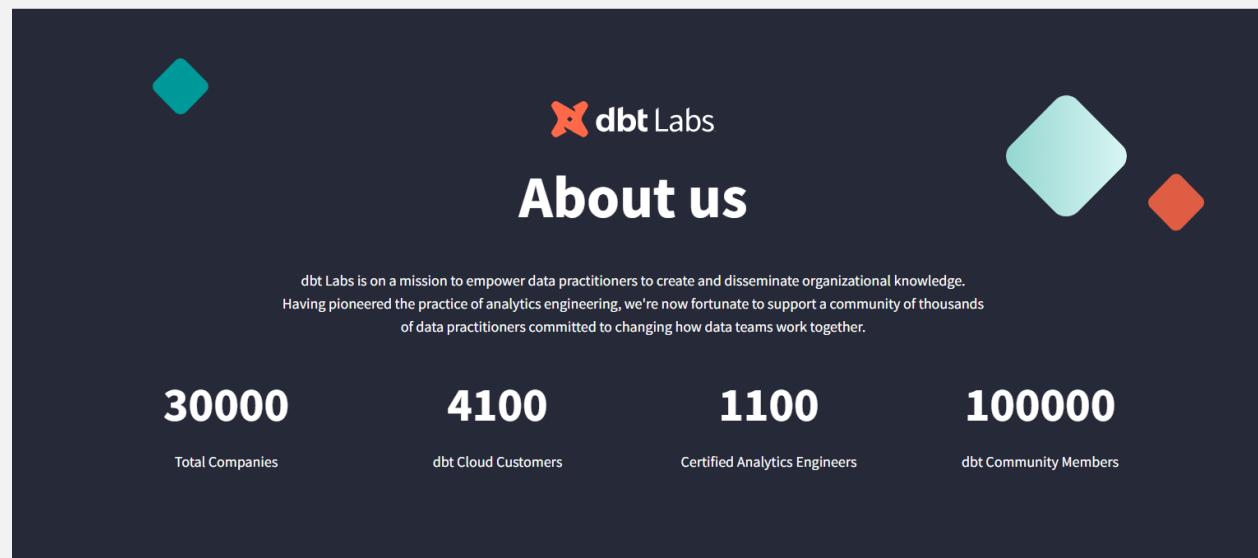
dbtとは

dbt (data build tool) は、2016年に設立されたFishtown Analytics (現dbt Labs) がデータアナリスト向けに開発したデータ変換プロセスを担うツールです。

2022年2月にシリーズDとして42億ドルの評価額で2億2200万ドルの資金を調達[1]し、なおも新機能のリリースや各データスタックとの連携を強めています。

■ dbtの役割

- ・ データ開発のライフサイクルを加速
- ・ エンジニアの生産性を向上
- ・ データの品質を維持
- ・ スケールと複雑性に対応



<https://www.getdbt.com/dbt-labs/about-us>

成長

dbt導入事例は、2016年から毎年200%増加。

要因として、自社のデータ基盤をSnowflakeやBigQuery、Databricks、Redshiftなどのクラウドベースのデータプラットフォームに移行し、クラウド移行後のデータ変換プラットフォームとして、dbtが採用されていると考えられます。

■ dbtのこれまで

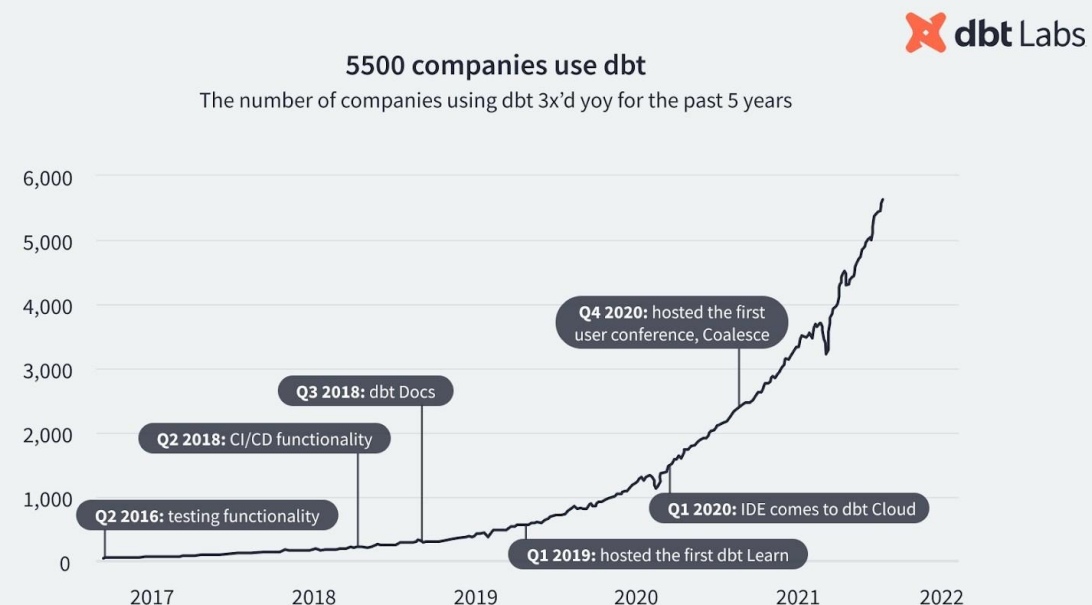
Q2 2016：テスト機能の追加

Q2 2018：CI/CD機能の追加

Q3 2018：ドキュメント機能の追加

Q1 2020：dbt Cloudのリリース

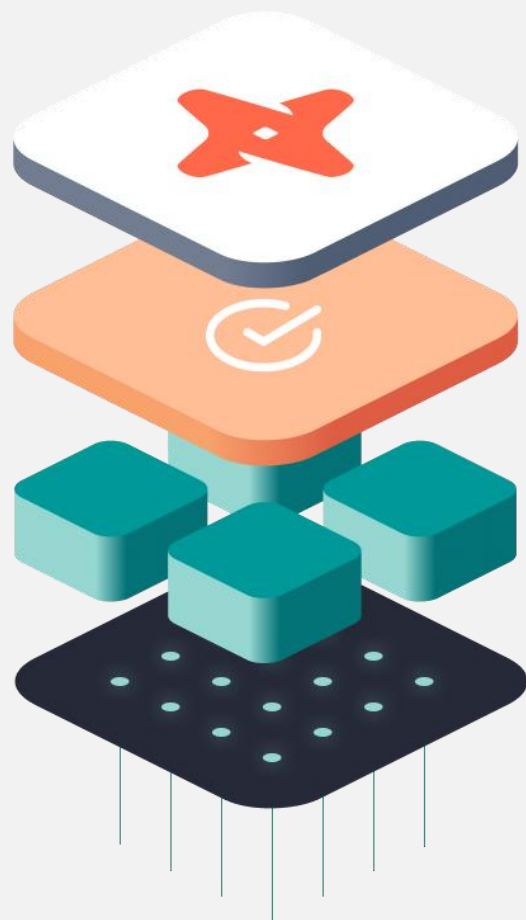
Q4 2020：初主催のユーザカンファレンス



<https://www.prnewswire.com/news-releases/fishtown-analytics-rebrands-as-dbt-labs-closes-150m-to-develop-open-source-analytics-engineering-software-301322673.html>

構成

dbtは、SQLファーストの変換ワークフローで、ソフトウェアエンジニアリングのベストプラクティスに従って、DWHに取り込んだソースデータから迅速かつ安全に分析データを構築しデータを提供します。



3. バージョン管理とCI/CD :

GitHub, GitLab, Gitサービスを使用してファイルのバージョン管理が可能
Gitフローを自動化し品質を維持したまま安全にデプロイできることで生産性が向上

2. テストとドキュメント :

変換処理が実行されたデータに対してデータの状態や整合性をチェックして反映
dbtが生成したメタデータで簡易ドキュメントを自動で作成。データ関係者へ共有

1. 開発 :

データ変換処理を.sqlまたは.pyファイルで記述
動的に実行環境を切り替え迅速かつ安全にデプロイを行い、開発スピードを加速

<https://www.getdbt.com/product/what-is-dbt>

OSSとSaaS

dbtには、OSS版の「dbt Core」とSaaS版の「dbt Cloud」があります。

	dbt Core	dbt Cloud
費用(月)	0	<u>0</u> / <u>100\$$\times$user</u> / <u>300\$$\times$user</u>
サービス	OSS	SaaS
作業場所	コードエディタ	Webブラウザ
機能	コマンドライン (CLI)	・ 統合開発環境 (CLI・Git・プレビューなど) ・ スケジューラー ・ 通知 ・ ドキュメント (Explore) ・ セキュリティ ・ CI/CD ・ チャットサポート

導入企業

国内



海外

CONDÉ NAST



sunrun

jetBlue

deputy



whatnot



導入効果（運用）

16h

プロジェクト1件あたりの
作業時間を短縮

CONDÉ NAST

30%

リネージ、データの民主化、
合理化されたワークフロー
により納品スピードが向上

 pepperstone

50%

複雑なデータ問題を診断し
解決するためのチケット数
を削減

SUNRUN

<https://www.getdbt.com/product/data-modernization>
<https://www.getdbt.com/product/governance>

導入効果（コスト）

0\$

dbt移行後のTransformにおけるランニングコストが0に

jetBlue

20%

コンピューティングコストの削減

deputy

1/10

メンテナンスコストが1/10に削減

whatnot

<https://www.getdbt.com/product/data-modernization>
<https://www.getdbt.com/product/cost-optimization>

導入効果（生産性）

55

dbt導入後、データ基盤に
参加したビジネスユーザ数



5x

過去データから配達時間を
5倍短縮



30%

データエンジニアのセルフ
サービスが増加

CONDÉ NAST

<https://www.getdbt.com/product/data-modernization>
<https://www.getdbt.com/product/data-vault>

アジェンダ

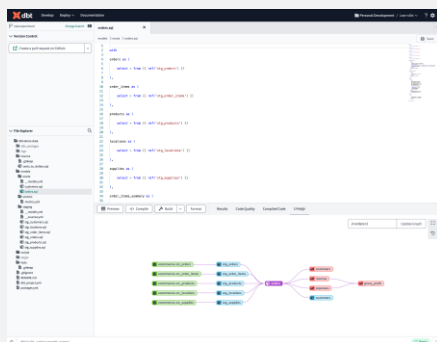
1 | 概要

2 | 機能紹介

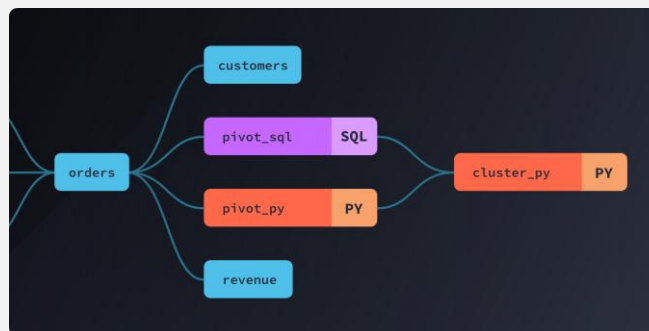
3 | 運用例

dbtの特徴

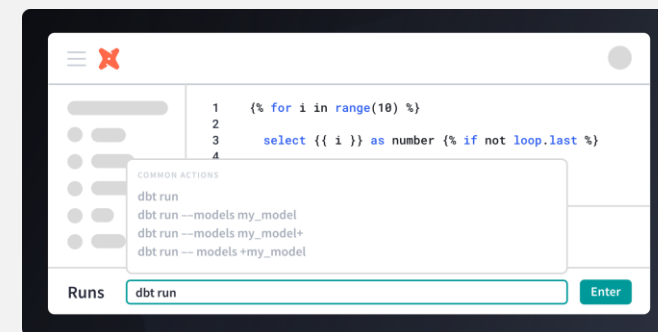
開発



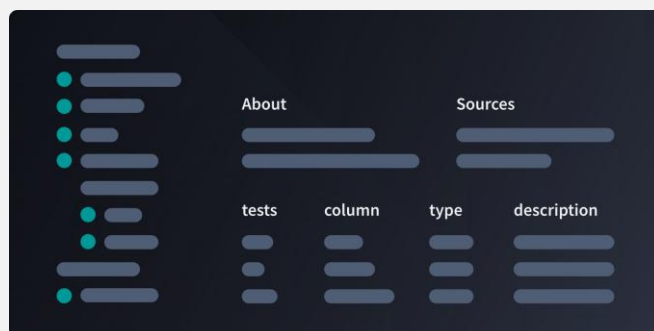
リネージ



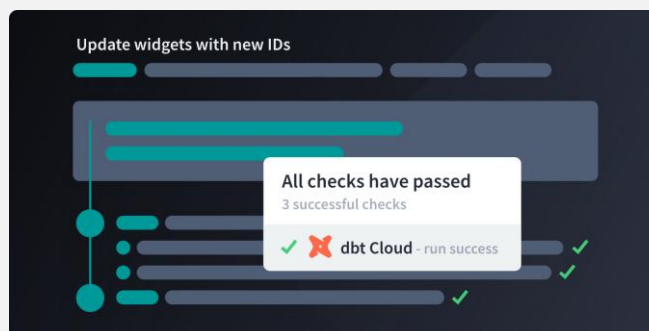
コマンド



ドキュメント



CI/CD

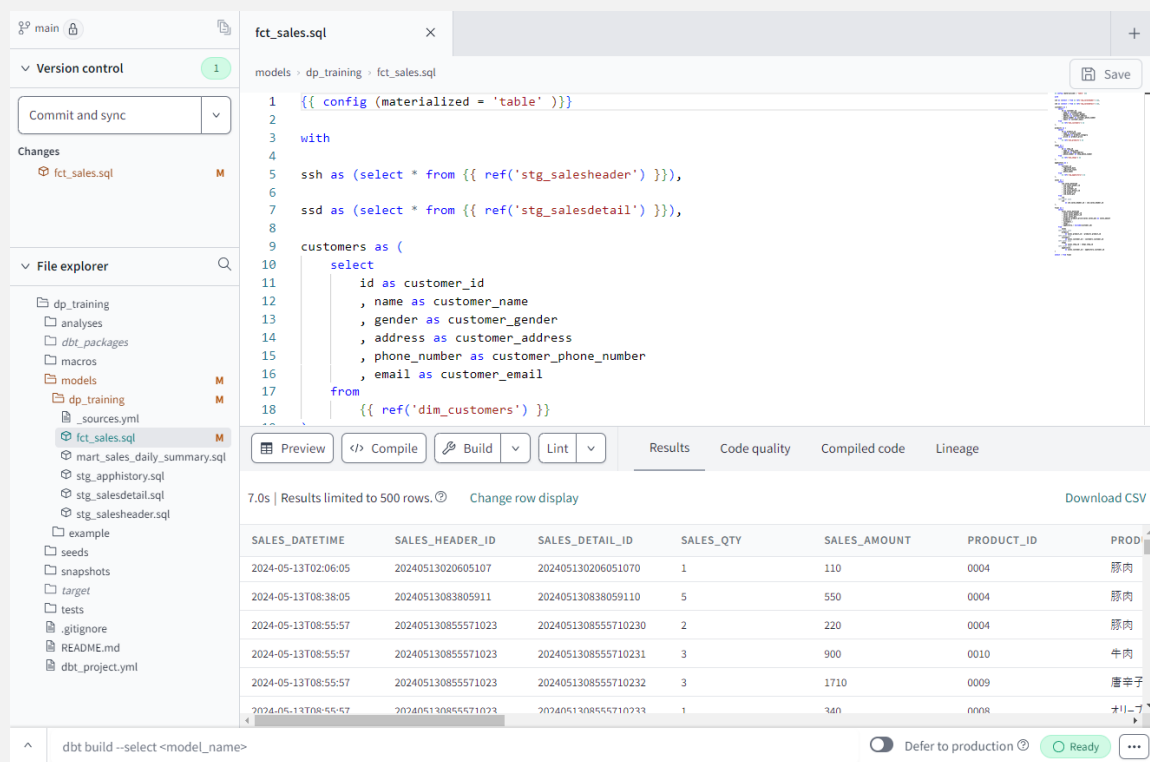


テスト



<https://www.getdbt.com/product/what-is-dbt>

開発



特徴

■ SQLとYAMLのみでパイプラインを構成

標準SQLとdbt設定、YAMLによるデータ定義を行うだけで、dbtがデータパイプラインを自動的に構築。従来行っていた依存関係の考慮やタスク管理作業は不要。

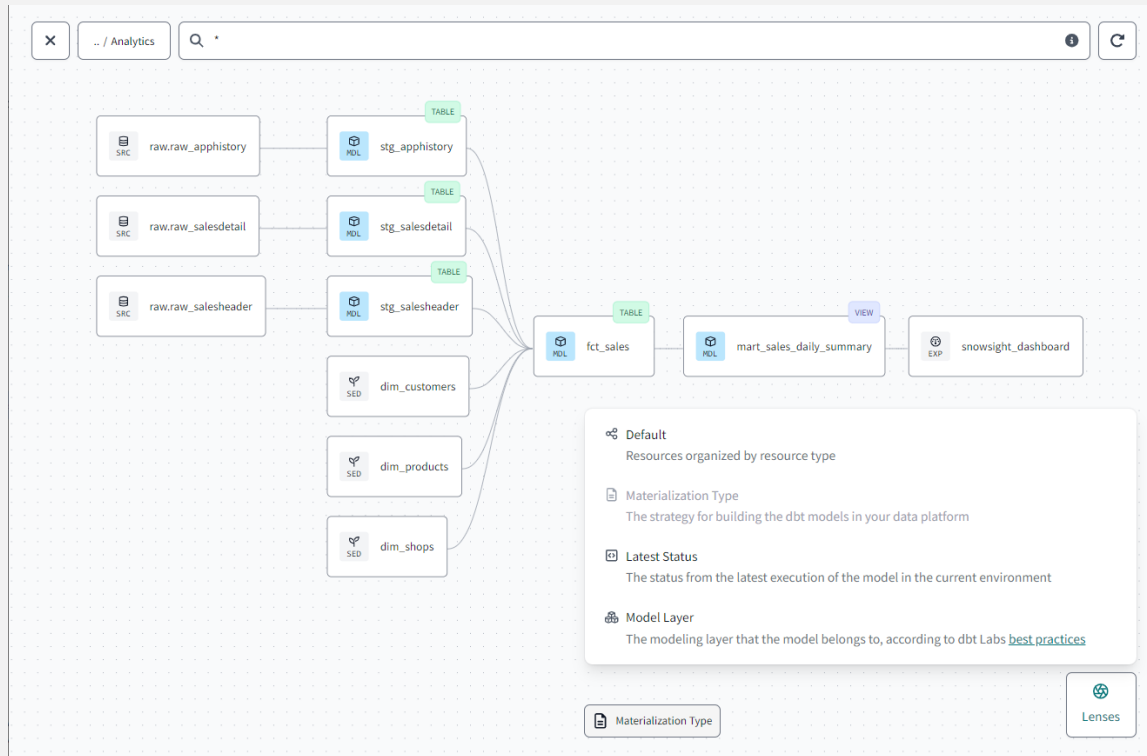
■ 開発をサポートする機能

- ・ プレビュー：記述したSQLの実行結果を確認
- ・ コンパイル：変換処理のコンパイル結果を確認
- ・ リンター：記述されたSQLのフォーマットチェック+自動修正
- ・ リネージ：開発しているモデルの依存関係をグラフ化
- ・ その他：マルチカーソルやコード補完機能も充実

■ Git機能

- ・ ブランチ作成：作業用のブランチを作成
- ・ コミット：dbt Cloud上での変更を反映
- ・ プルリク作成：変更に関するプルリクエストを作成

リネージ



特徴

■ データ同士の依存関係を自動でグラフ化

dbtがSQLを解析し、データ同士の依存関係を自動でグラフ化

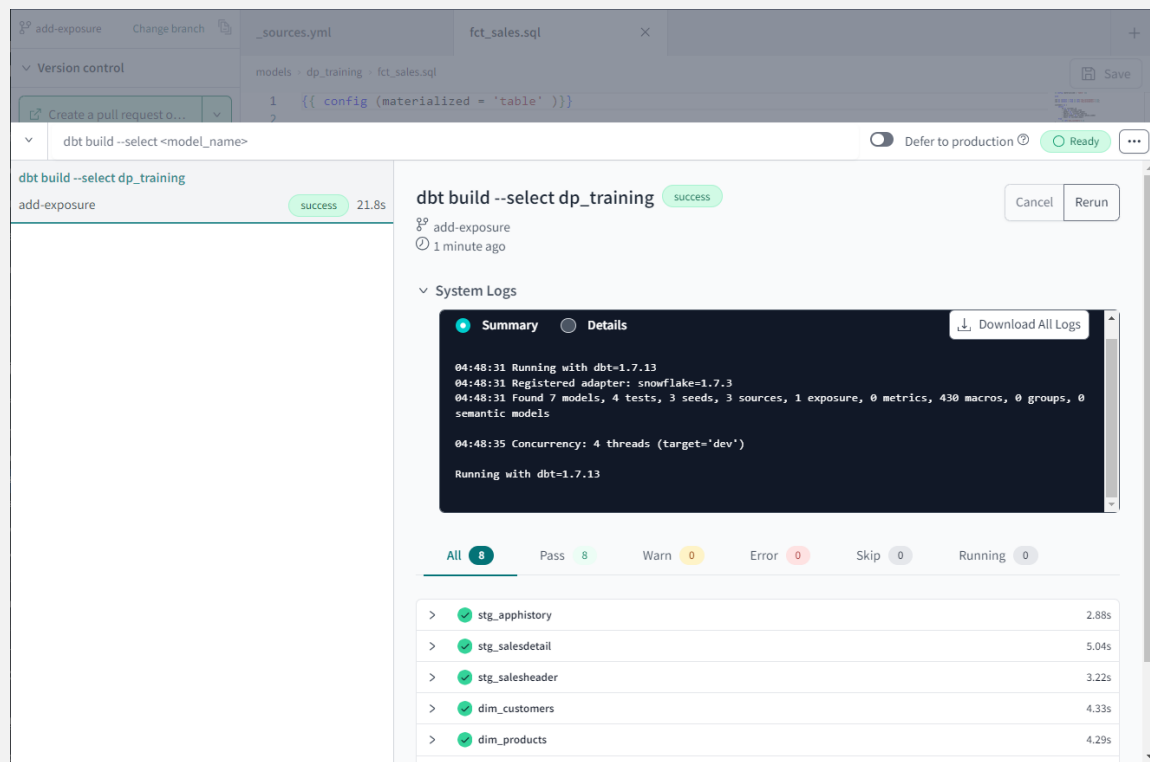
■ ソースデータや下流サービスもグラフに表示可能

dbt外に存在するソーステーブルや、ダッシュボードといったデータが参照されているサービスもリネージ上に乗せることが可能。処理の修正や障害時の影響範囲を早急に特定できるようになります。

■ フィルタ機能で参照範囲を制限

一部のパイプラインやフォルダ内のリネージのみ表示したい場合は、上部のメニューバーから指定が可能

コマンド



特徴

■ 変換処理の実行

dbtのコマンドを実行することで、開発作業や本番運用でモデルの変換処理をDWHに対して行うことができます。

■ 実行範囲の指定

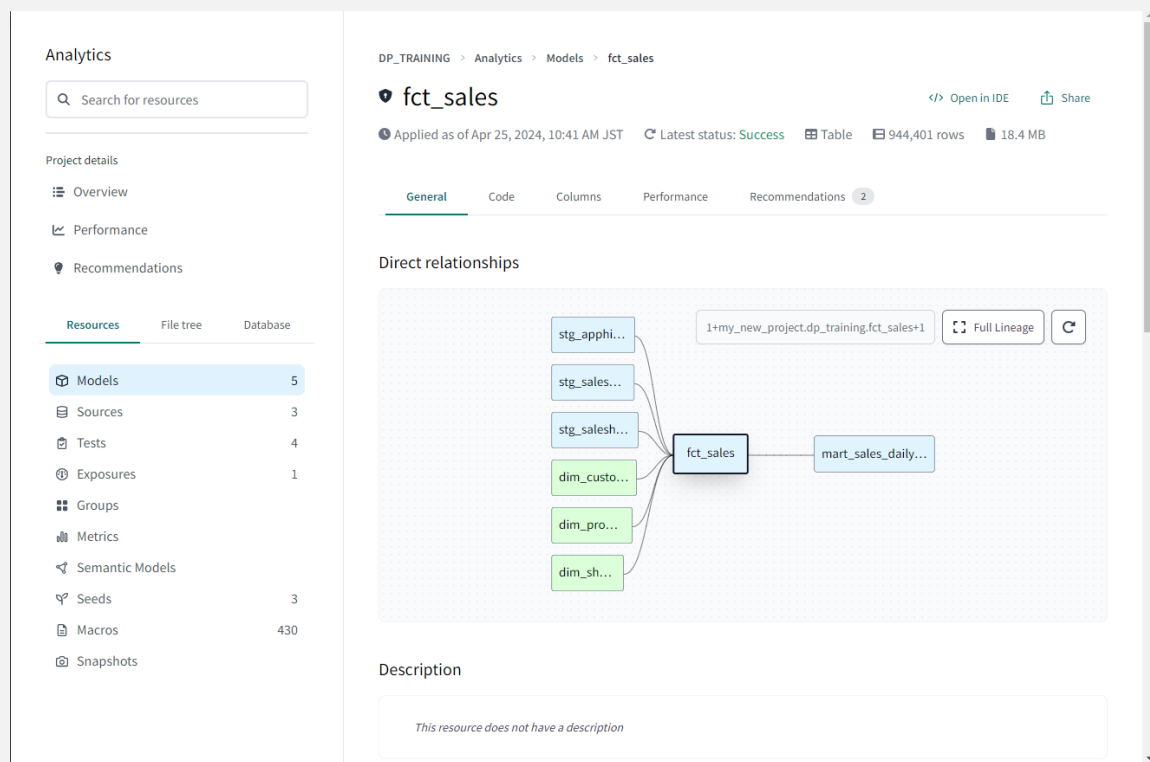
「あるモデルの下流や上流」、「フォルダ内のモデル」、「モデルに設定されたタグ指定」など、実行コマンドにオプションを設定することで、特定のパイプラインのみ実行することも可能です。

※今後、TableauやPowerBIのダッシュボードも更新可能

■ コマンド一覧

- ・ run：指定のモデルの変換処理を実行
- ・ test：指定のモデルのテストを実行
- ・ seed：指定のシード(CSV)をDWHにテーブル作成
- ・ build：seed/run/testを実行
- ・ docs：ドキュメントを生成
- ・ clone：指定のテーブル、スキーマなどをCLONE生成

ドキュメント（Explore）



特徴

■ dbt管理のメタデータからドキュメントを自動生成

コマンド実行時にdbtが管理・更新しているメタデータファイルを元に、WebUIで参照可能な簡易ドキュメントが自動で生成・更新されます。

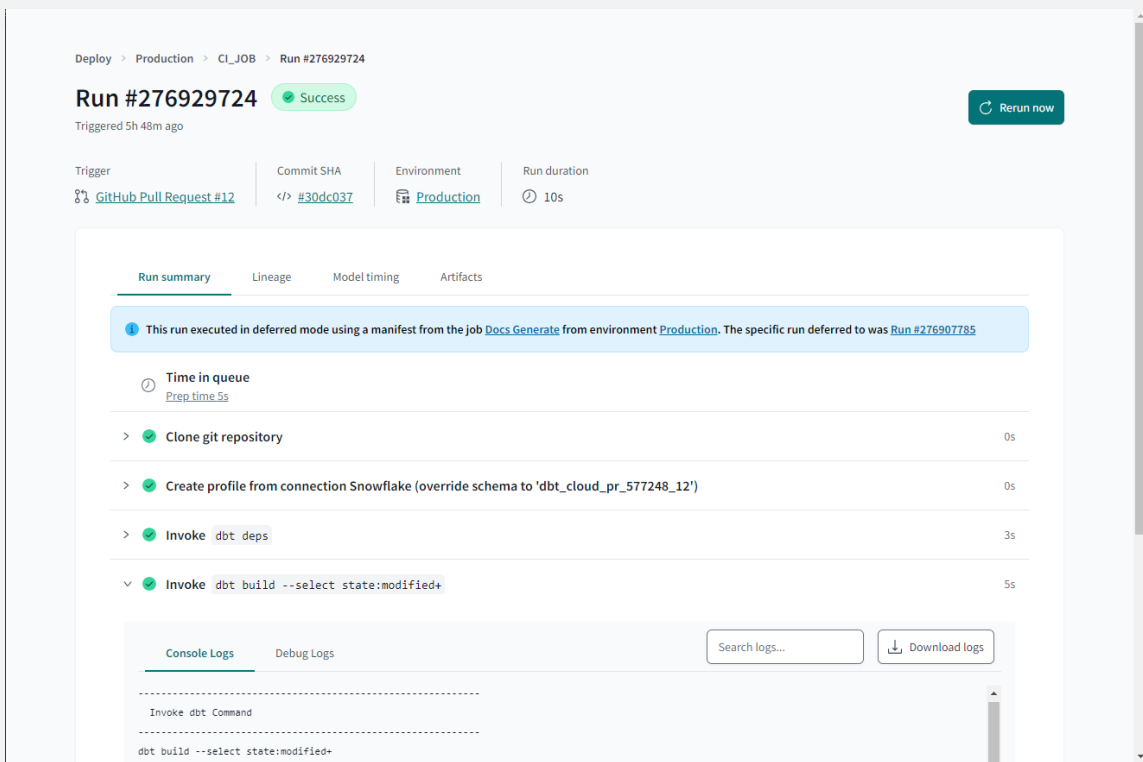
■ 部署や企業内でドキュメントを共有可能

データ基盤にかかわるビジネスサイドやエンドユーザを招待することで、最新のデータ内容を確認・把握することでデータ活用を促進します。

■ 管理情報

- ・ リネージ（オブジェクト・カラム）
- ・ 基礎情報（データ量・データサイズ・依存元依存先）
- ・ コード（変換クエリ）
- ・ カラム（物理名・論理名・データ型・制約）
- ・ パフォーマンス（更新時間・更新回数・ステータス）
- ・ レコメンド

CI/CD



特徴

■ レビュー作業を自動化（CI）

Gitを用いたバージョン管理を行うチーム開発の中で頻繁に発生するプルリクエストのレビュー作業。その中でも多くの工数を消費しながらもデプロイエラーの原因になっていた、手動によるレビュー作業を自動化し、工数削減と品質向上に貢献します。

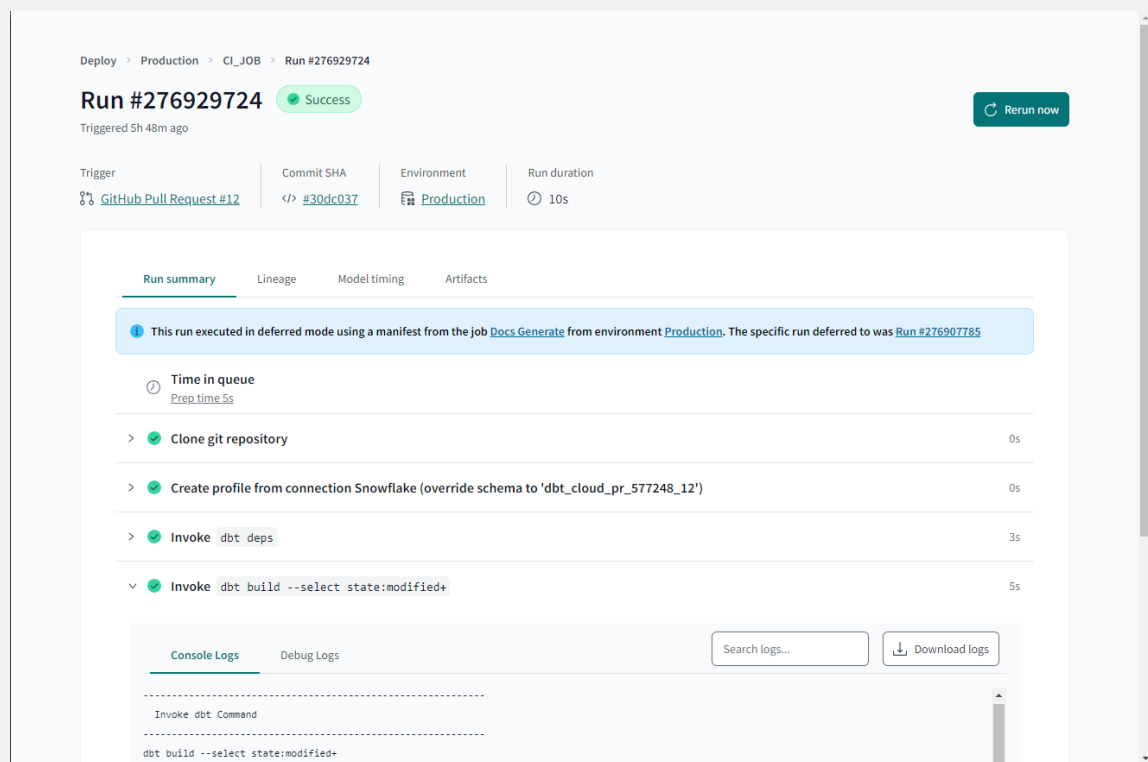
■ デプロイ作業を自動化（CD）

mainブランチへのマージ後、本番環境への反映は手動で行う必要があり、この作業も同様にヒューマンエラーの原因となっていました。mainブランチにマージされた内容のみを本番環境へ自動で反映することで、人為的なミスを最小限に抑えられます。

■ CI/CDを導入するメリット

- ・ 手動での作業を自動化し工数を削減
- ・ ヒューマンエラーの発生を最小限に抑える
- ・ エンジニアの精神状態が安定

テスト



特徴

■ 意図しないデータの混入を防ぐ

ユニークキーのような重複を許さないカラムのデータ。売上のような負の値が想定されないデータ。そうしたデータが生成された場合、あらかじめテストを設定していれば更新を停止し、後続の更新もスキップできます。

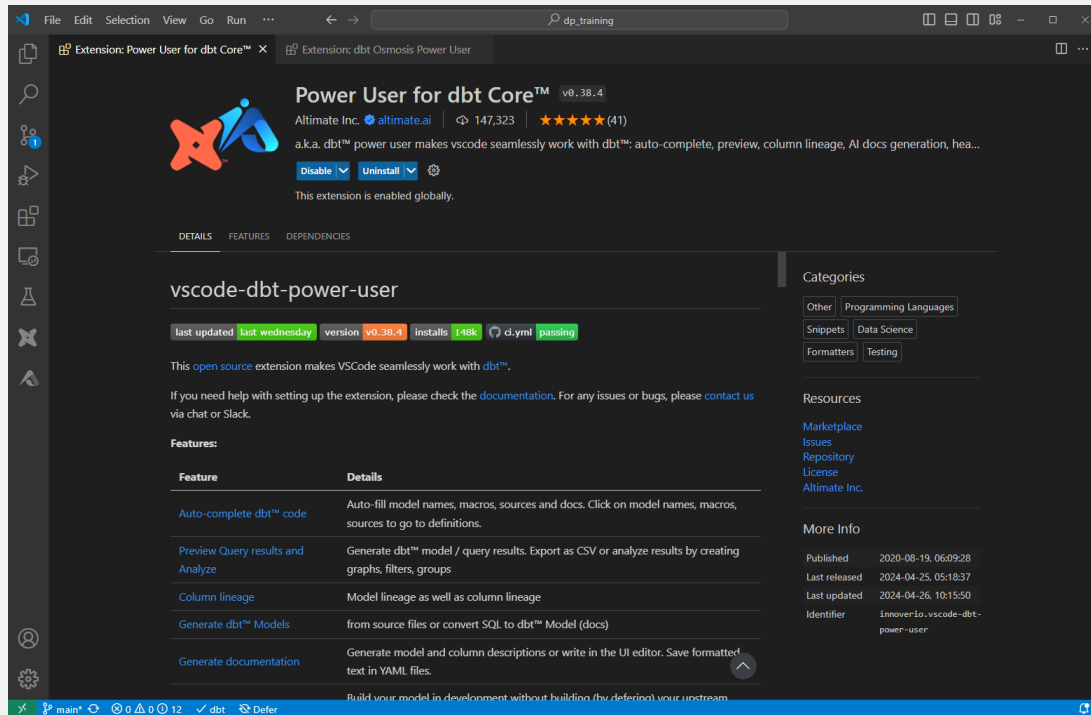
■ オリジナルのテスト

テストはSQLを記述して作成することも可能で、ビジネス特有なデータに対するテストを実現します。

■ データテストと単体テスト

従来、生成されるデータに対するテスト「データテスト」は可能でしたが、新たにモデルのロジックに対するテスト「単体テスト」が登場しました。データだけでなく、開発者が記述するコードに対してテストを適用することで、作業の効率が上がりコードの信頼性も向上します。

拡張機能



特徴

■ dbt Power User

VSCode上でdbtCloudと同等のdbt専用UIが使用可能。
dbt Cloudにはないメタデータ入力用UIなども搭載。

■ dbt Osmosis

モデルのメタデータ（ディスクリプションやデータ型）の登録作業を自動化し、作業工数を削減しつつデータ基盤内のメタデータを拡充。

dbtの特徴

開発

SQLで記述されたデータ処理をDMLに変換しDWHへ実行。環境構築や詳細な設定はYAMLで記述し管理する。

<https://docs.getdbt.com/docs/cloud/about-develop-dbt>

リネージ

データ依存関係をdbtが自動でグラフ化し可視化します。エラーや変更作業がどのオブジェクトに影響するかを把握できます。

<https://docs.getdbt.com/terms/data-lineage>

コマンド

変換プロセスはdbtコマンドによって実行されます。また、実行順序はリネージに従って行われ、オプションにより詳細な実行制御も可能になります。

<https://docs.getdbt.com/reference/dbt-commands>

ドキュメント

定期的に生成されるメタデータからユーザの介入なしに簡易的なドキュメントが生成されます。リネージと合わせて企業のデータ関係者と共有することができます。

<https://docs.getdbt.com/docs/collaborate/documentation>

CI/CD

新しい要望やバグの修正などで頻繁に繰り返される開発フローの一部を自動化し、データ品質と生産性を向上させます。

<https://docs.getdbt.com/guides/custom-cicd-pipelines>

テスト

意図しないデータが含まれる場合、データの更新をスキップしてデータ基盤内のデータを保護。ロジックに対するテストも可。

<https://docs.getdbt.com/docs/build/data-tests>

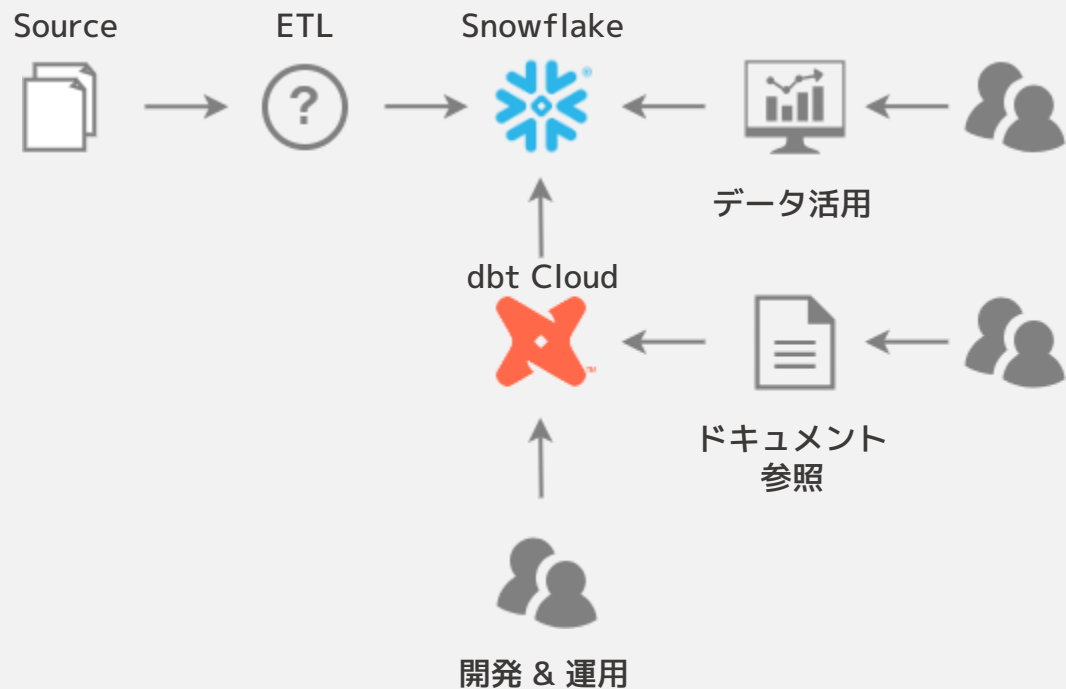
アジェンダ

1 | 概要

2 | 機能紹介

3 | 運用例

dbt Cloud



特徴

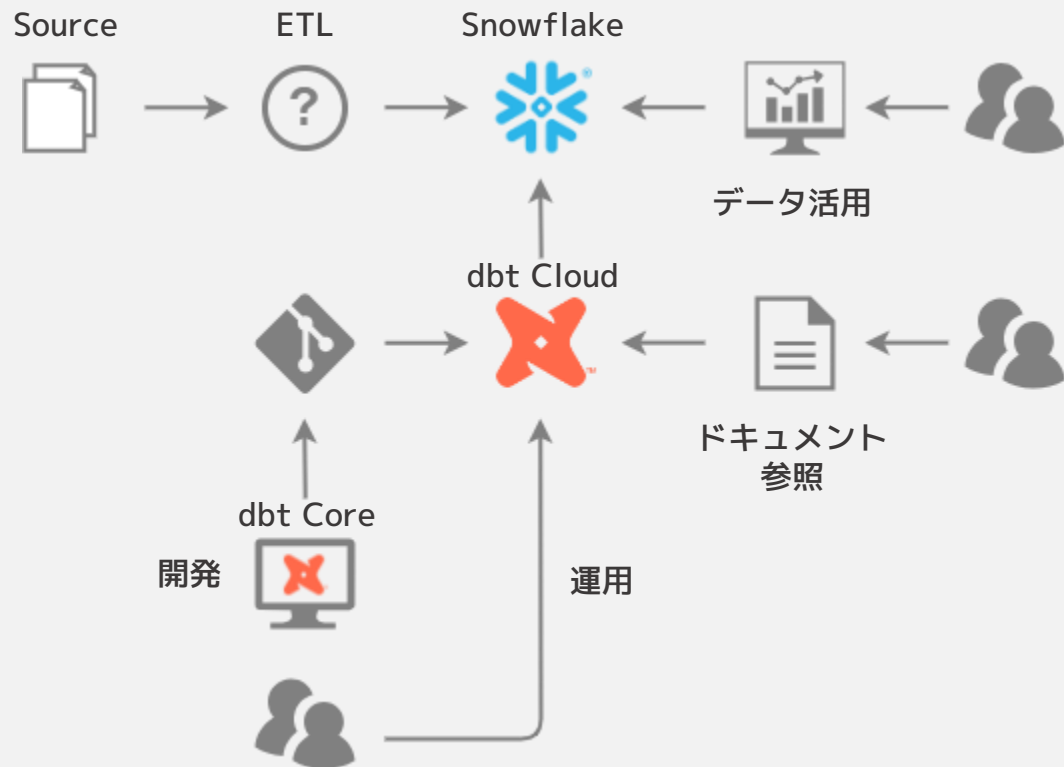
■ メリット

- サーバー管理不要
- dbt Cloud専用の機能を利用可能

■ デメリット

- コストが高い（開発ユーザ×100\$/month）
- VS Code拡張機能が使用できない

dbt Cloud + dbt Core



特徴

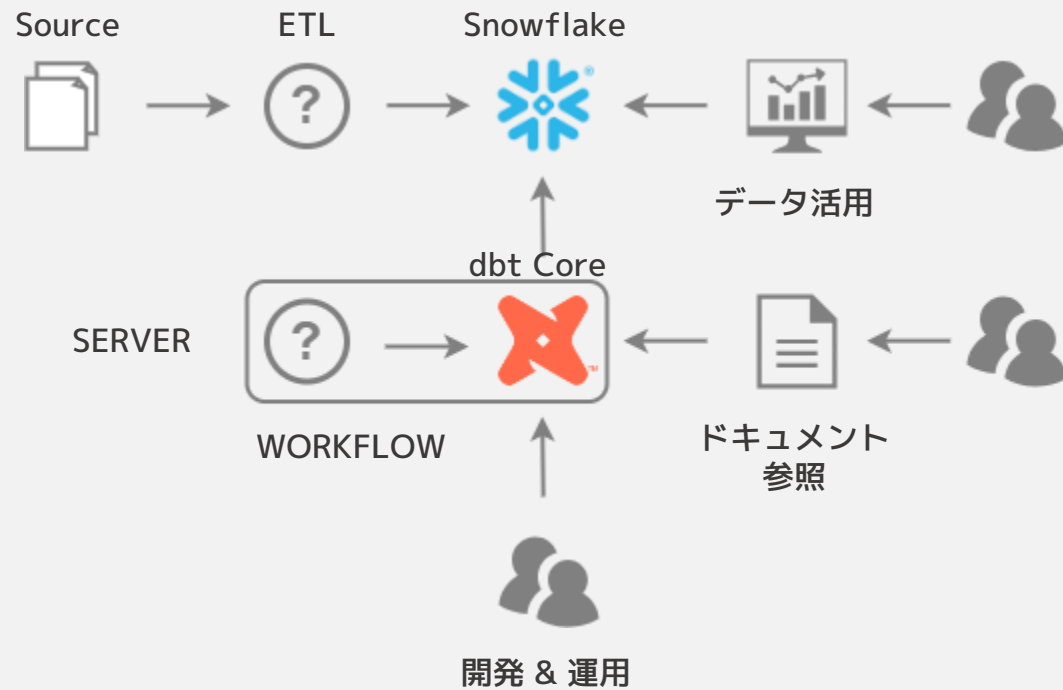
■ メリット

- サーバー管理不要
- 開発ユーザが増えてもコストは変わらない
- VS Code拡張機能が使用可能

■ デメリット

- ローカルPCで開発環境を用意する必要がある
- 開発でdbt Cloudの機能は使用できない

dbt Core + ワークフローツール



特徴

■ メリット

- 既存クラウド環境があればコスト不要
- 独自のスクリプトを使用可能

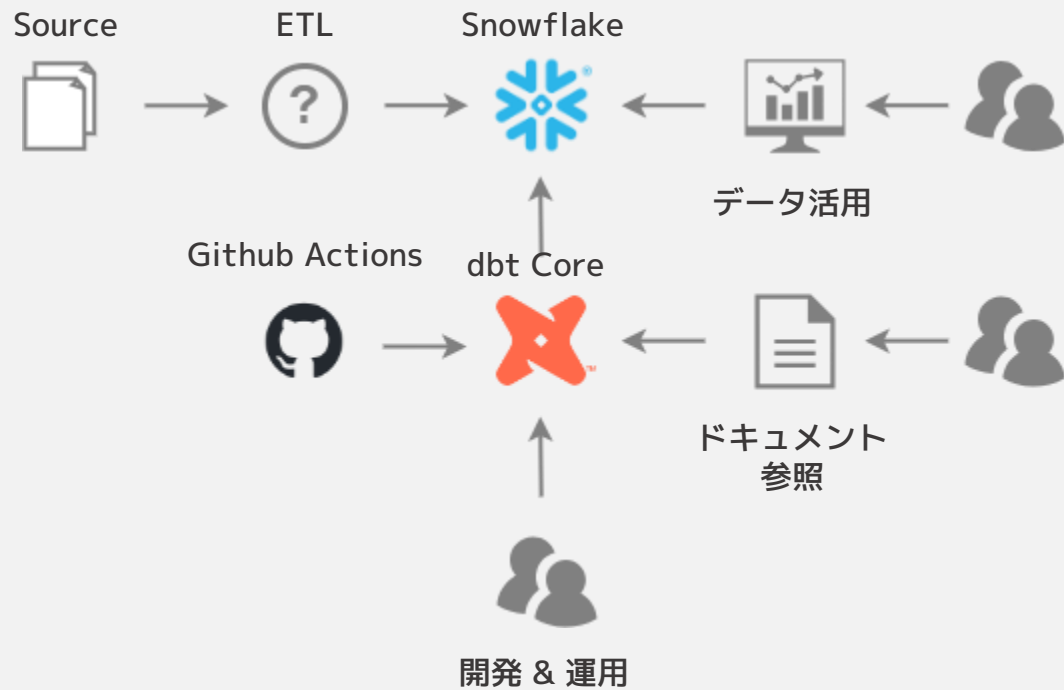
■ デメリット

- サーバ管理が発生する
- 学習コストが高い

※ワークフローツール

- Airflow
- Dagster
- TROCCO (ワークフロー機能)

dbt Core + Github Actions



特徴

■ メリット

- サーバー管理不要
- 独自のスクリプトを使用可能

■ デメリット

- 学習コストが高い
- Github Actionsが従量課金のためコスト増の可能性あり

運用例

dbtを導入する場合、いずれかの方法でホスティングして運用する必要があります。(2024年4月時点)

	dbt Core + ECS	dbt Core + SPCS*	dbt Cloud	dbt Core + dbt Cloud	dbt Core + Github Actions
サーバ管理	有	有	無	無	無
dbt拡張機能	使用可	使用可	使用不可	使用可	使用可
コスト (month)	32\$+α ※0.043\$/hour	223\$ ※0.3\$/hour	100\$×ユーザ数	100\$	従量課金

*Snowpark Container Services

ビッグデータを活用し、
より豊かな社会を創る

INSIGHT LAB

〒163-0533

東京都新宿区西新宿1-26-2 新宿野村ビル 33F

TEL 03-5909-1320

FAX 03-5909-1321

URL <https://insight-lab.co.jp>



Appendix



参考

[1] <https://archive.md/20220511115331/https://www.forbes.com/sites/kenrickcai/2022/02/24/dbt-labs-series-d-4-billion-less-than-planned/?sh=26c7c1cc67c3>

コスト

ECS forgate : <https://aws.amazon.com/jp/fargate/pricing/>

SPCS : <https://www.snowflake.com/legal-files/CreditConsumptionTable.pdf>

dbt Cloud : <https://www.getdbt.com/pricing>

Github Actions : <https://docs.github.com/ja/billing/managing-billing-for-github-actions/about-billing-for-github-actions>