データサイエンス

すいすい会

第22回
「データ分析基盤どうあるべき？
～データエンジニアリングの民主化
の方向性～」

GRI
BIGDATA SCIENCE
& RESEARCH

ForecastFlow

すいすい会の紹介

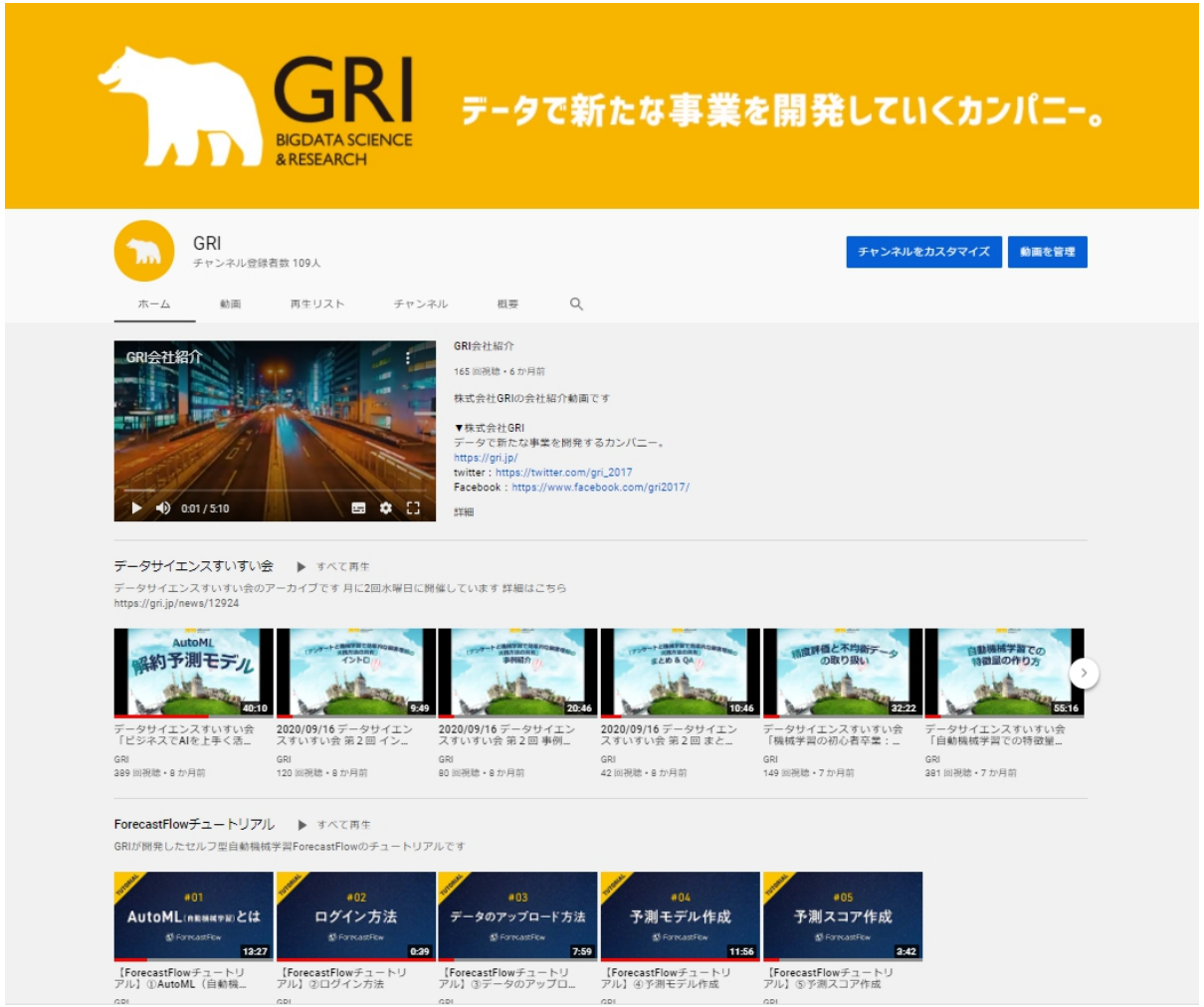
■データサイエンスに関する知見を共有しあい
　みんなで実践的な理解度を高めたい

■資料：　GRIホームページ　https://gri.jp/news/12924

■Slack　ForecastFlowチャンネル　https://join.slack.com/t/forecastflowusers/shared_invite/enQtNTgyMjcxOTg0NzcxLTBkOWEzYWMwNDJmNTkyMDQzYmIxYWU0YWI4ZmU3ZDU0ZTMxNDUwODAxMWFmYmU1YjJiZGI0MjRhYWYyYTNlZTQ

過去のすいすい会の動画（YouTube）

https://www.youtube.com/c/GRIinc



ナビゲータの背景

古幡征史
株式会社GRI　取締役

- 株式会社GRI 取締役
- Ph.D in Computer Science
- GRIにて50以上のAI, BI, 分析基盤構築プロジェクトをリード
- KPMGコンサルティング、University of Southern California、ドワンゴを経て、2016年9月より現職

©GRI Inc.

本日のアジェンダ

不明な所も多いので、突っ込んでいただくと幸いです

- データ分析基盤どうあるべき？
  - AIツールとBIツールの進化＆データ分析基盤の必要性
  - 分析基盤の基礎知識
  - おすすめのデータ分析基盤
    - 列指向DB, ELTツール, 自動機械学習ツール, BIツール
  - 大規模データの扱いの悩みのポイント

- ELT（Matillion）ツールのデモ

©GRI Inc.

データサイエンス

すいすい会

第22回
「データ分析基盤どうあるべき？
～データエンジニアリングの民主化
の方向性～」

GRI
BIGDATA SCIENCE
& RESEARCH

ForecastFlow

データ分析の民主化とデータ分析基盤の必要性

- AIやBIのツールが発達し「民主化」が進んでいる
  - 誰でもデータサイエンティスト
  - 誰でもビジネスアナリスト

AI

ForecastFlow　DataRobot　Prediction One

BI

tableau　Power BI　Qlik

データ分析の民主化とデータ分析基盤の必要性

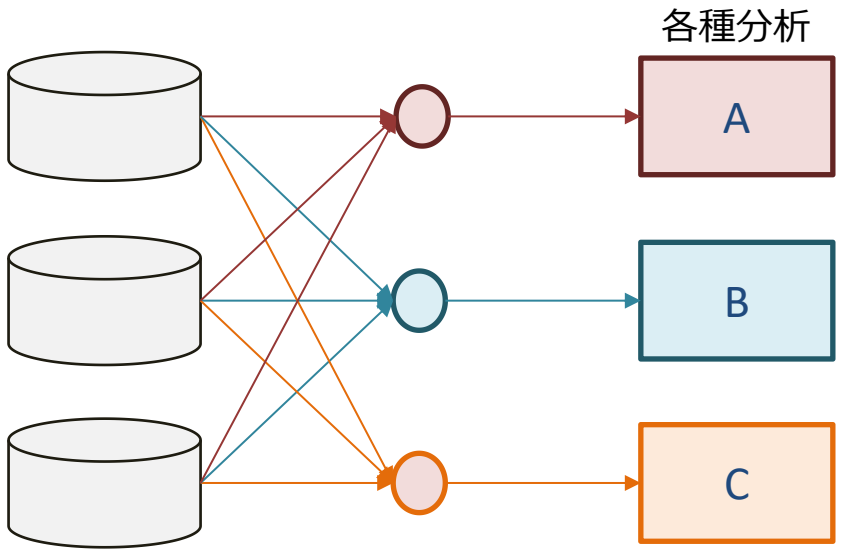
- AIやBIのツールが使えるようになってもデータの準備（前処理）に多くの時間が必要

- 優秀なデータ・サイエンティストの時間の使い方
  - ビジネス理解：　50%
  - 前処理：　30%　← 初心者は前処理に80%の時間を浪費
  - 分析：　20%

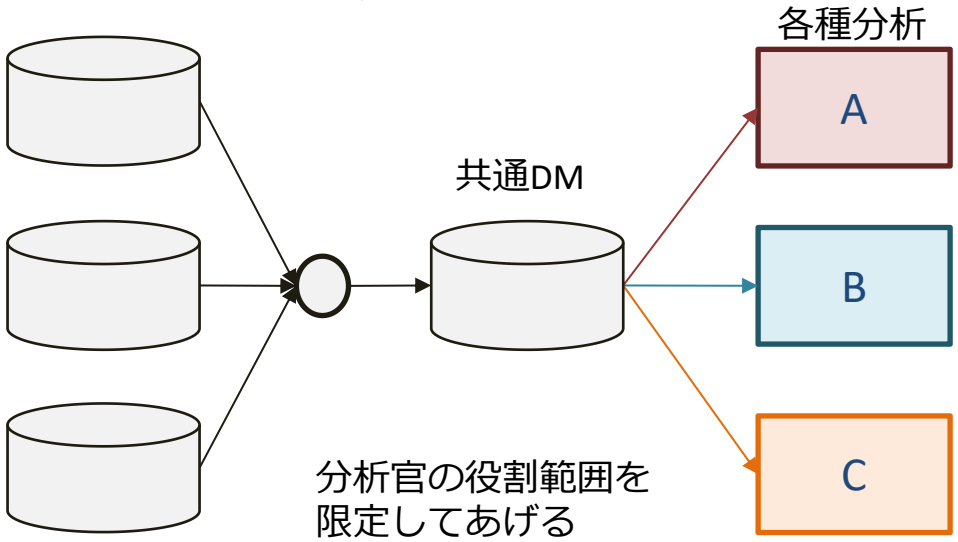
データエンジニアがいれば良いじゃないか？

- 前処理は多くの人が不得意とする部分
- 巨大組織ではデータパイプラインが乱立しがち
- 一人のデータエンジニアが分析しやすい汎用的なデータマート（大福帳）を用意したら、数人分のビジネスアナリストの効率がアップする

分析テーマA, B, Cをバラバラにデータパイプラインを作る

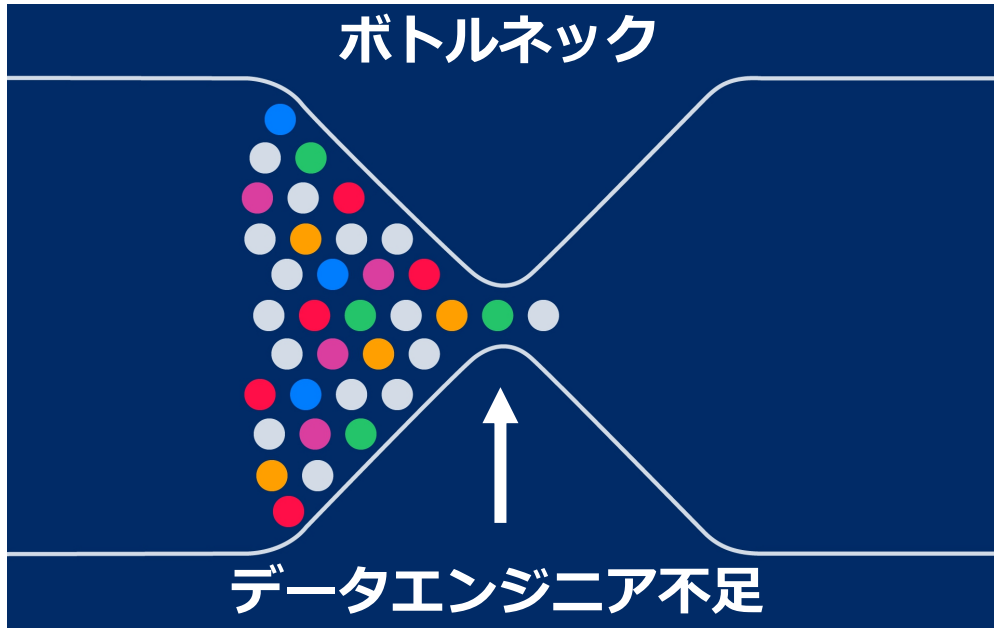


分析テーマA, B, Cに対し共通データマートを用意する



良いデータエンジニアの獲得は難しい

- しかしながら、データエンジニアは社内で枯渇気味なため、着手待ちの時間が長い
- 民主化に早めに着手した企業は、データエンジニアの重要性を理解し、囲い込みを行っている
- 転職市場から良質なデータエンジニアの確保は、データサイエンティストより困難



©GRI Inc.

では、どうすべきか？

- ハイスペックなデータエンジニア１人分くらいの金額の費用に相当するデータ分析基盤をご紹介

- その前に分析基盤に繋がる基礎知識を共有
  - （普通のエンジニアがデータエンジニアで失敗する理由）

©GRI Inc.

分析基盤の基礎知識

©GRI Inc.

分析基盤の話の前に実行系システムの基礎知識（行指向のDB）

- 実行系では、登録・変更・削除が頻繁に繰り返される前提
- この処理を効率的に行うため典型的なデータベース（Oracle, SQL Server, MySQL）では、行指向単位で管理
- データを正規化して、メモリなどで無駄なことを割ける（冗長でなくする）
- データは追加、削除、更新を一行単位で処理

取引ヘッダー

取引ID	取引日時	店舗ID	ユーザID	売上	小計	税
9001	2021/3/1 10:10:11	S01	10001	660	600	60
9002	2021/3/1 10:10:12	S01	10002	1100	1000	100
9003	2021/3/1 10:10:12	S01	10003	550	500	50

取引明細

取引ID	明細ID	商品ID	個数	単価
9001	001	P01	1	100
9001	002	P02	2	200
9001	003	P03	1	100

同じことを、いっぱい繰り返さないようにテーブルを分けて管理

©GRI Inc.

分析する人のうれしいテーブル（一枚表や大福帳）

- 分析する時には、一つのテーブルになっている状態で、分析しやすい形になっている方が、すぐに分析に入れる
- 縦にも横にも長いテーブル

分析用の売上データ

取引ID	取引日時	店舗ID	店舗名	ユーザID	性別	売上	小計	税	明細ID	商品ID	商品名	個数	単価
9001	2021/3/1 10:10:11	S01	銀座店	10001	男性	660	600	60	001	P01	鉛筆	1	100
9001	2021/3/1 10:10:11	S01	銀座店	10001	男性	660	600	60	002	P02	ペン	2	200
9001	2021/3/1 10:10:11	S01	銀座店	10001	男性	660	600	60	003	P03	消しゴム	1	100
9002	2021/3/1 10:10:12	S01	銀座店	10002	男性	1100	1000	100	001	P04	本	1	1000
9003	2021/3/1 10:10:12	S01	銀座店	10003	男性	550	500	50	001	P05	日記帳	1	500

©GRI Inc.

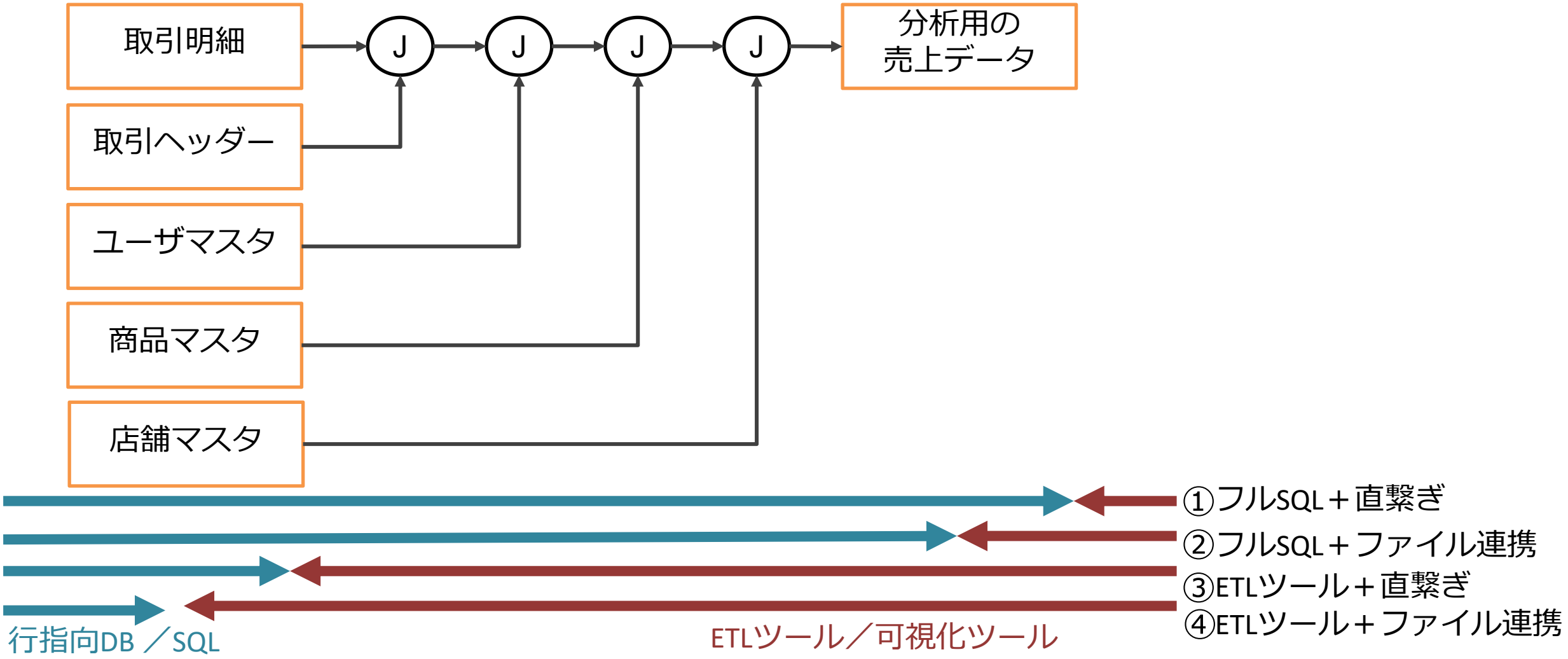
実行系システムからすると一枚表や大福帳は非常識

- 実行系システムに慣れている人に一枚表や大福帳のテーブルを持っておいてください、と言うと、非常識な人として扱われます
- そんなことをすると、実行系システムの性能を維持するのが困難なため

©GRI Inc.

前処理のデータフローの範囲例

- よくあるのが、行指向DBでのSQLやETLツールを使って前処理を記述



SQLで集計表を作るときのよくある課題

- 属人化してしまう
  - 誰かの記述したSQLを他の人が見ても理解が困難
  - ルールの変更時に変更対応すべき箇所が分かりにくい（影響範囲が分からず修正困難）
- SQLを書ける人が少なく、依頼が面倒で、エンジニアの着手から作業完成の待ち時間も多い
- エンジニアはデータの肌感覚がないため、データが正しいか分からないことが多く（検算できない）、手戻りが多い
- 行指向DBに格納してしまい、分析作業でクエリが返るまで遅い

```
SET NUMERIC_ROUNDABORT OFF;
SET ANSI_PADDING, ANSI_WARNINGS, CONCAT_NULL_YIELDS_NULL, ARITHA
   QUOTED_IDENTIFIER, ANSI_NULLS ON;

--従業員テーブルの作成とデータ挿入
create table EMPLOYEE_V (
EMPLOYEE_ID nchar(5)
, NAME nvarchar(20) NOT NULL
, KANANAME nvarchar(20)
, BATHDAY DATE
, SEX nchar(1)
, JOIN_DATE DATE
, POST_CODE nchar(7)
, ADDRESS nvarchar(100)
, TEL nvarchar(11)
)
insert into [dbo].[EMPLOYEE_V] values('T0001','太郎','タロウ','1
insert into [dbo].[EMPLOYEE_V] values('T0002','花子','ハナコ','1
insert into [dbo].[EMPLOYEE_V] values('T0003','次郎','ジロウ','1
insert into [dbo].[EMPLOYEE_V] values('T0004','三郎','サブロウ',


--部門テーブルの作成とデータ挿入
create table BUMON_V(
BUMON_CD nchar(3)
, BUMON_NAME nvarchar(20)
, primary key CLUSTERED (BUMON_CD)
)
insert into [dbo].[BUMON_V] values('001','営業部')
```

©GRI Inc.

ETLツールの良い所、悪い所

- ETLツール全般の良い所
  - データフローが可視化され属人化を防ぎやすい
  - 現場に任せやすい（分析官に配りやすい金額）

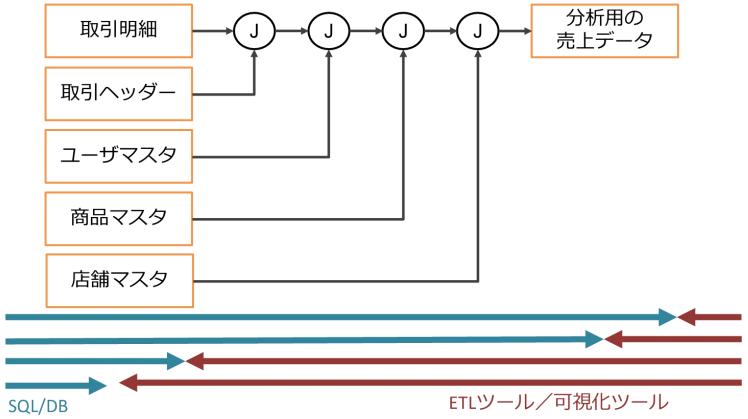
- デスクトップ版のETLツールの微妙な所
  - 実行系システムからデータを抜かなければいけない
  - 巨大データは扱えない

- サーバ版のETLツールの微妙な所
  - サーバの巨大メモリに乗らないと処理できない
  - コストがかかる（１エンジニアの給料くらいはかかる）

©GRI Inc.

ETLツールの良い所、悪い所

対応方法	費用面	分析のボトルネック	運用のボトルネック
①フルSQL＋直繋ぎ	多額のエンジニアリング投資	分析時のレスポンスが微妙	変更時にエンジニアに頼らなければならない
②フルSQL＋ファイル連携	そこそこのエンジニアリング費用	出力ファイルサイズがネックになる	SQLの管理が属人化する
③ETLツール＋直繋ぎ	ツール費用＆レプリカの準備	巨大データでの処理時間	同左
④ETLツール＋ファイル連携	ツール費用＆出力費用	ファイル連携の手間＆巨大データでの処理時間	同左



列指向DB（大規模データの分析に適している）

- 列指向DB：　列方向にデータを圧縮して格納
  - 分析する時は列方向に集計するので、列方向の処理にフォーカス
  - ID-POSデータ、ウェブアクセスログなどを扱った分析に向いている
  - BigQuery, Redshift, Snowflake, TreasureData, Vertica, (Tableau Hyper)
  - 高速に集計できる
  - 圧縮時に時間が必要。１件のデータ修正は遅い

行指向DB

ユーザID	性別	登録日	都道府県
10001	男性	2020/1/1	東京
10002	男性	2020/1/1	神奈川
10003	男性	2020/1/2	神奈川
10004	女性	2020/1/2	大阪
10005	女性	2020/1/2	大阪

→ 列圧縮

列指向DB

ユーザID	性別	登録日	都道府県
10001	男性x3	2020/1/1x2	東京
+1			神奈川x2
+1		2020/1/2x3	
+1	女性x2		大阪x2
+1			

©GRI Inc.

列指向DB（大規模データの分析に適している）

列指向DBは行指向DBと比較して10倍から10,000倍以上速い
→多くの行数のデータを分析できる

	DB名	SELECT		
		1万	100万	1億
行指向	MySQL	0.06s	1.5s	55.5m
行指向	PostgreSQL	0.06s	0.7s	1.2h
列指向	TREASURE DATA	2s	2s	8s
列指向	Redshift	0.2s	0.3s	0.4s

https://www.slideshare.net/optsendai/ss-81845870

©GRI Inc.

列指向DBの速度（肌感覚）

BigQuery
Snowflake
Redshift
Vertica

>>

10~100倍くらいの速度の差（個人の感想）

TreasureData
Athena

※Join, Unionなど、それぞれ一長一短あり、課金体系で使い方が異なる

※Tableau OnlineのHyperの速度なので、分析目的であれば選択肢から外れる
（それ以外の理由が必要）

©GRI Inc.

分析目的での列指向DBの選択

列指向DB	選択理由	注意が必要な所
BigQuery	GAを使うならば一択。クラウドから自由に選べるならばGCPでBigQueryを第１候補に考えている（コスパ）	• 使う側がクエリ課金のことを知っている必要がある。（安くなるテーブル構造） • 冪等性やシャーディングについて理解できている
Snowflake	AWSを使うなら第１候補。速度（費用）の選択肢が多く、逃げ道が多い。多くの分析官で使う場合、費用と速度を制御しやすい	• 意外と費用がかかることがある（Snowflakeへデータロードする際の費用に疑問点…）
Redshift	超大人数で使う場合、データサイズ課金が魅力	• 容量課金なので性能の逃げ道としての集約して別テーブルなどがコスト上限にかかる • 同接数が増えるとパフォーマンス低下 • 契約サイズを変える時の移行が大変
Vertica	オンプレならばVertica（高額だが）。	

- AzureのSynapseやOSSの列指向DB（2013年以降）の経験があまりなく、知見がおありでしたら、お教えください
- Hadoopはエンジニアやサーバの管理が大変（数百台以上のサーバを稼働させているならば、数日に１回はディスクエラーが起きる）

©GRI Inc.

列指向DBのビミョーな所

- データを収集して分析基盤の一か所に集めるのが大変

- 列指向DBにデータを投入した後、データ整形処理が必要な時、SQLやETLツールでの本質的な問題点を解決できない（列指向DB側でデータマートまで作っておくべき）
  - 例外：　AlteryxのIn-Databaseモード（ただしBigQueryを除く）

- 手元の管理ファイルや簡易集計データも、わざわざ列指向DBに入れるのが面倒

©GRI Inc.

おすすめのデータ分析基盤

©GRI Inc.

おすすめの分析基盤（少ないエンジニア数の会社）

- クラウド上の列指向DB（BigQuery, Redshift, Snowflake）
  - 大規模データの格納
  - 集計
- ELTツール（Matillion, Xplenty, Stitch）
  - 列指向DBへのデータ投入
  - データパイプライン
  - ジョブスケジューリング管理
- BIツール（Tableau, PowerBI, Qlik, Looker）
  - データ可視化＆共有
- 自動機械学習ツール（ForecastFlow, DataRobot, Prediction One）
  - 予測モデル構築（訓練）＆推論

©GRI Inc.

おすすめの分析基盤としてクラウド列指向DB＋ELTツール

- ELTツールを使う（Extract（抽出）、Load（ロード）、Transform（整形））
  - 例）Matillion, Xplenty, Stitch

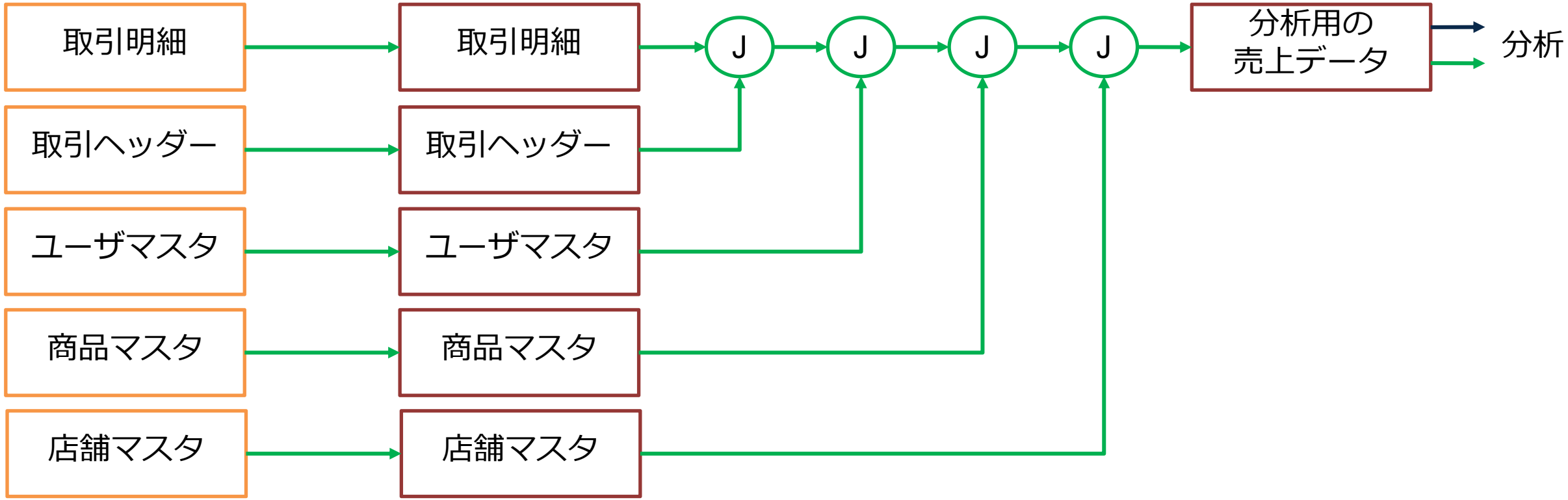
- 主なメリット
  - データを列指向DBに繋ぎこむのが簡単
  - データ整形処理を高速な列指向DBのパワーを使える
  - 複雑なパイプラインを可視化できる
  - 複雑なジョブ管理ができる（古くはJP1やTivoliの値段を考えると格安。Digdag+Embulkよりエンジニアの技術レベルが低くて使いやすい）

※なお、クラウドサービスで提供しているツールは、すごく高額で低速でコスパが悪かった

©GRI Inc.

現代的な分析基盤の例（Matillionを中心とした場合の流れ）

おすすめの分析基盤としてクラウド列指向DB＋ELTツール



実行系システムのレプリカDB
ダンプ済みファイル
分析基盤のデータレイク

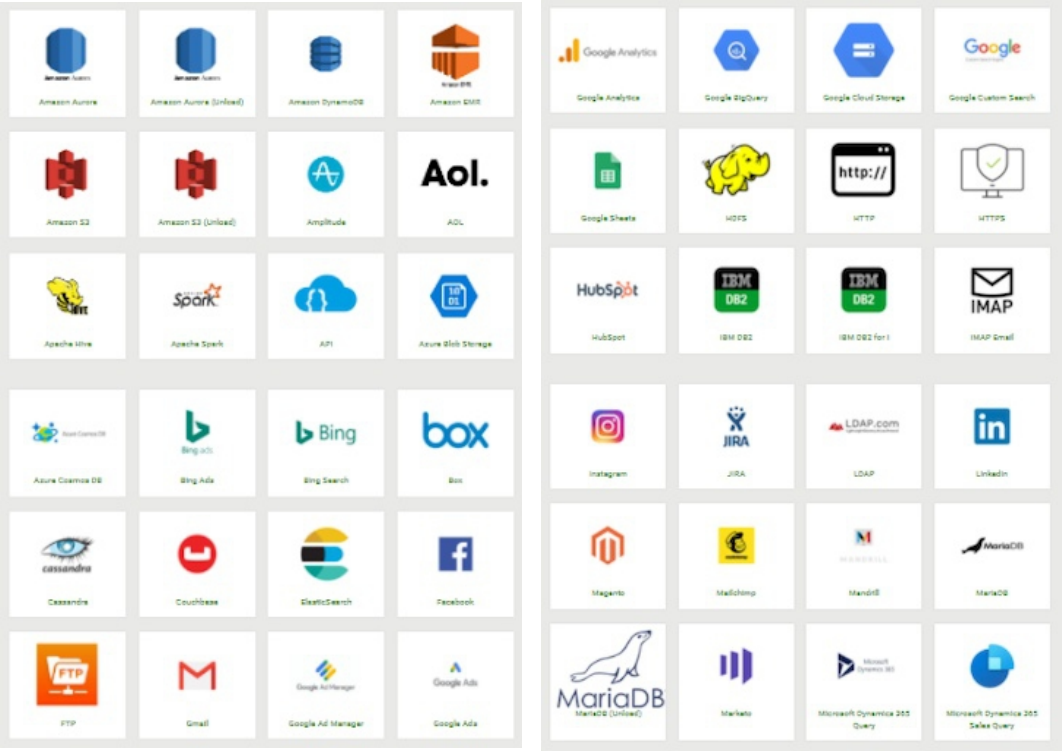
分析基盤の列指向DB

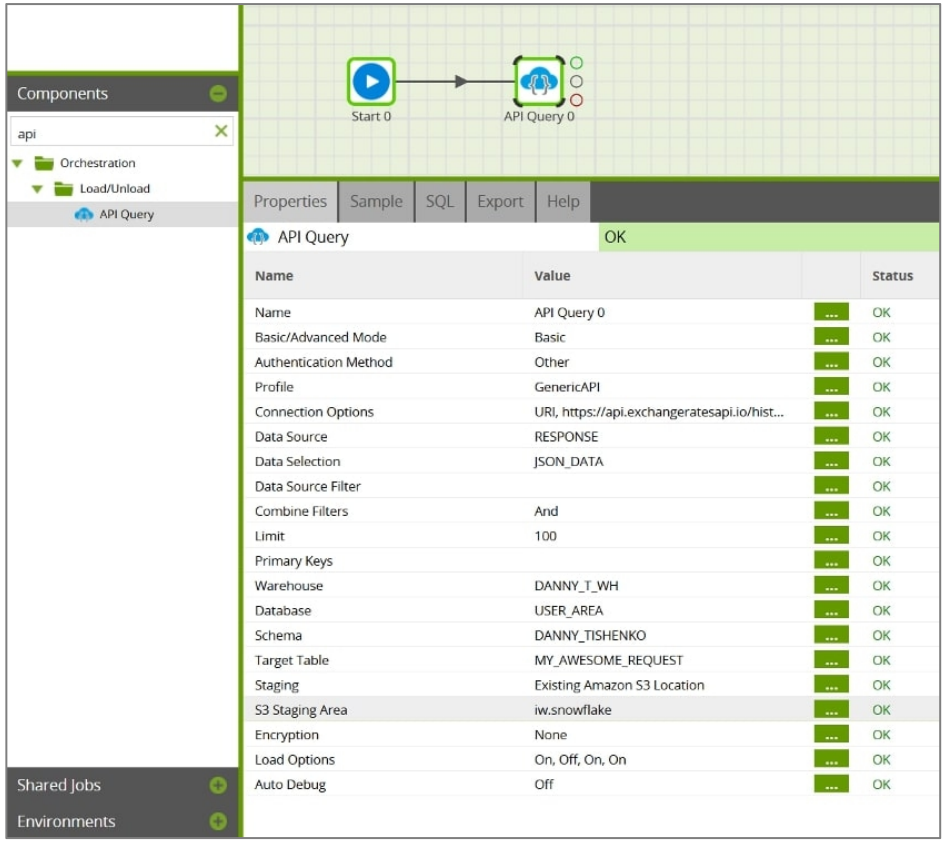
ELTツール

分析ツール

ELTツールを使うと分析基盤への繋ぎ込み工数が激減

- 【Orchestration】理由は、コンポーネントをポチポチ設定するだけで連携できるので、I/Fの要件定義書・概要設計書・詳細設計書・テストを簡略化できる
- システム部門のキーパーソンに頼まなくてOK



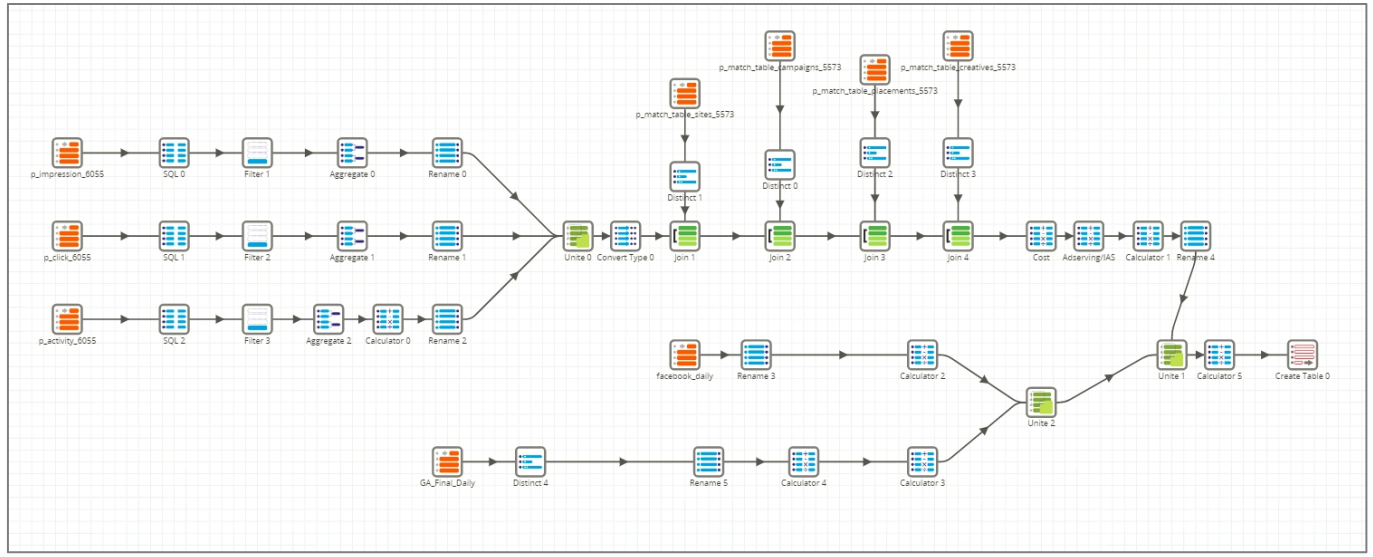


ELTツールは列指向DB側のパワーをそのまま使える

- ビッグデータをクラウド列指向データベース（Redshift/BigQuery/Snowflake）のパワーをフル活用できる
  - 分析上のデータ活用を安価で高速処理を実現
  - 遅くて古い種類のデータベース処理に戻れない

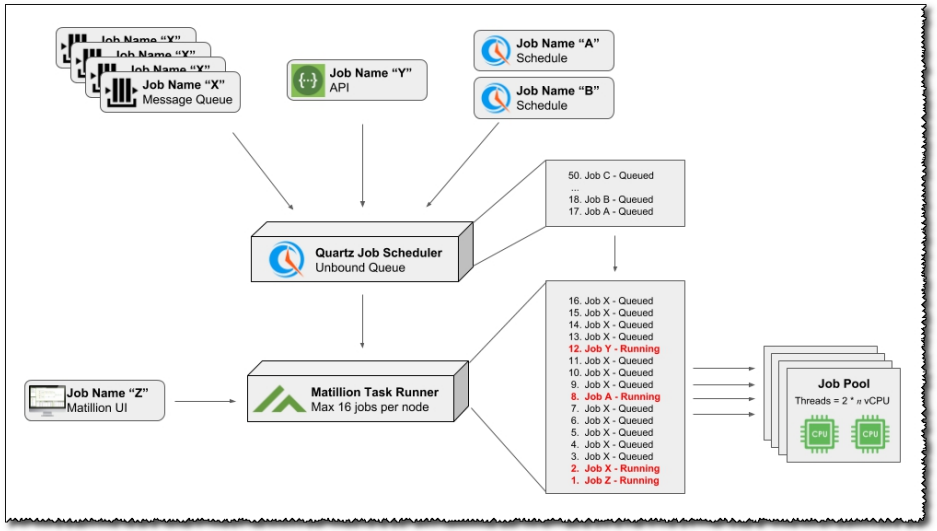
ELTツールはクラウド上でデータパイプラインを共有しやすい

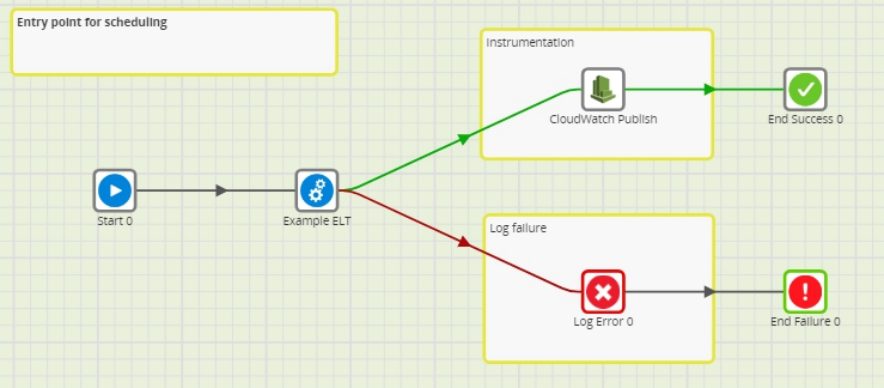
- 【Transformation】可視化されているので、担当者間でパイプラインの中味を理解しやすい
- データが途中途中で、どのように変化していくかトラッキングできるので、頭で考えながらパイプラインを設計＆構築しやすい

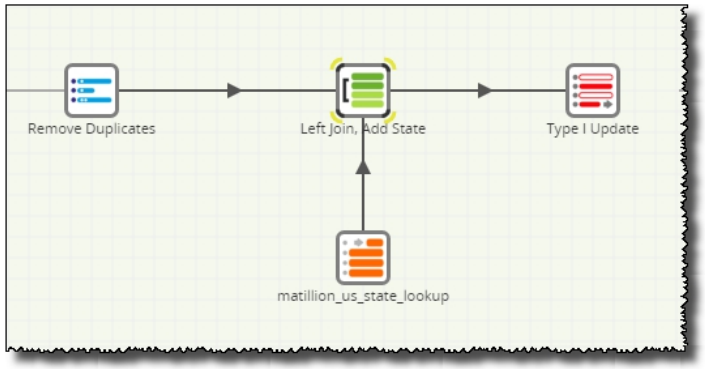


ELTツールはジョブ管理やスケジューリング機能が充実

- ジョブの依存関係、複雑な分岐、エラー処理
- ジョブの定義書の自動生成







定義書の自動生成

Parameter	Value
Main Table	Remove Duplicates
Main Table Alias	airports
Joins	matillion_us_state_lookup, states, Left
Join Expressions	"airports"."state"="states"."code", airports_Left_states
Output Columns	airports.iata, iata, airports.airport, airport, airports.city, city, airports.state, state, airports.country, country, airports.lat, lat, airports.long, long, airports.last_updated, last_updated, states.name, name

自動機械学習ツールの良い所、悩み所

- 良い所
  - 顧客ID単位での予測スコアや予測の理由が分かる
  - ノーコードで実行できる

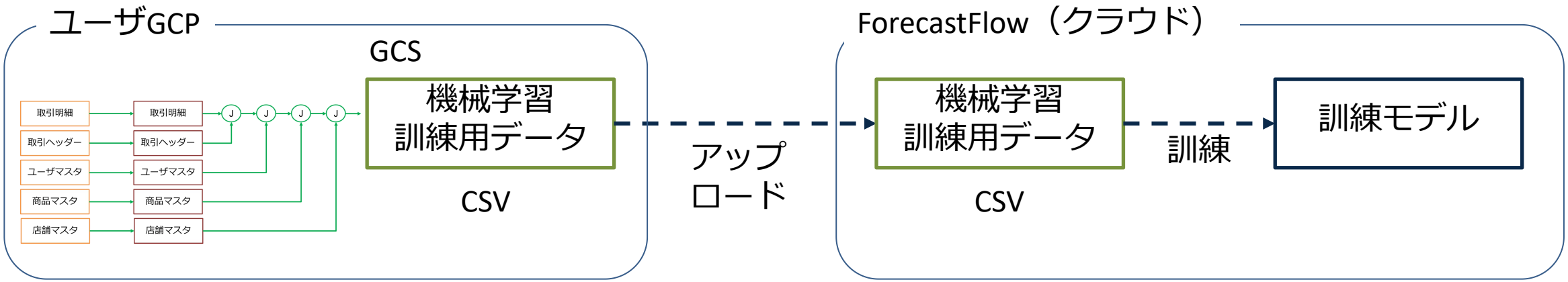
- 悩み所
  - 良いモデル（特徴量）でないと、精度も示唆もない
  - 自動機械学習ツールに備わっている前処理機能より、前処理専門ツールの方が使い勝手が良い
  - 予測スコアは単体では面白くなく、実行系システムのデータと連携させたり、施策実行結果と結合させることで意味が出てくる
  - 超大規模データで運用が回るか？
    - 例）数千万IDに対して数種類の予測結果を付与
  - 高額費用のツールの場合、２年目には魔法が効かない
    - 初年度驚いた結果は、２年目には既知情報である

©GRI Inc.

自動機械学習＋クラウド列指向DB＋ELTツール

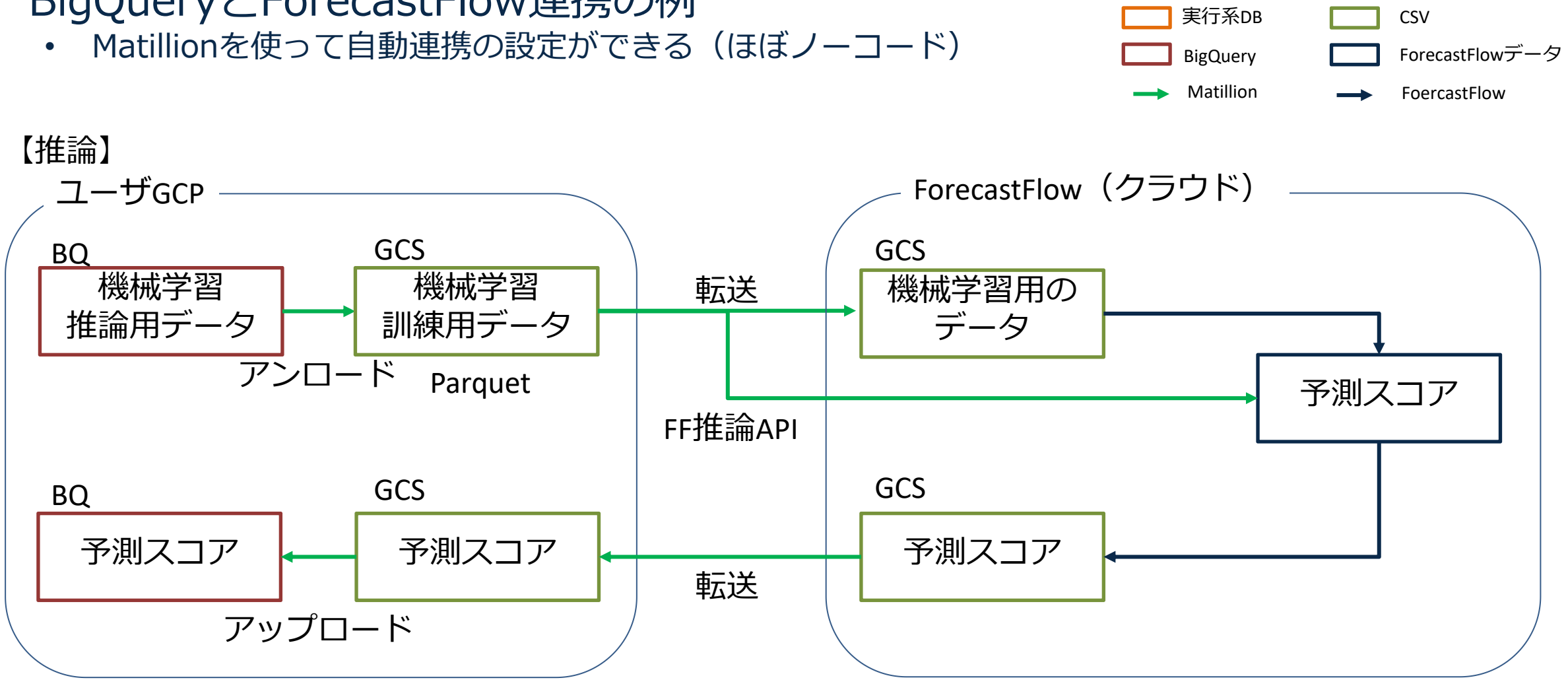
- BigQueryとForecastFlow連携の例

【訓練】



自動機械学習＋クラウド列指向DB＋ELTツール

- BigQueryとForecastFlow連携の例
  - Matillionを使って自動連携の設定ができる（ほぼノーコード）



社内実験ではBigQuery - AutoML Tables連携よりForecastFlow連携の方が高速だった（2,000万IDの推論で15分と12分）

分析基盤と連携することで自動機械学習ツールの悩み所を解消

- 悩み所
  - 良いモデル（特徴量）でないと、精度も示唆もない
    - 特徴量を作りやすい
  - 自動機械学習ツールに備わっている前処理機能より、前処理専門ツールの方が使い勝手が良い
  - 予測スコアは単体では面白くなく、実行系システムのデータと連携させたり、施策実行結果と結合させることで意味が出てくる
  - 超大規模データで運用が回るか？→回る
    - 例）数千万IDに対して数種類の予測結果を付与
  - 高額費用のツールの場合、２年目には魔法が効かない
    - 初年度驚いた結果は、２年目には既知情報である
    - ツールによる

©GRI Inc.

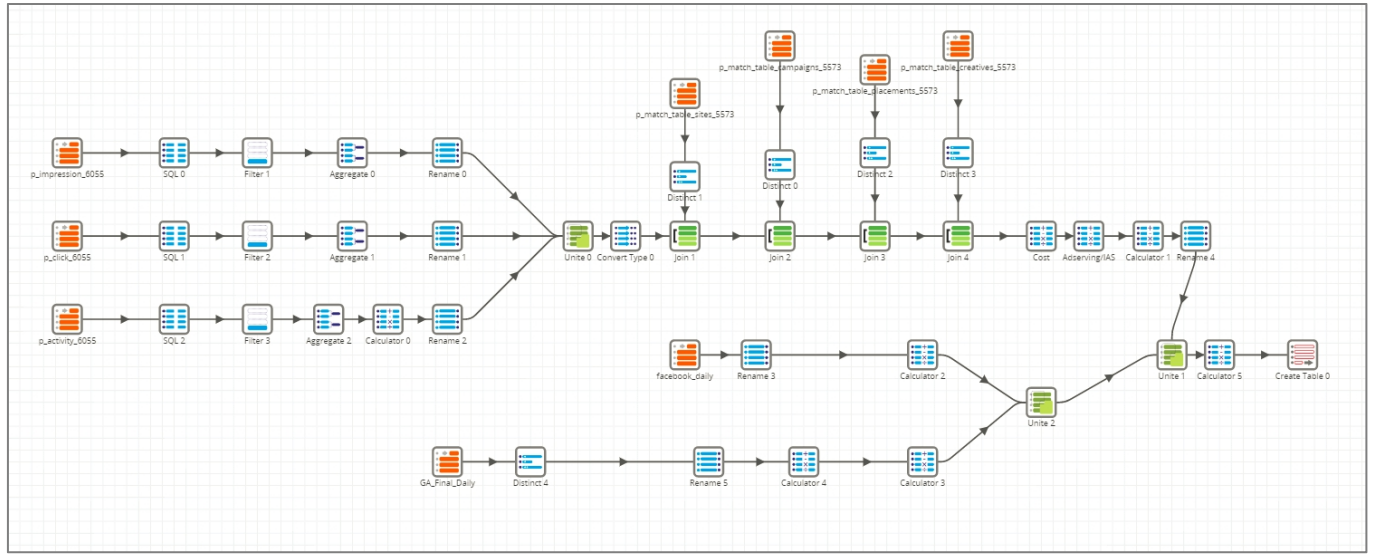
Matillionデモ

©GRI Inc.

ELTツールはクラウド上でデータパイプラインを共有しやすい

ここに絞ってデモ

- 【Transformation】可視化されているので、担当者間でパイプラインの中味を理解しやすい
- データが途中途中で、どのように変化していくかトラッキングできるので、頭で考えながらパイプラインを設計＆構築しやすい



デモの想定 ～月額制有料Webメディア運営会社～

■目的

–デモグラ情報やWebアクセスログから特徴を抽出し分析、解約しそうな顧客に対しては事前にアプローチしたい

■現状

–サイトにGoogleAnalyticsを仕込んでアクセスログを取得、データをBigQueryに連携するところまでは着手済み

■課題

–アクセスログを各顧客ごとに集計して行動を表す特徴量を作成したいが、SQLは書いたことがない、かつ周りにも詳しい人はいない、、、

▶ Google Analytics ▶ ▶ ・・・

デモ

id	age	gender	pref
1	45	女性	岩手県
2	33	男性	東京都
3	52	女性	山口県
4	58	男性	青森県
5	27	男性	沖縄県
6	48	男性	新潟県

＋

GAのアクセスログから各顧客ごとにWeb行動を集約・追加

▼例えば

・PV
・来訪頻度
・直近の来訪からの経過日数

©GRI Inc.

デモのまとめ：Matillion導入のメリット

- データ整形のフローがグラフィカルでわかりやすく、非エンジニアの人でも保守がしやすい
- 処理の途中でデータが望む形になっているか、都度確認しながら整形フローを作れるため、間違いに早く気づける

©GRI Inc.

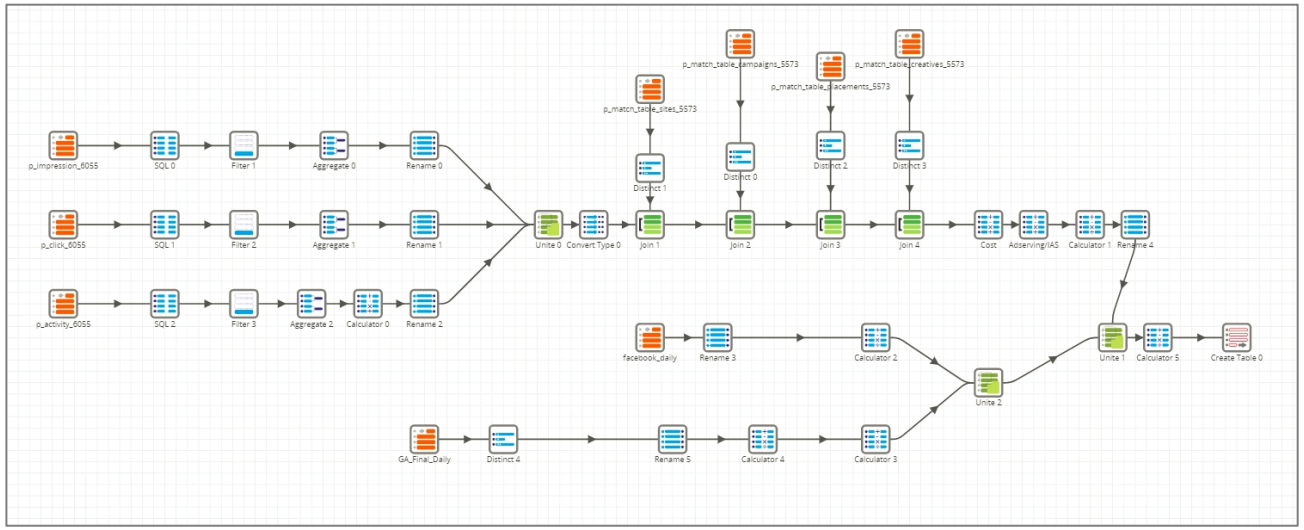
デモのまとめ：Matillion導入のメリット

■データ整形のフローがグラフィカルでわかりやすく、非エンジニアの人でも保守がしやすい

【個人的にSQLで一番辛いところ】
直列な記述なので、中間テーブルの依存関係がわかりにくい、、、

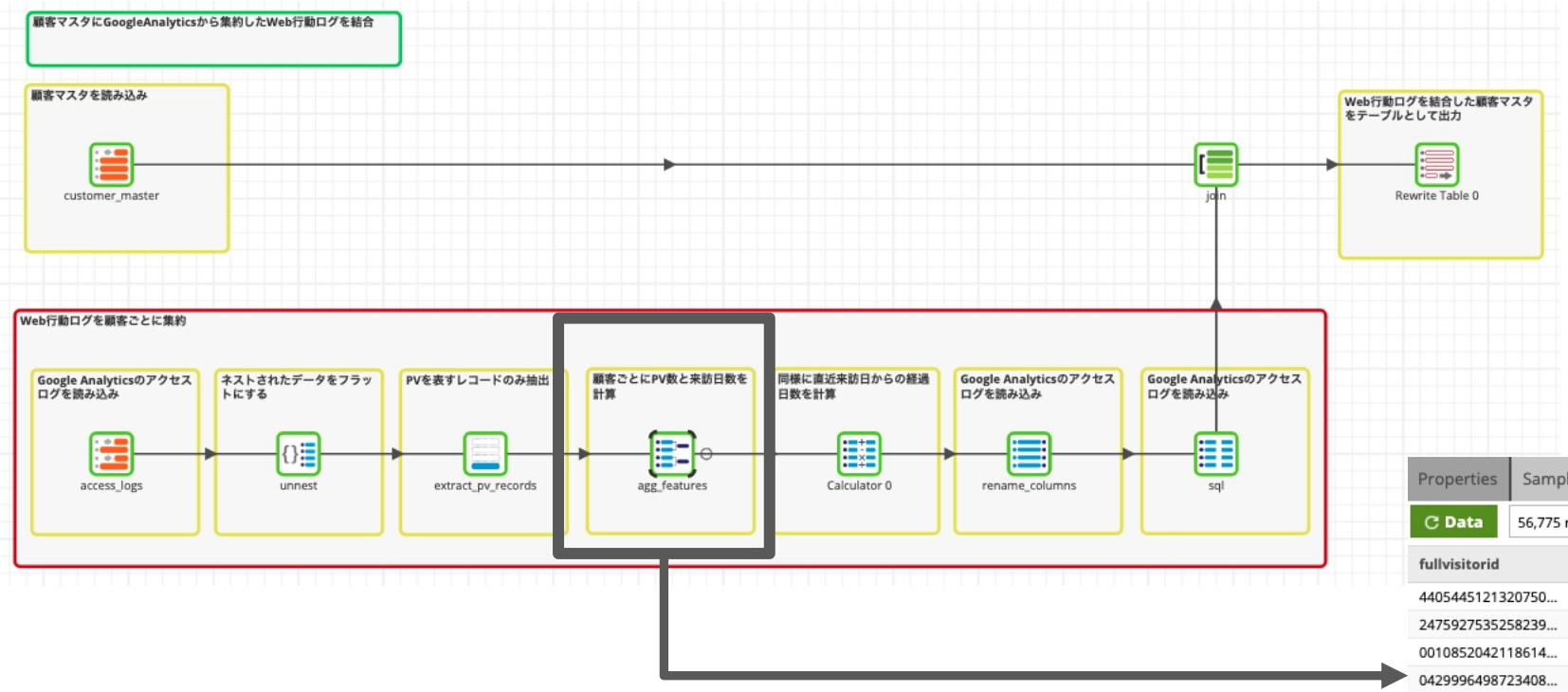
```
SET NUMERIC_ROUNDABORT OFF;
SET ANSI_PADDING, ANSI_WARNINGS, CONCAT_NULL_YIELDS_NULL, ARITHA
   QUOTED_IDENTIFIER, ANSI_NULLS ON;

--従業員テーブルの作成とデータ挿入
create table EMPLOYEE_V (
EMPLOYEE_ID nchar(5)
, NAME nvarchar(20) NOT NULL
, KANANAME nvarchar(20)
, BATHDAY DATE
, SEX nchar(1)
, JOIN_DATE DATE
, POST_CODE nchar(7)
, ADDRESS nvarchar(100)
, TEL nvarchar(11)
)
insert into [dbo].[EMPLOYEE_V] values('T0001','太郎','タロウ','1
insert into [dbo].[EMPLOYEE_V] values('T0002','花子','ハナコ','1
insert into [dbo].[EMPLOYEE_V] values('T0003','次郎','ジロウ','1
insert into [dbo].[EMPLOYEE_V] values('T0004','三郎','サブロウ',


--部門テーブルの作成とデータ挿入
create table BUMON_V(
BUMON_CD nchar(3)
, BUMON_NAME nvarchar(20)
, primary key CLUSTERED (BUMON_CD)
)
insert into [dbo].[BUMON_V] values('001','営業部')
```

Matillionだと平面的な表現になり、各入力テーブルの後続処理とそれらの結合処理が明らか



デモのまとめ：Matillion導入のメリット

■処理の途中でデータが望む形になっているか、都度確認しながら整形フローを作れるため、間違いに早く気づける



Properties | Sample | Metadata | SQL | Help

Data | 56,775 rows | Filter Not Set

fullvisitorid	count_date	count_distinct_date	max_date
4405445121320750...	5	3	20170712
2475927535258239...	1	1	20170703
0010852042118614...	1	1	20170703
0429996498723408...	1	1	20170703
0555928795747435...	1	1	20170703
7975449158982553...	1	1	20170703
9223970203704409...	1	1	20170703
8781616657504676...	1	1	20170703
0180751994544293...	1	1	20170703
3877631794017768...	1	1	20170703
1397685846819614...	1	1	20170703

©GRI Inc.

参考

■ForecastFlowとの連携

- 【ブログ記事】http://forecastflow.jp/datascicene/report3/
- 【動画（過去のすいすい会）】
  https://www.youtube.com/watch?v=TanBxco-kcA

■ブログ（https://gri-blog.hatenablog.com/）

- Matillionは日本語ドキュメントが手薄なところがネックであるため、弊社ブログで情報発信中

©GRI Inc.

まとめの話と
本日扱わなかった
データ分析基盤の話

©GRI Inc.

本日のまとめ

- 分析側からは、分析を熱いうちに行いたい（分析をしたいが、データがないため、エンジニアに依頼して、数か月後にデータが出来上がる環境を打破したい）
- 列指向DB + ELTツール + BIツール + AIツールの組み合わせで、解決できることも多い
- ツールごとのコストパフォーマンスは、価格体系に直結しており、データの種類、分析方法、分析で利用する人数によって変わる（なかなか一概には言えない…。案件ご相談いただけましたら幸いです）
- 分析基盤の運用コストを回収できる知見や成果を、データ利活用からどのように出せるか？（これに尽きるが、データ分析基盤がないとスタート地点に立てない）

©GRI Inc.

ETLTという考え方

- ELTを中心にしたデータ基盤では、基盤に入れる前に、匿名化処理をしてから入れることが多い
- ETLTは、データ基盤側で匿名化処理をする考え方

https://dzone.com/articles/what-is-etlt-merging-the-best-of-etl-and-elt-into-1

©GRI Inc.

それ以外のおすすめの分析基盤

- 超大規模データの分析では、Pythonで並列処理を書ける人＋データパイプラインの管理
  - 例）スマホゲームのアプリログ、超大量のIoTデータ

python™

Spark

databricks

Python人材を集められたが、基盤側の人材が少ない場合、選択肢に

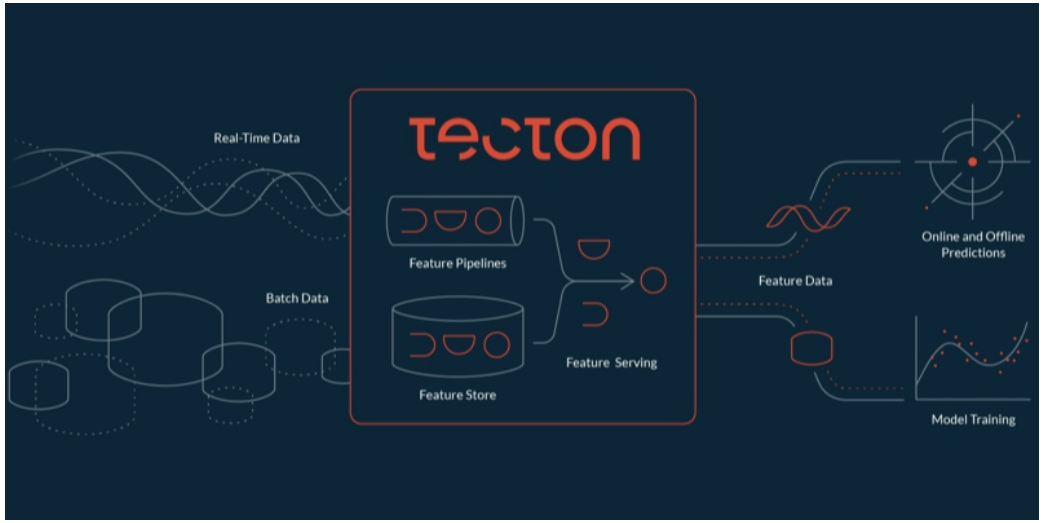
Apache Airflow

PREFECT

それ以外のおすすめの分析基盤: Feature Store

- 大規模データをオンラインとオフラインで機械学習する必要がある場合、Python＋機械学習の深い知識＋基盤知識

アーキテクチャ名	種別	概要
Michelangelo	内製	Uberで運用されている、Feature Storeに限らないML基盤
Zipline	内製	Airbnbで運用されているFeature Store
Tecton	マネージド	大規模なMLシステム向けの商用Feature Store
Amazon SageMaker Feature Store	マネージド	re:Invent 2020で発表されたAWS謹製のFeature Store
Hopsworks	OSS	KubeflowのようなML基盤で、その中の機能の一つとしてFeature Storeがある
Feast	OSS	GO-JEKとGoogleCloudが共同で開発を進めているFeature Store



https://aws.amazon.com/blogs/startups/tecton-feature-store-brings-devops-to-ml-data/

https://f6wbl6.hatenablog.com/entry/feature-store-overview

次回のすいすい会

- 2021年9月29日（水）18:30-19:30

- （仮）ECデータを自動機械学習ツールで優良顧客の定着を予測する

©GRI Inc.

残っている質問（今後、説明していきたい内容）

- ビジネスの実践例を知りたい
- どのくらい正解データの数があるべきか
- 未来の施策と機械学習の実行タイミングの関係性
- ダウンサンプリングとProbability Calibration
- 様々な自動機械学習ツールの比較
- Feature Store、分析基盤、CDPの説明

©GRI Inc.

ForecastFlowの
リリース案内

©GRI Inc.

ForecastFlowに関する共有事項

- 自動機械学習ツールForecastFlowの情報源
  - http://forecastflow.jp/
  - 90日間の無料トライアルあり

©GRI Inc.

ForecastFlowのロードマップ

- 2021年9月リリース予定
  - チーム機能（チーム内でデータとモデルを共有）
  - 多言語化（日本語を含む）
  - 予測モデルの解釈画面のダッシュボード化