Engineer

スタースキーマ

★ 一つのファクトテーブルの周囲にディメンションテーブルを置く構成

どのようなものか

スタースキーマは、分析のためのテーブル配置です。中央の一つのテーブルに、売上、出荷、支払いなど、業務上の出来事を数値として記録します。その周囲には、出来事を説明するテーブルを置きます。どの商品、どの店舗、どの日、どの顧客なのかを表すものです。中央のテーブルから周囲の各テーブルへ線を引いて図にすると、星のような形になります。

中央にあるのがファクトテーブルで、周囲にあるのがディメンションテーブルです。Kimball Groupは、「スタースキーマは、関連するディメンションテーブルにつながるファクトテーブルで構成されることが特徴」と簡潔に説明しています。この接続に使うのは、普通のキーです。各ディメンションテーブルは一つの主キーを持ち、各ファクト行は、出来事を説明するディメンションごとに一つずつ、そのキーを外部キーとして保持します。

この分け方は、Kimball Groupが示す測定値と文脈の区別に基づきます。ディメンショナルモデルの中心にあるのは、業務プロセスにおける測定の対象となる出来事です。レジで商品を読み取る、注文を出荷する、保険金を支払う、といった出来事です。数量や金額など、その出来事が生む数値がファクトです。誰が、何を、どこで、いつ、なぜ、どのように、を説明するものはすべて文脈であり、ディメンションに入ります。

二種類のテーブルは、対照的な形をしています。最も詳細なレベルでは、ファクトテーブルは出来事ごとに一行を持ち、数値の測定値、各ディメンションへの外部キー、ときには請求書番号やレシート番号などの参照情報を保持します。文書の日付、顧客、店舗をディメンションへ移すと、その文書から残るのは、その番号だけです。そのため、対応するディメンションテーブルを作らず、ファクト行に番号を置きます。Kimball Groupは、これを縮退ディメンションと呼びます。一方、ディメンションテーブルは幅が広く、フラットで、意図的に非正規化さ

れ、説明用のテキスト列を多数持ちます。商品ディメンションでは、ブランド、カテゴリー、部門を別々のテーブルに分けず、各商品の行に繰り返し記録します。

スタースキーマは、ディメンショナルモデルをリレーショナルな形で表したもので、通常のSQLデータベースの通常のテーブルで構築します。Kimball Groupが説明するもう一つの形が、同じ属性とファクトを保持する多次元データベース、OLAPキューブです。MDXなど、SQLより分析能力の高い言語で問い合わせ、アクセス用のプロトコルはXMLAが提供します。キューブは、利用者へデータを公開する直前の最終工程や、集約値の層として、スタースキーマから作ることもよくあります。どちらにも同じモデリングのルールが当てはまります。Kimball Groupが、それらをアーキテクチャーに依存しないとするのは、Kimball方式のデータウェアハウスにも、Bill Inmonのハブアンドスポーク型Corporate Information Factoryにも適用できるためです。

概念を判断に生かす

「スタースキーマとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 営業、運用、経理の数値をまとめ、同じ会議に部門ごとに異なる答えを持ち寄る状況を解消します。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 地理情報パイプラインを50倍に高速化 — 50x 地理データパイプラインの高速化

地理情報パイプラインを50倍に高速化 お問い合わせ →

なぜ重要なのか

この形にすると、分析の質問が同じような構造になります。合計するファクトを選び、それを説明するディメンションへファクトテーブルを結合し、一部のディメンション列で絞り込み、別の列でグループ化します。Kimball Groupは、計算や動的な集約の主な対象がファクトテーブルであり、問い合わせの制約条件、グループ分け、レポートのラベルの主な取得元がディメンション属性だと説明しています。一度このパターンを学んだ分析者は、助けを借りずに新しいファクトテーブルへの問い合わせを書けます。どのスタースキーマも同じ配置なので、レポート作成ツールも、フィールドの一覧から問い合わせを組み立てられます。

二つ目の理由は、読みやすさです。業務用のデータベースは、各事実を一度だけ保存し、一貫性を保つよう設計します。そのため、商品のカテゴリーは通常、別の参照テーブルに置き、idで結合します。注文を受けるシステムには適切ですが、カテゴリー別の売上を知りたい人は、まず、どのテーブルをどの順で結合するかを見極めなければなりません。その用途には不向きです。フラットなディメ

ンションなら、説明用の全列が、言葉で書かれた形で一か所に集まります。Kimball Groupは、その逆、つまりディメンションの階層をさらに別のテーブルへ正規化する方法を避けるよう注意しています。業務部門の利用者には、結果を理解したり、たどったりしにくくなるためです。

三つ目の理由は、適切に設計したスタースキーマなら、設計時には想定しなかった質問にも答えられることです。ファクトテーブルの粒度、つまり一行が正確に何を意味するかは、ディメンションやファクトを決める前に宣言します。Kimball Groupは、業務プロセスが記録する最も細かなレベル、アトミックな粒度から始めるよう推奨しています。レシートの明細一行ごとに一行を持つテーブルなら、日、店舗、ブランド、あるいはその組み合わせで集計できます。店舗ごとの日次合計しか保持しないテーブルは、質問の仕方をどう変えても、商品別に分解できません。Kimball Groupの言い方では、アトミックなデータは予測不能な問い合わせに耐えられますが、集計済みデータは、後から問われる質問を前提にしています。

その代償は、意図的な重複です。部門名は各商品の行に繰り返され、スタースキーマを構築するロード処理は、それらの重複の整合性を保ち、説明が変わった際の扱いも決めなければなりません。スタースキーマは業務モデルを置き換えるものでもありません。ロード処理によって業務モデルから作るため、最新性は最後にロードした時点までに限られます。

実際の活用例

小売業者が、商品カテゴリー別、月別の売上を求めています。業務プロセスはレジのレシートで、粒度は、レシートの明細一行につき一行と宣言します。商品ディメンションは一つのフラットなテーブルです。ファクトテーブルは、各ディメンションのキー、縮退ディメンションであるレシート番号、二つの測定値を持ちます。

```
CREATE TABLE dim_product (
    product_key  bigint PRIMARY KEY,   -- surrogate key, owned by the warehouse
    sku          text NOT NULL,        -- the code in the source system
    product_name text NOT NULL,
    brand        text NOT NULL,
    category     text NOT NULL,
    department   text NOT NULL
);

CREATE TABLE fact_sales (
    date_key       integer NOT NULL REFERENCES dim_date,
    store_key      integer NOT NULL REFERENCES dim_store,
```

```
    product_key    bigint  NOT NULL REFERENCES dim_product,
    receipt_number text    NOT NULL,   -- degenerate dimension
    quantity       integer NOT NULL,
    sales_amount   numeric(12, 2) NOT NULL
);

SELECT d.year_month, p.category, sum(f.sales_amount) AS sales
FROM fact_sales AS f
JOIN dim_date AS d USING (date_key)
JOIN dim_product AS p USING (product_key)
WHERE p.department = 'Grocery'
GROUP BY d.year_month, p.category
ORDER BY d.year_month, p.category;
```

問い合わせは、どのスター型の問い合わせにも共通する形をしています。ファクトテーブル、ディメンションごとの一つの結合、ディメンションのテキストに対する絞り込みとグループ化、ファクトに対する合計です。レポートに店舗の地域を追加するなら、結合を一つと、`GROUP BY` に列を一つ追加するだけで、ファクトテーブルは変わりません。レシート番号はファクト行に残るため、一つの買い物の明細をまとめて数えることもできます。それ以外に説明すべきことがないため、番号用のディメンションテーブルは作りません。

スタースキーマは、データベース設計・モデリングサービスが約束する、業務の実態に合い、後で必要になる質問にも答えられる構造を実現する一つの方法です。その元になる業務モデルが、この仕事のもう半分です。NextMarinerでの現在の業務では、創業者が、専門職、企業、船舶、求人、レビューを含む海事分野を、34個のPostgreSQLスキーマにまたがる348個の正規化されたテーブルとしてモデル化しました。そこでは、状態、種類、カテゴリーなどの小さく安定した一覧をSMALLINTの参照テーブルにし、行を小さく、結合を低コストに保っています。実際の外部キーと制約によって、整合性はデータベースの責務になります。これは、一行ずつ書き込み、正しさを保つ必要のあるシステムに合った形です。その上にレポート用のスタースキーマを構築する場合は、逆を行います。読むのはアプリではなく人なので、フラットなディメンション内で、コードを言葉へ戻します。

問題が起きるところ

ディメンションを正規化し、スタースキーマがスノーフレークになる。 商品ディメンションからブランド、カテゴリー、部門を別々のテーブルへ分けると、業務用の設計に慣れた人には整然と見えます。Kimball Groupは、この結果をスノーフレークと呼び、避けるよう助言しています。業務部門の利用者には理解してたどることが難しくなり、結合が増えると問い合わせが遅くなる場合もあります

が、フラットなディメンションでも、まったく同じ情報を保持できるためです。各ディメンションは一つの幅広いテーブルに保ち、ロード処理で各行に階層を繰り返し書き込みます。

最初のレポートに合わせて粒度を決める。 店舗ごとの日次売上を持つファクトテーブルは、想定したダッシュボードには答えられますが、それより細かな問いには答えられません。単一の商品や買い物かごについて最初に質問された時点で、ソースからの再ロードが必要になります。日次合計と個別明細を混ぜたテーブルは、さらに悪く、合計を取るたびに同じ売上を二重に数えます。まず粒度を宣言し、そのプロセスが記録するアトミックなレベルから始め、粒度ごとに専用のファクトテーブルを設けます。速度が必要なら、その上に別のテーブルとして集計テーブルを追加できます。

ディメンション属性がソースシステムのコードになっている。 業務システムは短いコードを保存します。それをディメンションへそのままコピーすると、行のラベルを理解できるのがソースシステムだけ、というレポートになります。Kimball Groupは、属性としてコードを持つことを認めていますが、最も役に立つ属性は詳しい説明だと述べています。レポートのラベルになるのは、それらの説明だからです。ロード時にすべてのコードを言葉へ展開し、行をソースまでさかのぼって追跡する必要のある人のため、説明の隣の列にコードも残します。

ファクトテーブルがレポートの形をしている。 業務で既に使っている表計算シートから設計を始め、月ごとに列を設けたり、経営レポートの一行ごとに行を作ったりしたくなります。Kimball Groupは、ファクトテーブルの設計は、後で作るレポートではなく、測定される実際の活動に基づくと明言しています。レポートのレイアウトは変わっても、レジのレシートは変わりません。宣言した粒度で発生一回につき一行として出来事をモデル化し、レイアウトは問い合わせに生成させます。

参考資料

- Star Schemas and OLAP Cubes — 定義、測定値と文脈としてのファクトとディメンション、キューブとスタースキーマの関係を説明しています。
- Grain — 一つのファクト行が意味するものの宣言と、アトミックな粒度を先に決める理由を説明しています。
- Fact Table Structure — レポートではなく活動に基づいて設計し、出来事ごとの行に数値の測定値と外部キーを置く構造を説明しています。
- Dimension Table Structure — 絞り込み、グループ化、ラベルに使う説明属性を持つ、幅広く、フラットで、非正規化されたテーブルを説明しています。

- Snowflaked Dimensions — ディメンションの階層を正規化すると、使いにくくなる理由を説明しています。
- Microsoft: Multidimensional model data access — 問い合わせと計算の言語としてのMDXと、クライアントがMDX文を送るためのインターフェースであるXMLAを説明しています。

用語集

実際の制約に合わせて選ぶ

代表的なデータセット、そこから答えを得たい問い、現在のボトルネックをお知らせください。品質、処理量、復旧の測り方を合意できます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「スタースキーマとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/star-schema/