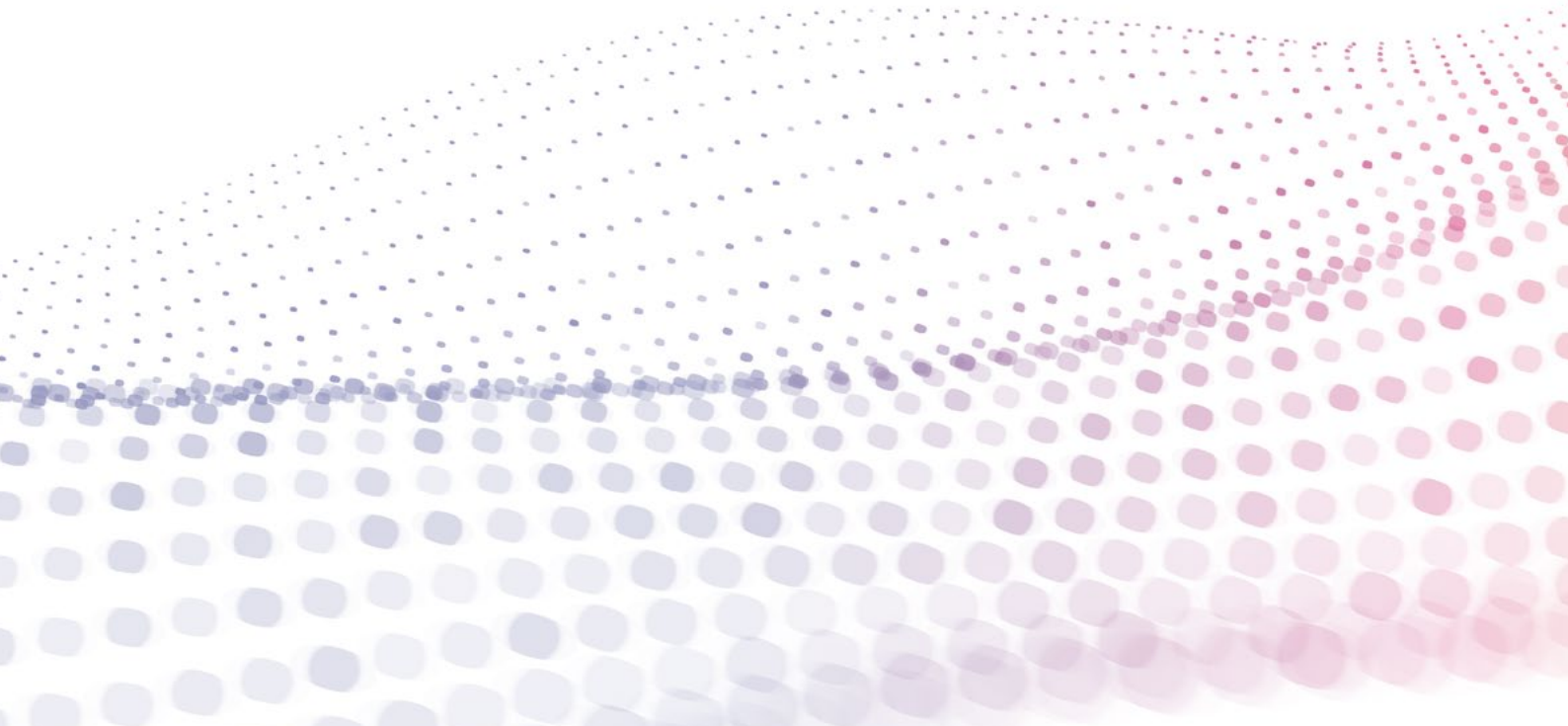
MarkLogic

スマートなメタデータ管理

MARKLOGICホワイトペーパー ・ 2018年7月

データには大きな価値がありますが、その価値は、データを真に理解して初めて引き出すことができます。データを理解するにはメタデータを利用する必要がありますが、メタデータは通常、サイロに閉じ込められ、その真の価値は認識されていません。このホワイトペーパーでは、組織がメタデータの価値を引き出す方法について説明し、MarkLogic® データベースによるスマートなメタデータ管理を実現する優れたデータモデリング / アーキテクチャ手法を紹介します。



目次

> MarkLogicデータベースを使用すれば、データとメタデータをまとめて扱うスマートなメタデータ管理を実現できます」

はじめに

データ：データは、ビジネスに必要であり、業務の基盤であり、またビジネスの中核となるものです。しかし残念ながら、部分的なデータにばかり注意を向け、全体としてのデータを理解できないことが多いのが現実です。この場合、データはさまざまなグループに分断され、隔離されています。仮にこれらのグループ間の連携が許可されたとしても、通常は夜間の ETL ジョブによってこれを行うため、時間がかかり、間違いが起きやすくなります。

データサイロは、DBA にとって悩みの種となるだけでなく、データを理解できない原因にもなります。ガバナンスとデータの品質が犠牲になり、監査は恐れる対象となり、組織の発展が阻害されます。

メタデータの導入：メタデータを適切に管理すれば、データの統合やガバナンスを実現し、さまざまな困難なビジネスニーズを満たすことができます。問題は、メタデータとそれに付随するあらゆる価値がサイロに分断されており、メタデータを全社的に管理する方法や、その価値を最大限に引き出す方法がないことです。

幸いなことに、MarkLogic データベースを使用すれば、データとメタデータをまとめて扱うスマートなメタデータ管理を実現できます。このホワイトペーパーでは、メタデータとは何か、またなぜメタデータが重要なのかを紹介していきます。その後、MarkLogic によるスマートなメタデータ管理について説明し、MarkLogic のマルチモデルアプローチ、検索とインデックス付け、オペレーショナルデータハブパターンについて詳しく解説します。オペレーショナルデータハブパターンにより、データとメタデータを素早く簡単に統合、管理でき、そこからかつてないほど迅速に価値が得られます。

メタデータとは

メタデータは、広い意味では、データに関するデータです。この定義は最初にメタデータを説明する際には便利ですが、極めて表層的です。

メタデータの定義が困難な理由の 1 つは、人にとって、メタデータの意味するところがばらばらだということです。MarkLogic ユーザーに対する調査では、メタデータの意味は、各組織におけるデータの作成、管理、使用方法によって違っていることが判明しました。

エンドユーザー（データ利用者）は、メタデータとは本質的にオントロジーまたはタクソノミー的なもの、あるいはデータの関係性に関わるものと考えるでしょう。一方、データアーキテクトやデータベースプログラマーは、メタデータとは、データモデル、データアーティファクトの履歴および維持、または組織内のワークフローに関するものと考える傾向があります。

また、メタデータの定義は業界やユースケースによって異なります。例えば、金融サービスの IoT には、ATM、支払処理端末、支払い用のスマートフォンやスマートウォッチ、セキュリティセンサーに関するメタデータがあります。これらのコンポーネントのほとんどには、IT アセット管理用のメタデータがあります。このメタデータのうち、ごく一部はデバイス自体から提供されています（ファームウェアバージョン、物理的な場所、モデル、エラーログ、シリアル番号、所有者、交換部品の供給会社、操作マニュアルの URL など）。また、これらのデバイスに間接的に関係する業務用メタデータがあります（セキュリティ証明書、スキーマ、スケジュールなど）。

このように、メタデータにはさまざまなものがあります。メタデータとデータの境界は流動的です。見方によって、ある組織でデータと呼ばれるものが他の組織ではメタデータと呼ばれる可能性もあります。どちらにしても、何を把握できる / すべきか、ならびにその管理方法の特定が重要です。

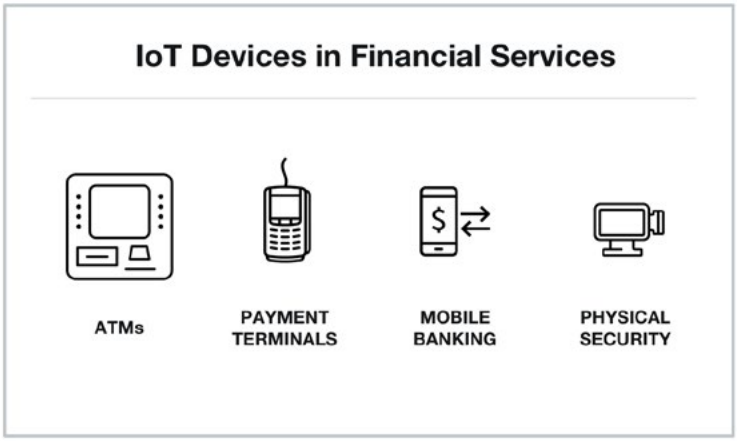


図1：金融サービスのIoTデバイスは、活用できる多様なメタデータの好例です。

メタデータが重要な理由

検索と分析

データは多くの組織にとって最も貴重なアセットです。データは過去の記録ですが、適切に分析することで現在に関する知見が得られます。また成功している分野に注力することで、未来への道筋を描くことができます。データの適切な活用は健全なビジネスに不可欠ですが、あらゆるプロセスでこのような活用を実現するには、メタデータが鍵となります。なぜならメタデータは、一元化されアクションへと繋がるデータの全体的な（「360度」の）理解をもたらすからです。データを理解できなければ、分析やその他の目的においても、データを適切に活用できません。

ガバナンス

適切なガバナンスがないと、すぐにデータは雑然としたり、使いにくくなったり、壊れたり、さらには利用不能になる可能性もあります。メタデータにより優れたデータガバナンスが実現され、データの信頼性やセキュリティが大幅に改善されます。詳細は後述しますが、ここで重要なのは、MarkLogicには適切なデータガバナンスの実現に必要なツールや機能（出自やリネージのトラッキング、非常にきめ細かいのデータのセキュリティ保護など）が備わっているということです。

データの出自とリネージに関するメタデータは、データの出所、いつ誰が何を変更したかに関する情報を提供します。このようなメタデータを知っておくことは常に重要です。しかしETL処理によってデータが変更されるとこのような管理用メタデータが失われることもあります。メタデータ管理におけるこの問題は、元のソースデータとともに、データとメタデータをまとめて保存することで解決できます。データをどのように、どのような頻度で変更しても、メタデータが完全な監査証跡を提供します。

適切なメタデータ管理は、データセキュリティにとっても重要です。これにより、誰がどのデータにアクセスできるかを制御できます。メタデータを使って、ユーザー、ユーザーの役割、それらの役割に伴うパーミッション、またこれらが時間の経過とともにどのように変化するのかを定義できます。

適切なメタデータ管理は、規制要件への準拠の「実証」など、データ所有者に適用されるさまざまな規制要件を満たすうえでも重要です。特にビジネスルールや規制要件は変更されるため、元のデータとメタデータを一緒に保存しないと、データ統合プロセスを監査できません。ほとんどの組織は、使用しているデータがどこで生成されたか、どのように変更されたか、どのように使用されているかをはっきりと認識していません。規制機関や監査役から将来どのようなことを尋ねられるかは誰も（その規制機関や監査役でさえ）知らないのです。

データアーキテクトが注意すべき理由

データアーキテクトなら、データならびに「メタデータ」の一貫性に深く注意を払うべきです。しかし、エンタープライズ組織によく見られるようなサイロすべてにおいて適切なメタデータを作成・管理することは、非常に困難です。

> 特にビジネスルールや規制要件は変更されるため、元のデータとメタデータを一緒に保存しないと、データ統合プロセスを監査できません」

データアーキテクトはデータならびにビジネスの優れた世話役となる責任があります。具体的には、メタデータを使用して、効率性とアジャイル性の両方を実現します。特定 API による特定データへのアクセス方法やデータの管理方法を開発者が正確に把握していたら、あるいは、コンプライアンス担当が規制機関の要求に応じて直ちに取引履歴を確認できたら、あるいは、ビジネスアナリストが複雑なライセンス契約を即座に確認できたら、どれだけの効率性とアジャイル性を達成できるかを考えてみてください。データアーキテクトは、スマートメタデータ管理によりこれらすべてを実現できます。

スマートなメタデータ管理実現への障害

企業のデータ環境は雑然としています。データが複数サイロに分散しているため、アクセスしにくく時間もかかります。さらに悪いことに、現実世界では同一であるエンティティを表現する類似レコードが複数サイロに存在することがよくあります。このため、さまざまなデータベースを再同期するバッチジョブが頻繁に実行され、最悪の場合、データの整合性が完全に損なわれます。

今日、ほとんどのエンタープライズデータは、構造化データ用の RDBMS（リレーショナルデータベース管理システム）内にあります。このようなシステムは、構造を事前指定する厳密なスキーマを使用しており変更は困難です。このように柔軟性が欠如しているため、データアーキテクトは、スキーマの改善に時間を費やすのではなくスキーマをごまかしています。とりあえず動かすために、データを既存の列に無理やり入れたり、列やテーブルが場当たり的に追加されます。また、データ（およびメタデータ）が、事前に決められたスキーマに適合しないという理由で ETL ジョブ中に破棄されることもあります。リレーショナルデータベースには柔軟性がないため、データの品質に問題が生じ、またデータから最大限の価値を得ることができません。

いずれ場合でも、「被害者に非がある」わけではありません。つまりリレーショナルの手法に問題があるのであって、開発者やデータアーキテクトが悪いわけではありません。手持ちの唯一の道具が「SQL」なのであれば、各データ（およびメタデータ）はそれに応じた方法で扱われることになります。単純なことに、開発者やアーキテクトがデータの全体像をつかむために本当に必要なのは、メタデータを管理するための優れた戦略なのです。

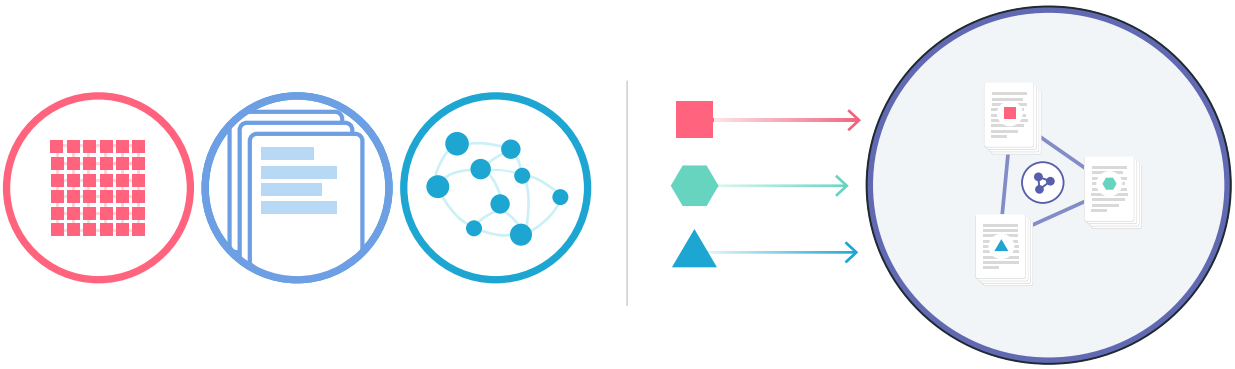


図2：**MarkLogic**データベースではマルチモデルを使用し、単一の統合プラットフォームですべてのデータとメタデータを一緒に統合します。

MarkLogicによるスマートなメタデータ管理

MarkLogic は、より上手く、早く、少ないコストでデータサイロを統合できるように設計されたエンタープライズ NoSQL データベースです。MarkLogic は、データおよびメタデータを統合するための優れたデータベースです。その理由として、マルチモデル手法によりデータを簡単に読み込めること、業界標準の API とクエリ言語でそのデータに素早くアクセスできること、またエンタープライズ仕様のセキュリティを備えていることが挙げられます。

マルチモデル手法、検索およびインデックス付け機能、データセキュリティという 3 つの要因がすべて組み合わさることで、MarkLogic はメタデータ管理のための優れた選択肢となっています。

マルチモデルアプローチ

「マルチモデル」データベースには、2 つの相互補完的な定義があります。

第 1 に、マルチモデルデータベースは、類似するビジネスエンティティの複数の異なるモデルを処理するものです。例えば、組織内の各業務部門が、ユーザーレコードを独自のスキーマ（モデル）で表現している場合があります。マルチモデルデータベースは、こういったさまざまなモデルを処理できるように設計されています。

第 2 に、マルチモデルデータベースは、単一の統合基盤を保持しながらも、複数のモデリング手法でデータを表現できます。MarkLogic の場合、モデリング手法として「ドキュメント」と「セマンティックトリプル」があります。これらのドキュメントモデルとトリプルモデルはいずれも、メタデータをデータ自体と一緒に格納できるため、メタデータ管理の実現に大きく貢献します。

それではドキュメントとトリプルの価値を確認するために、従来の RDBMS とマルチモデルデータベースにおける行の変更履歴フィールドを比較してみましょう。

データベース内の任意のレコードについて、いつ更新されたか（あるいはそもそも更新されたことがあるのか）を知りたいとします。リレーショナルモデルでこれを行うには、**last_updated** 列を追加したりするでしょう。リレーショナルデータはテーブルに保存されるため、1000 万行あれば、1000 万個のセルをデータに新規追加することになります。一部のレコードのみが更新される場合は、Null のセルが多数生じ（疎なデータ）、インデックスおよび統計量の更新が大量に必要となる可能性もあります。シンプルなタスクなはずなのに、大変な作業になってしまいます。

一方、MarkLogic はマルチモデルデータベースなので、データをドキュメントとして格納し、任意のドキュメント内の任意の枝（ブランチ）に新しい枝をいつでも追加することで、極めて柔軟にドキュメントを拡張できます。今回の例のように 10 個のレコードだけを更新する場合、該当する 10 件のドキュメントにだけ **last_updated** の値が追加され、残りの 999 万 9990 個のレコードに空のフィールドが追加されることはありません。

検索およびインデックス付け機能

MarkLogic がメタデータ管理に優れている 2 つ目の理由として、検索およびインデックス付け機能があります。MarkLogic のユニバーサルインデックスについては後ほど詳しく説明します。今のところは、MarkLogic では読

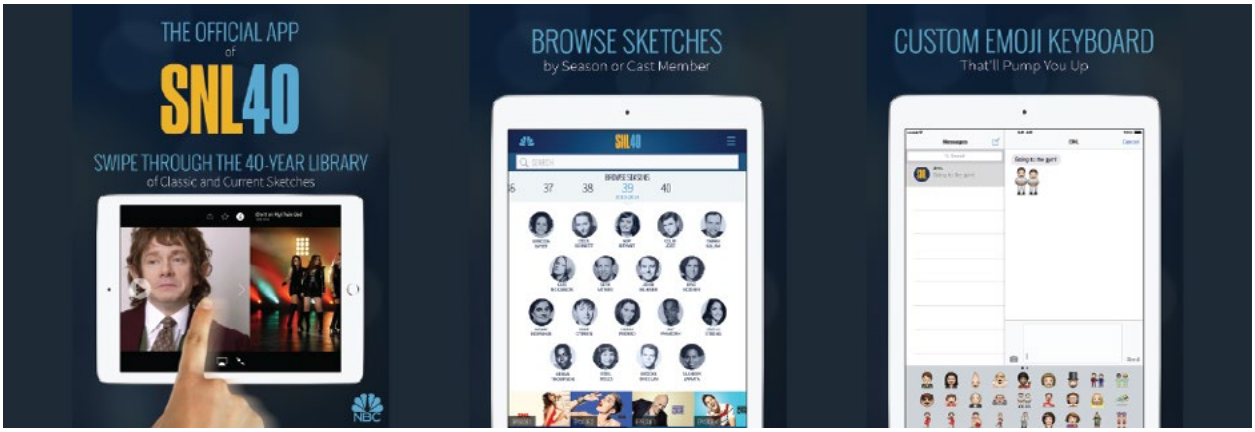


図3：**SNL 40**アプリは、**MarkLogic**データベースで生成されたスマートコンテンツをベースにしており、ファンは過去**40**年間のビデオの中からお気に入りの動画をすばやく簡単に見つけることができます。このアプリには、**MarkLogic**のセマンティックによるレコメンデーションエンジンが搭載されており、ファンの好みに合わせて、ファンがこれまで気づかなかった新しいキャラクターやコントも発見できます。

み込み時にすべてのデータおよびメタデータに対してインデックスを作成することを理解しておいてください。

優れたインデックス付けおよび検索機能により、メタデータ管理がどのように向上するのでしょうか？　常に問題となるのはスピードと効率性です。

業務が極めてサイロ化（分断）された状態でメタデータ管理を行おうとすると、メタデータへのアクセスが難しいという問題がよく起こります。「エンティティ」（顧客など）の全体像を把握したくても、情報が一か所にまとまっていないためできません。一か所にまとめられていたとしても、検索はできません。MarkLogic ではこのような問題は発生しません。というのもすべてのデータにインデックス付けされ、Google のように検索できるためです。データおよびメタデータを一緒にすぐ確認できるので、データモデリングとガバナンスのプロセス全体にとってプラスになります。

MarkLogic のユニバーサルインデックスも極めて高速です。ユニバーサルインデックスは MarkLogic データベースの中核であり、他の主要データベースに劣らない高速なクエリを実現できるように設計されています。これは特に、大規模なデータセットにおけるメタデータ管理において重要です。

データのセキュリティ

データは価値の高いアセットであり、不正アクセスから保護する必要があります。データを「外部」の侵入から保護しなければならないのは当然ですが、「組織内」からの不正アクセス対策も同様に重要です。

MarkLogic によるメタデータ管理には、適切なセキュリティ制御が含まれています。例えば、RBAC（ロールベースのアクセス制御）を簡単に実装できます（RBAC がメタデータ自体の一部となります）。RBAC では、適切な人が適切なタイミングで適切なデータを確認でき、許可されていない機密情報にはアクセスできないようにすることで、機密データを保護します。

この RBAC セキュリティは、MarkLogic が提供する組織全体でのメタデータ管理に最適です。MarkLogic ではメタデータが元のデータとともに保存されるため、ドキュメントレベル、さらにはサブドキュメントレベル（リレーショナルデータベースの「セル」レベルセキュリティと似ています）で、適切かつきめ細かなアクセスを簡単に提供できます。

MarkLogicによるメタデータサクセスストーリー

MarkLogic は、さまざまな業界において組織的なメタデータ管理の改善をお手伝いしています。特定のメタデータ（レコードの作成時期など）の取得は、業界を問わず重要です。しかし、各業界ごとに固有のメタデータとユースケースがあります（前述の金融サービスの IoT のように）。

ライフサイエンス業界では、リアルワールドエビデンスに関心が集まっています。この場合、病気情報、データの収集方法、患者情報などがメタデータとして表現できます。エンターテイメント業界の例としては、映画会社は、

> MarkLogicでは、時間とコストを大幅に節約できるだけではありません。データをそのまま読み込む機能と、MarkLogicで捕捉および生成されるメタデータは、データガバナンスにとって非常に重要です」

ある映画の監督や上映期間を知りたいと考えています。

エンターテイメント業界のもう一つの例として、MarkLogic のケーススタディを簡単にご紹介しましょう。NBC ユニバーサルは「The Saturday Night Live 40th Anniversary」アプリを MarkLogic で構築しました。これは、スマートなメタデータ管理によってビジネス価値がもたらされた一例です。このアプリの主なコンテンツはラージバイナリのビデオファイルですが、このアプリがユニークなのは、そのコンテンツに対するメタデータの活用法です。このアプリでは、各スキットに関する多様なメタデータ（レギュラーメンバー、ゲスト、放送日など）や、そのすべてのメタデータの相互関係から、検索 / ナビゲーションが可能なグラフを作成しています。このアプリにはレコメンデーションもあり、さらなる価値を提供しています。これは、この豊富なメタデータの組み合わせを活用して、視聴者が見たいと思うビデオを提案するものです。

統合ライフサイクルを通じた、よりスマートなメタデータ管理

よりスマートなメタデータ管理を実現するには、適切なアーキテクチャが必要です。MarkLogic のマルチモデルアプローチを活用する際の共通的なアーキテクチャパターンとして、「ODH（オペレーショナルデータハブ）」があります。MarkLogic の ODH アーキテクチャは、データやメタデータを管理するエンドツーエンドのプロセスを扱います。このアーキテクチャは、各プロセスつまり（1）データの読み込み、（2）データキュレーション、（3）セキュリティの適用、（4）データアクセスをサポートしています。

ODH パターンにより、アーキテクトは「ビジネスの観察（observe the business）」（分析、ビジネスアナリスト重視）と「ビジネスの遂行（run the business）」（取引、顧客対応）の両方に容易に対応できます。

データウェアハウスやデータマートには部分的なデータや時には古くて使えないデータ（しかも ETL ジョブが必要）も含まれますが、ODH ではデータやメタデータをリアルタイムで管理できます。またデータレイクでは生データを読み込んでもキュレーションは行いませんが、ODH ではデータおよびメタデータの読み込みとキュレーションの両方を行い、本当に業務で利用できるようにします。これに加えて ODH は、キュレーションしたデータやメタデータに対して、業界標準の API で非常に簡単にアクセスできます。マルチモデルデータベースである MarkLogic は、データをドキュメント、グラフ、構造化リレーショナルデータとして表現できます。

MarkLogic と ODH の概要がわかったところで、MarkLogicをスマートなメタデータ管理のプラットフォームとして使用する方法を確認します。データ統合のステップを詳しく確認していきます。

データの読み込み

MarkLogic では、あらゆるソースからのデータを「そのまま」読み込み、読み込み時にインデックスを付けます。この機能は、データ統合に非常に重要です。Oracle、SQL Server、Db2、メインフレーム、Hadoop などのソースからのデータはすべて MarkLogic に素早く取り込むことができます。その際、これに対応するメタデータのセットが付けられます。

このアプローチを、従来の RDBMS アプローチに基づく典型的なデータ統合方法と比較してみましょう。データアーキテクトは何か月（場合によっては何年）もかけて、既存のすべてのデータやスキーマの確認、現在のデータ構造を正確に反映する新しいスキーマの作成、データを新規データベースにコピーするためのあらゆる ETL ジョブの作成と実行などを行います。

また多くの場合、これとは別にメタデータも同様に処理しなくてはなりません。しかし ETL 構築中にソースデータのスキーマが変更された場合、再度ソースデータを確認したりさらなる ETL ジョブが必要になるなど、問題

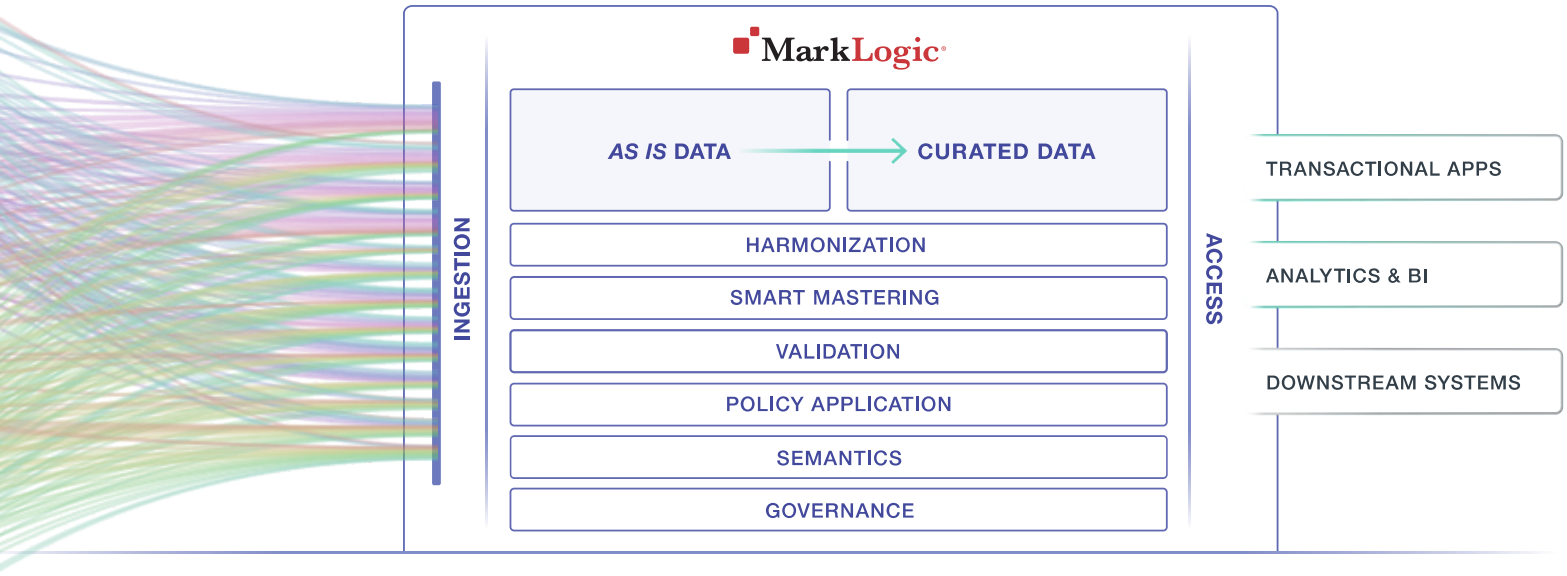


図4：**ODH**（オペレーショナルデータハブ）は、データの操作および分析に**MarkLogic**の主要機能を活用するエンタープライズアーキテクチャパターンです。これにより、データをそのまま読み込み、そのデータをキュレーションして、統合され活用可能な**360**度ビューを形成し、その**360**度ビューに簡単にアクセスできるようになります。

が累積していきます。リレーショナルデータベースでは、データの読み込みは繊細かつ厄介なプロセスです。エラーに影響を受けやすく、1回のミスで、大量の修正やプログラミングのやり直しが必要になる場合があります。

MarkLogicでは、時間とコストを大幅に節約できるだけではありません。データを「そのまま」読み込む機能と、MarkLogicで捕捉および生成されるメタデータは、データガバナンスにとって非常に重要です。この読み込みプロセスは対象部分以外に影響を与えず、ソースデータは変更されません。これはプロセスを追加していくものであり、完全に監査可能です。

メタデータは、作成されると元のデータと一緒に格納され、データ統合プロセス全体を通じて一緒に保持されます。これは従来のアプローチと大きく異なります。

それでは、これはどのように行われるのでしょうか？

MarkLogicの機能はドキュメントモデルに基づいています。ドキュメントモデルは極めて柔軟で、データを「そのまま」取り込んでから、キュレーション（整理）し、下流からアクセス可能です。「そのまま」のデータとキュレーションされたデータの両方が、MarkLogicの同一クラスタ内のデータベースに配置されます（図4参照）。

データキュレーションにおけるデータのハーモナイズは、ドキュメントモデルによって実現されています。ドキュメントモデルの柔軟性により、さまざまなソースからのデータのハーモナイズにおける反復処理が簡単かつ高速にできます。また、データのハーモナイズには「エンベロープパターン」というデータモデリング手法をお勧めします。これについてはこのセクションで詳しく説明します。簡単に言うと、読み込み時にXMLまたはJSONドキュメントとして格納されたオリジナルの生データを保持しながら、これをメタデータを含むエンベロープでラップするという手法です。

エンベロープのメタデータ部分には、必要とされる管理メタデータ、構造メタデータ、説明的メタデータなどが保持されます。通常、これには、レコードの適切なガバナンス、セキュリティ、使用に役立つメタデータが含まれます。例としては、レコードの作成時間、アクセスできる人を表すメタデータなどがあります。

次の例は、MarkLogicへの読み込み時に追加されたメタデータのエンベロープパターンの構造を示しています。

```
{
"envelope": {
      "metadata": [],
      "source": {
            "your data": "goes here"
    }
  }
}
```

次に、実際のデータを入力したときにエンベロープパターンがどのようになるかの実例を示します。

```
{
"metadata : {
      "Source" : "POS" ,
      "Date" : "2016-04-17" ,
      "Lineage" : "v01 transform" } ,
"source" : {
      "Customer_ID" : 2001 ,
      "Given_Name" : "Karen" ,
      "Family_Name" : "Bender" ,
      "Shipping_Address" : {
         "Street" : "324 Some Road" ,
         "City" : "San Francisco" ,
         "State" : "CA" ,
         "Postal" : "94111" ,
         "Country" : "USA" } ,
      "Billing_Address" : {
         "Street" : "847 Another Ave" ,
         "City" : "San Carlos" ,
         "State" : "CA" ,
         "Postal" : "94070" ,
         "Country" : "USA" } ,
      "Phone" : [
         {  "Type" : "Home" ,
            "Number" : "415-555-6789" } ,
         {  "Type" : "Mobile" ,
            "Number" : "415-555-6789" } ]
}
```

ご覧のように、エンベロープパターンそれ自体だけでなく、そこに含まれるメタデータにも価値があります。しかし、メタデータはさらにデータ統合プロセス全体に価値をもたらします。

データ読み込みは反復的なプロセスだということを念頭に置いておいてください。大企業では通常、データはさまざまなタイミングで読み込まれるため、これは厄介です。しかし MarkLogic では、これは問題になりません。というのも、読み込まれたデータが「そのまま」ステージングデータベースに格納されるため、複数の読み込みサイクルを異なるタイミングで実行できるからです。これは、バッチで、または統合プロセスを開始するトリガーによってリアルタイムで行うことができます。これらの機能は、データ統合プロセスの次の段階で重要になります。次の段階では、データをハーモナイズして、360 度ビューを提供します。

ユニバーサルインデックスをメタデータ管理に役立てるもう一つの方法

MarkLogic のユニバーサルインデックスでは、すべてのデータ（データのコンテンツや構造を含む）に対して読み込み時にインデックスを付けます。この結果、読み込まれたデータに対してすぐにクエリを実行できます。これはデータの発見（データディスカバリー）において非常に有用で、データの検証でも便利です。データの検証とは、文字どおり、読み込まれたデータの品質を一連のルールおよび条件で検証するプロセスです。ユニバーサルインデックスはこのプロセスで役立ちます。ユニバーサルインデックスにより、データアーキテクトは MarkLogic のデータ検証チェックを素早く実行できます。無効なデータを拒否することも、受け入れることもできます。受け入れる場合、確認が済むまで使用されないようにフラグを立てることや、利用を限定することもできます。あるいは、特定の人のみがアクセスできるようにし、それ以外の人には表示されないようにすることも可能です。データ検証プロセスが実行されると、メタデータが生成され、インデックスが付けられます。

データキュレーション

データは、1 回または複数サイクルの読み込みにより、MarkLogic のステージングデータベースに入ります。しかし、読み込みサイクルの終了までに、同じエンティティを表すレコードが複数存在するようになります。エンティティの包括的な全体像を本当に実現するには、エンティティのそれらのレコードをハーモナイズ（調整・調和）するプロセスが必要です。通常実行されるその他のステップとしては、検証、参照データの非正規化、重複排除、マッチング / マージ（MarkLogic のスマートマスタリング機能を使用して実行）、ポリシーの適用などがあります。これらを合わせたものが、「データキュレーション」プロセスになります。

MarkLogic によるデータ統合は、その柔軟なデータモデルと洗練されたインデックスにより、従来のリレーショナルよりもシンプルになります。

ユニバーサルインデックスがすでに作成されているため、エンティティの構成要素であるレコードを簡単かつ高速に比較できます。データアーキテクトにとって、このスピードのメリットは小さくありません。RDBMS では、「データ発見」のためにエンタープライズ規模のデータセットに対してクエリを実行した場合、数分、時には数時間もかかるためです。組織全体には多数のさまざまなエンティティがあるた

> データアーキテクトにとって、このスピードのメリットは小さくありません。RDBMSでは、『データ発見』のためにエンタープライズ規模のデータセットに対してクエリを実行した場合、数分、時には数時間もかかるためです」

め、この MarkLogic のスピードによって、データベースアーキテクトはキュレーションの時間を大幅に節約できます。

読み込んだレコードをすぐに確認できるため、データアーキテクトはすぐにデータを理解できます（従来は、読み込み前のデータの確認は時間がかかりすぎるため現実的ではありませんでした）。データを理解することで、すべてのデータを正確に反映するモデルを作成できるようになります。このため、メタデータを作成することにより、前述のすべてのメリットが実現されるのです。

エンベロープパターンにより、メタデータを作成・管理し、元データと一緒に格納できます。前述のように、各ドキュメントは読み込み時にエンベロープにラップされます。この時点で最初のメタデータが作成されます。この時点では、ドキュメントのメタデータは本質的に管理用のものであり、読み込みのタイムスタンプ、データソース、データ型、ユーザー指定のメタデータ（例：標準化されたファーストネームのラベル）などの情報となります。

データのハーモナイズ担当のデータアーキテクトが「データ発見」を通じてデータの理解を深めたので、次にドキュメントの特定部分をハーモナイズします。ハーモナイズ対象として追加するデータ項目は、データアーキテクトや実装されるビジネスルールによって決めます（制限します）。

続けて前述の例を使って説明します。顧客エンティティに関する統合マスターレコードを作成する際には、顧客の郵便番号や「ファーストネーム」（前の例で使用）など、特定のプロパティを選んで活用できます。しかし、エンベロープは、顧客に関する派生データや算出データ（その顧客の平均購入金額など）の保存にも使用できます。また、元データはレコードのコンテンツフィールドに格納しておくのが一般的ですが、顧客情報を確認しコンテンツの誤字を修正する場合には、修正前の元の値をエンベロープのハーモナイズ部分に格納しておくこともできます（この場合、データを保持すると同時に変換もしています）。

ハーモナイズに利用されたデータは、エンベロープ内の新しいセクションに入れられます。

```
{
  "envelope": {
      "harmonized": [],
      "metadata": [],
      "source": {
            "your data": "goes here"
    }
  }
}
```

次に、特定のプロパティ（「Zip」など）をハーモナイズ部分に活用した例を示します。

```
{
"harmonized" : { "Zip" : [ 94111 , 94070 ] }
,
"metadata : {
      "Source" : "POS" ,
      "Date" : "2016-04-17" ,
      "Lineage" : "v01 transform" } ,
"source" : {
      "Customer_ID" : 2001 ,
      "Given_Name" : "Karen" ,
      "Family_Name" : "Bender" ,
      "Shipping_Address" : {
         "Street" : "324 Some Road" ,
         "City" : "San Francisco" ,
         "State" : "CA" ,
         "Postal" : "94111" ,
         "Country" : "USA" } ,
      "Billing_Address" : {
         "Street" : "847 Another Ave" ,
         "City" : "San Francisco" ,
         "State" : "CA" ,
         "Postal" : "94070" ,
         "Country" : "USA" } ,
      "Phone" : [
         {  "Type" : "Home" ,
            "Number" : "415-555-6789" } ,
         {  "Type" : "Mobile" ,
            "Number" : "415-555-6789" } ]
}
```

最後に、エンベロープにセマンティックトリプルのセクションを含めることもできます（RDF トリプル形式のメタデータを格納）。

他のメタデータセクションと同様、セマンティックトリプルのメタデータは、読み込み時またはハーモナイズ時に作成できます。読み込み時に作成する場合は、トリプルはジョイン情報から生成できます。

ドキュメントモデルはデータを非正規化したものなので、このトリプルが役に立ちます。どちらの場合も、データのセマンティック（関係性）を検索などに活用するものです。この結果、データがよりリッチになり、高度なセマンティック検索などが実現されます。

セマンティックトリプルをエンベロープ内の別のセクションとしてモデルに追加した場合、構造は次のようになります。

```
{
  "envelope": {
      "harmonized": [],
      "metadata": [],
      "triples": [],
      "source": {
         "your data": "goes here"
      }
   }
}
```

次に、エンベロープパターンにトリプルが追加された例を示します。

```
{
"harmonized" : { "Zip" : [ 94111 , 94070 ] } ,
"metadata : {
      "Source" : "POS" ,
      "Date" : "2016-04-17" ,
      "Lineage" : "v01 transform" } ,
"triples" : [
{ "triple" : { "subject" : "Customer 2001"
, "predicate" : "placed" , "object" : "Order
8001" } } ,
{ "triple" : { "subject" : "Order 8001"
, "predicate" : "contains" , "object" :
"Product 7001" } } ,
"source" : {
      "Customer_ID" : 2001 ,
      "Given_Name" : "Karen" ,
      "Family_Name" : "Bender" ,
      "Shipping_Address" : {
         "Street" : "324 Some Road" ,
         "City" : "San Francisco" ,
         "State" : "CA" ,
         "Postal" : "94111" ,
         "Country" : "USA" } ,
      "Billing_Address" : {
         "Street" : "847 Another Ave" ,
         "City" : "San Carlos" ,
         "State" : "CA" ,
         "Postal" : "94070" ,
         "Country" : "USA" } ,
      "Phone" : [
         {  "Type" : "Home" ,
            "Number" : "415-555-6789" } ,
         {  "Type" : "Mobile" ,
            "Number" : "415-555-6789" } ]
}
```

MarkLogic のデータ統合プロセスにおけるセマンティックデータ管理の具体例については、プレゼンテーション「Effective Audit Trail of Data with PROV-O（PROV-O による効果的なデータの監査証跡）」をご覧ください。これは PROV-O（出自オントロジー）による出自メタデータの格納方法を説明しています。PROV-O は、機械可読な出自メタデータ記録の W3C 標準です。このホワイトペーパーでは PROV-O の詳細について説明しませんが、エンベロープパターンによって、PROV-O メタデータと参照元データを一緒に捕捉できるということを覚えておいてください。

まとめると、エンベロープパターンはデータとメタデータを一緒に保持するための優れた方法です。データ統合においては、メタデータを徐々に増やしていった方がメタデータ管理システムの立ち上がりが早くなります（すぐに価値が得られます）。「ビッグバン」的なプロジェクト（一度に全部を対象とする）では数年かかってしまいます。

データのセキュリティ保護

データやメタデータを一か所にまとめると、攻撃の格好の標的になりかねません。そのため、MarkLogic では

図5：同一ドキュメントを3つの異なる方法で表示した例。MarkLogicのセキュリティ制御では、異なるレベルのデータアクセスが提供されます。BI用または品質保証および開発用のエクスポートでは、データの匿名化、リダクション、またはその両方を行うことができます。

認証済みのきめ細かなセキュリティを提供し、今日のサイバー脅威から組織を守ります。MarkLogic は、最もセキュリティの高い NoSQL データベースと見なされています。また、リレーショナルデータベースのトップベンダーと同じセキュリティ認証を受けています。

では、MarkLogic の高度なセキュリティ機能はどのように活用できるのでしょうか？　また、さらに重要なことですが、データガバナンスの向上には、メタデータをどのように使用すればよいのでしょうか？

信頼性を確立するためには、この段階で以下のような難しい質問について考えることが重要です。

- 誰がデータを見る権限を持つのか？
- 誰がデータを変更する権限を持つのか？
- これらのパーミッションはどのくらいきめ細かくする必要があるのか？

信頼性に関するこれらの問題には、MarkLogic のメタデータ管理アプローチで対応できます。エンベロープパターンを使用すれば、メタデータの無制限の拡張と完全なガバナンスが可能です。すべてのレコードについて、誰がメタデータを変更できるか、および誰がメタデータを変更したかを示すメタデータをエンベロープ内に保持できます。

MarkLogic では、RBAC（ロールベースのアクセス制御）によって可視性と共有を管理することで、これを実現します。これにより、メタデータ管理用の信頼できる環境が構築されます。エンタープライズ規模の可視性は、スマートメタデータ管理に欠かせません。これにより、適切なガバナンスを適用した普遍的な「真実」を獲得できます。この際、リネージと出自を変更せずに監査可能な状態でデータを保持できます。

例えば、社会保障番号を持つ個人エンティティのレコードを考えてみましょう。社会保障番号は非常に機密性が高い個人情報で、その利用に関して法律および規制の適用対象となっています。MarkLogic のきめ細かいアクセス制御を使用して、このような重要なメタデータを保護することが重要です。BI 用または品質保証および開発においても、MarkLogic の匿名化およびリダクション機能によってデータを安全に共有できます。

データへのアクセス

読み込み、キュレーション、そしてもちろん、セキュリティ保護の結果、データとそのメタデータを業務ならびに分析で利用できるようになります。MarkLogic では、データアクセスを容易にするさまざまな業界標準 API が標準装備されています。業界固有または特別用途のアプリケーションにおいてカスタムのコネクタが必要になった場合でも、標準装備のものを基にして開発できます。

MarkLogic のメタデータ管理機能は非常に柔軟性が高く、あらゆる業界で使用できます。メタデータ処理には優れた柔軟性が必要だという点をさらに強調するために、MarkLogic のケーススタディを 2 つご紹介します。同じ業界ながら、メタデータの扱い方が大きく異なる 2 つの組織を取り上げます。

A 社は、優れた視聴体験を提供し、将来の映画製作やマーケティングに活用するため、自社のすべての映画を

深く理解したいと考えていました。A 社が必要としていたのは、特定のコンテンツ（カリフォルニアでのカーチェイスなど）を含む映画を探せる検索アプリケーションでした。A 社の場合、対象データはこれまでに公開した映画すべてとなります。この場合、メタデータは各映画のフレームごとのコンテンツです。先ほどのカーチェイスのシーンを検索した場合、MarkLogic を使用した A 社の検索アプリケーションでは、そのシーンがあるのはどの映画かだけでなく、開始何分後なのかもわかります。

一方、B 社は、ライセンス管理とコンテンツ再利用を改善するために、自社のすべての映画を深く理解したいと考えました。先ほどの例のようにマークアップされた映画コンテンツのメタデータを MarkLogic で捕捉するのではなく、ファイル形式、ファイルサイズ、上映時間、さまざまな言語版やその他のバージョンに関するメタデータを捕捉しています。これにより、B 社も MarkLogic で検索アプリケーションを構築しましたが、この検索アプリケーションの用途は、A 社とは非常に違っています。

用途は違いますが、どちらのアプリケーションでも、スマートなメタデータ管理プラットフォームとして MarkLogic の柔軟性とパワーを活用しています。

スマートなメタデータ管理による好循環

MarkLogic は、メタデータを格納および検索するための優れたプラットフォームを提供します。これによりメタデータの保守と更新が簡単になり、それによって、さらにプラットフォームの価値が高まるという好循環が生まれます。

この好循環は、MarkLogic の柔軟なデータモデルおよびインデックスによって可能になります。サイロ化されたリレーショナルシステムよりもずっと迅速に、数億個のレコードを読み込み、ハーモナイズできます。その後、より多くのデータおよびメタデータが追加されハーモナイズされるに従い、データ品質とビジネスの成果は改善し続けます。

MarkLogic データベースの総合的な価値は、データ量が増えるにつれ、ますます大きくなっていきます。ほとんどのデータレイクやデータウェアハウスでは、データが複雑化し、乱雑になり、サイロ化するのに伴い、次第に価値が低下していきますが、MarkLogic はその逆です。MarkLogic を使用することで、より迅速に、さらなる価値が得られます。それがスマートなメタデータ管理なのです。

主なリソース

Web ページ
MarkLogic ODH ソリューション

Web ページ
データハブフレームワークの概要

ブログ投稿
Making the Case for Semantic Metadata（セマンティックメタデータの説明）

MarkLogicについて

MarkLogic は、分断されたデータの統合に世界で最も適したベースです。この概要データシートでは、具体的な顧客事例など、MarkLogic の差別化要因について確認できます。

続きを読む

マークロジック株式会社 MARKLOGIC K.K.
150-0001 東京都渋谷区神宮前1-5-8 神宮前タワービルディング 13F
03 4540 0337 | jp.marklogic.com | MarkLogic-JP@marklogic.com

150-0001 東京都渋谷区神宮前1-5-8 神宮前タワービルディング 13F
03 4540 0337
jp.marklogic.com | MarkLogic-JP@marklogic.com