

進化するスキーマ全体に対する金融サービスの 規制報告

MODELDRとMARKLOGIC - データポイントモデリング

MARKLOGICホワイトペーパー ・ 2015年6月 ・ CHRIS ATKINSON



目次

規制当局を満足させることはますます困難に	1
データポイントモデリングの導入	1
MarkLogicデータベースプラットフォーム	2
ModelDRソリューションアーキテクチャ	3
MarkLogicとModelDRの統合	3
MarkLogicとModelDRのメリット	3
データサイロを削減	
データ品質の向上	
情報配信を加速	
ユーザーの情報発見を強化	
バイテンポラル機能を備えた規制報告の作成を迅速化	
スピーディにナレッジを増強	
さまざまなソース間でナレッジを発見	
まとめ	5

金融規制当局の要求の充足がますます困難に

金融規制の負担増などにより、銀行が資本コスト以上に稼ぐことが難しくなっています。SIB (国際的にシステム上重要な銀行) の多くも、規制要件のためにデータの集約、ガバナンス、アーキテクチャ、プロセスを改善しなくてはなりませんが、これはなかなか困難です。規制では、リスク管理、レポート提出、意思決定における改善が求められています。銀行は、データのガバナンスや品質を保証できません。というのもデータアーキテクチャの扱いやデータ集約が難しいからです。

金融市場データモデルは頻繁に変更されます。このため金融機関では、特定時点のデータを他のデータに合わせて調整し標準化すること(「congruence=統合/適合」と呼ばれます)が困難です。この問題を解決するため、多くの銀行ではETL (抽出と変換) 処理を導入し、業務部門ごとに分断された複数のデータサイロを繋ごうとしています。しかしこれによりデータの質が損なわれ、規制リスクが増える結果となっています。データ間の調整にかかる作業やコストは大幅に増加していますが、その大きな要因の一つはこういったETLの維持コストです。

複数のサイロの統合を行うため、情報の一部をリレーショナルデータウェアハウスに置き、スキーマに対してクエリを実行することがよく行われています。これらのスキーマの変更には、かなりの作業が必要です。スキーマを変更する場合、これをテストし、クエリで検証し、管

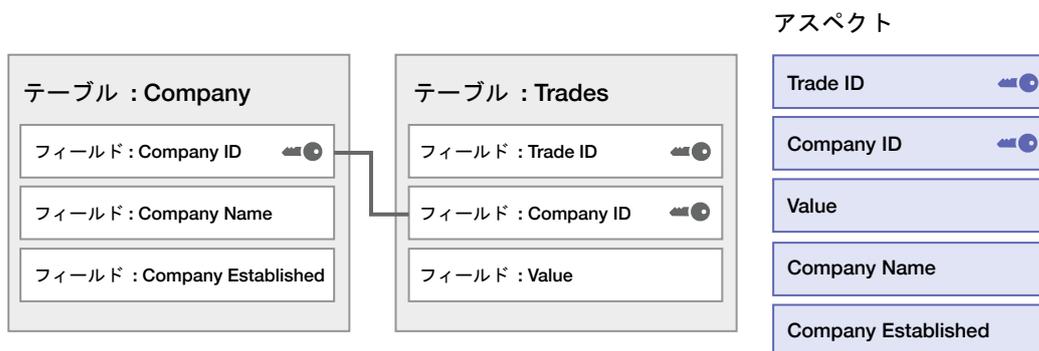
理プロセスもそれに応じて変更しなくてはなりません。その後ユーザーがデータを利用できるようにするため、データの再処理が必要ですが、それには数時間、数日、数か月かかることもあります。また複雑なデータ環境を視覚化するツールがないため、情報から知見が得られないことが多々あります。

時間が経過するほど、データの統合は困難になります。というのも、データ視覚化の処理はETL処理内に固定化されて実装されているからです。データはずっと変化し続けます。このため、こういった処理の保守・維持はますます複雑かつ困難になってしまいます。

データポイントモデリングの導入

従来のリレーショナルデータモデリングよりもはるかに優れたアプローチとして、データポイントモデリング(DPM)があります。データポイントモデルはXBRL (eXtensible Business Reporting Language) 準拠で、データモデルと出力形式を分けて扱えるようにしています。これにより特定のデータや柔軟性もたらされます。これを使って、リレーショナルなデータ構造を非リレーショナルなデータポイントモデルにリエンジニアリングすることもできます。この手法では、テーブル内のフィールドを分析して一連のアスペクト(「ファセット」)を作ります。これらのアスペクトの作成の際に、異なるアスペクト間の関係性を記録することで、参照整合性を実現しています。(図1)

図1



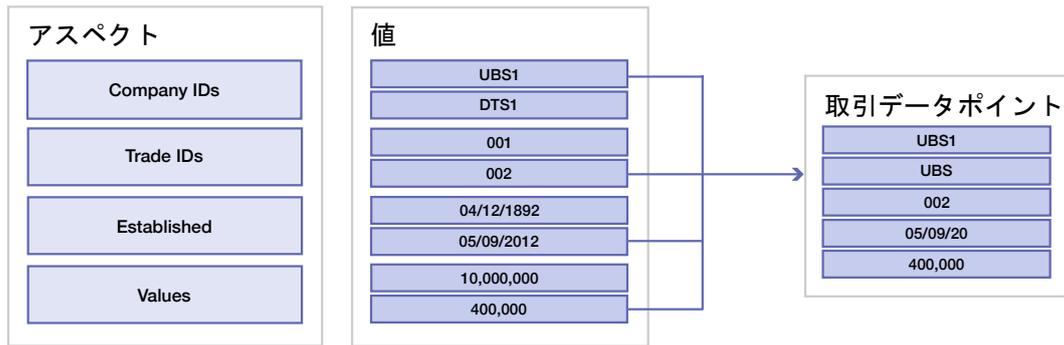


図2

それぞれのアспектに関して、異なる値（離散値）すべてを含むリストが生成されます。これらはすべて値として格納されます。各値は一度だけ記録されます。その後、データポイントがこれらのアспект内の値に関連付けられます。（図2）

アспект作成の際には、さまざまなソースからのデータを統合するための情報も記録されます。この統合作業を説明するために、次のような例を考えてみましょう。ある同一イベントに関するデータがさまざまな場所に記録されています。これらのデータでは、形式やモデルが異なっていることが想定されます。データを統合するため、対象となるデータは一か所にまとめられます。このデータのモデルと形式は、ETLによって共通的なデータモデルに標準化されるのが一般的です。すべてのソースシステムからのデータがきちんと調和していることが確認されると、このデータは統合されたこととなります。つまり、この時点におけるデータ像は完全であり、欠けている情報はなく、全体像が得られた、ということになります。

DPMアプローチの主なメリットは、以下のようになります。

- 分散しているデータセットの統合が素早く行える。複雑で業務リスクが大きい処理を行う必要がない
- DPM内で変換された情報には、参照整合性がある
- データのタクソノミーをインポートすることで、このモデルにインポートされている他のデータとの融合・結合ができる
- 厳密かつ固定的に定義されたスキーマに対してクエリを実行するのではなく、任意のデータタクソノミーを使ってデータを素早く確認できる

しかしこのアプローチには欠点があることが、従来から指摘されてきました。柔軟なDPMからの情報を表現するには、非リレーショナル形式であるRDF (Resource Description Framework) トリプルが良いと言われます。しかしこれによって更なる問題が発生するのです。使用中のデータベース以外のソースデータに基づいてRDFトリプルを生成した場合、生成されたRDFトリプル内の情報は、ソースデータと分断されています。このため、この情報を後から変更する場合、ソースデータの再処理が必要となります。ここで本当に必要なのは、ソース情報ならびにRDFトリプルを使用する際に、再処理が不要なメカニズムです。つまり金融レポートの形式が変更された場合でも、新しいRDFを生成でき、必要な調整は内部的に行えるメカニズムが必要なのです。

幸いなことに、こういった従来の問題を解決できるデータベースプラットフォーム/モデリングソリューションが今では存在しています。

- MarkLogic®は、非リレーショナル/マルチモデルのデータベースプラットフォームで、RDFトリプルストアとドキュメントデータベースが統合されています。元のソースデータはネイティブの形式で格納され、生成されたRDFと一緒に保持されます
- ModelDRIは、RDFの生成とマッピング、その後の変換デザインのMarkLogicへの展開を、視覚化し管理するためのデータポイントモデリングソリューションです

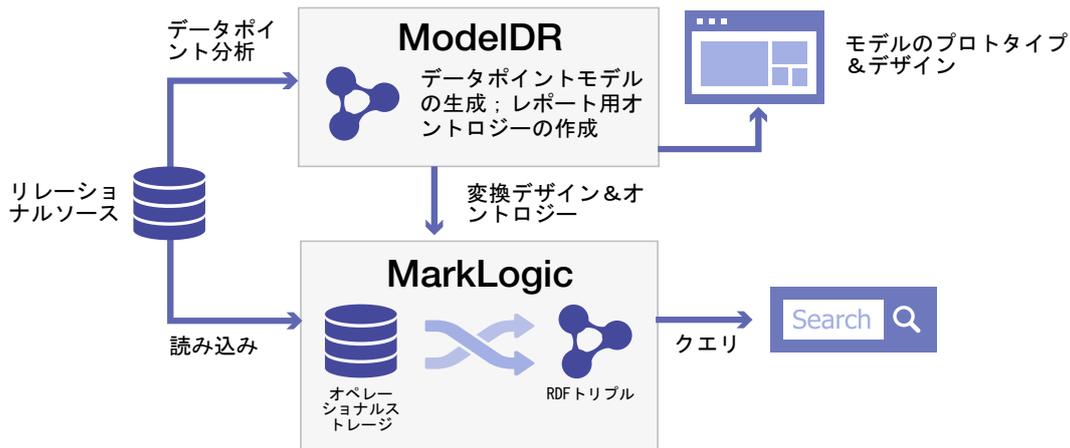


図3

MARKLOGIC データベースプラットフォーム

MarkLogicはエンタープライズ仕様のNoSQLデータベースプラットフォームです。現代のデータ(ドキュメント、関係性、メタデータなど)を扱うように設計されています。JSON、XML、RDFなどをネイティブに格納し、素早くクエリを実行できるデータベースはMarkLogicだけです。この強力なプラットフォーム1つだけであらゆるデータに対応できます。これはドキュメント指向データモデルで、スキーマに依存しません。このため新しく読み込まれるデータをテーブル形式に押し込む必要はなく、データを柔軟にモデリングできます。MarkLogicでは、トランザクションは「アズイズ(そのまま)」の自然な形で格納されます。またデータ変換を行わないためETL処理が不要です。このため変換によるデータの信頼性/コンテキストの損失が発生しません。複数ソースからのデータの読み込み、また時間の経過に伴う変化への対応が、より楽になります。

MODELDRソリューションのアーキテクチャ

ModelDRは、リレーショナルなデータソースをDPMにマッピングするGUIを提供します。このDPMは、その後オントロジーとして表現されます。次に、このオントロジーを使って、RDFトリプル作成用の変換マップを生成します。この変換マップは、RDFオントロジー対応の他の任意のデータベースでも利用できます。ModelDRは、既存の業界オントロジー(FIBOなど)に基づくテンプレートを提供します。これらのテンプレートとDPMと一緒に使って、オントロジーならびに変換処理のためのマップを生成できます。ModelDRでは、各ユーザーが所有する独自のビジネスオントロジーも利用できます。

MARKLOGICと MODELDRを統合する

ModelDRは、フロントエンドのUIが備わった処理エンジンを提供します。これにより、リレーショナルなソースの分析、オントロジーのインポート、リレーショナルデータ/RDF変換のデザインを行えます。

MarkLogicデータベースプラットフォームとModelDRの統合は、次のように行います。(図3)

- DPMを生成する
MarkLogicと連携したModelDRでは、リレーショナルモデルを分析し、これをDPMに変換します。この際に、ModelDRが提供する一般的な金融業界オントロジーテンプレートを利用できます。あるいはユーザー独自のタクソノミーを利用できます
- デザインレイヤーを作成する
データ統合は、DPMレイヤーで行います
- 変換をデザインする
ModelDRは、SPARQL(RDF用のクエリ言語)に基づく変換用に、アーキテクチャに関するデータデザインを行います
- 変換デザインを読み込む
MarkLogicはこの変換デザインを利用して、ソースからRDFトリプルを作成します。この処理は、後から行うこともできますし、あるいはデータベースへの読み込み時に行うこともできます
- データへのクエリ(活用)
生成されたDPMオントロジーとSPARQLクエリと一緒に使って、データに対してクエリを実行します

MARKLOGICとMODELDRを一緒に使うメリット

MarkLogicは、ModelDRに対して柔軟でアジャイルなデータベースプラットフォームを提供します。また「スキーマ・オン・リード（出力時のスキーマ適用）」アプローチにより、複数のデータモデルを1つのオペレーショナルデータストアに集約する際の問題を解消します。これらを一緒に利用するメリットはたくさんあります。

ModelDRは、MarkLogicに対してレポート作成用オントロジー/タクソノミー/データの複雑な関係性を生成/確認するための、業界に特化したわかりやすいメカニズムを提供します。必要な出力の生成/操作/確認を、より短期間かつ正確に行えます。

データサイロの削減

MarkLogicは、複数のデータモデルが利用可能な共通オペレーショナルデータストアを提供します。このマルチモデル機能により、ソースのデータモデル変更に伴って必要な作業が大幅に削減されます。ソースデータを「アズイズ（そのまま）」で読み込めるので、MarkLogicを一つの統合データハブとして活用できます。データを無理に変換する必要はありません。

データの品質改善

データはMarkLogicにネイティブの形式で読み込めるので、大掛かりな変換処理は不要です。これに加えてデータサイロも削減されるので、「シャドーIT」機能の必要性が大幅に減ります。ユーザーが自分たちだけでソースデータを読み込んだり、業務に応じてデータを変換し独自のサイロに格納したりすることは、頻繁に行われています。通常、こういった「シャドーIT」が発生するのは、データモデルに制約がある、またセントラルデータハブへのデータ提供に時間がかかるからです。MarkLogicのマルチモデル機能により、ユーザーは自分のデータモデルの柔軟性を犠牲にすることなく、セントラルサービスを利用できます。ModelDRが生成した変換デザインは、読み込みの際に活用できます。あるいは後から利用して、レポート作成に必要なRDFを生成することもできます。これによって、元のソースデータが変更されることはありません。共通データハブでは、オリジナルデータとそこから派生したデータを一か所で確認できます。これらのデータの入出力は簡単に行えます。

迅速な情報提供

MarkLogicデータベースプラットフォームのさまざまな機能により、情報に関する作業の全体（作成から利用ま

で）が容易になります。スキーマ変更を心配する必要がなく、データ形式が変更されてもすぐに読み込めます。また業務プロセスから情報を生成できます。固定的なリレーショナルスキーマが存在しないので、データの大掛かりな再処理が不要です。またETL処理も不要です。というのも従来、業務のアジャイル性のために必要とされた複数のサイロがもはや不要だからです。

情報発見のお手伝い

ModelDRが生成したオントロジーは公開され、必要な結果が簡単に得られます。情報のどの部分が必要となるのかを事前に知っておく必要はありません。情報のディスカバリー（発見）業務において、オントロジーを使いながら自由に情報にアクセスできます。

バイテンポラルにより規制レポート作成が容易に

MarkLogicのプラットフォームには、「バイテンポラル」（2つの時間軸を利用する）クエリ機能が備わっています。これによって極めて厳しい規制要求にも対応できます。複数の異なるデータモデルやソースからの情報に関して、過去の特定時点の状況を確認するクエリを実行できるため、データ間の調整作業が削減されます。既存のModelDRテンプレートや業界/業務オントロジーを利用することで、必要なデータの全体像を容易に把握できます。結果的に、特定の規制レポートの生成作業が大幅に削減されます。

ナレッジを迅速に拡大

RDF技術により、データのコンテキストに基づいて新しい情報を作成できます。「推論」では、データポイント間の明示的な関係を使って、明示的な関係性を新しく作成します。これによりナレッジ（知識）が増えます。（図4）

ModelDRは軽量なモデリングソリューションです。データを無理に変換せず、「インプレース」（その場所においたまま）で利用します。これにより、ユーザーエクスペリエンスがよりアジャイルになります。またデータからすぐに価値が得られます。この「インプレースモデリング」は、MarkLogicの「アズイズで格納」機能を完璧に補完します。

一般に、データからのナレッジを増やすと、データ自体が爆発的に増えます。これはデータモデル内のさまざまな場所に重複するデータが存在するようになるからです。DPMは、値を複数のデータポイントで利用することで、データの増殖を抑えます（増殖を抑えるために、値を保存して複数回再利用して結果を作成します）。

推論の例



図4

すべてのソースを活用してナレッジを発見する

構造ならびにメタデータがデータ自体の中に明示的に存在しているため、ナレッジに基づいたクエリが利用できます。データベースの構造を変更せずに、規則や関係性を利用できます。こういった新しい関係性は、その後MarkLogicデータベースレイヤー内でRDFとして表現されます。DPMにより、ModelDRはあらゆるデータスキーマを理解できます。スキーマを把握したあと、ソリューションをMarkLogicデータベースプラットフォームに読み込みます。MarkLogicでは1つのクエリで、ModelDRが作成したRDFならびに元ソースから情報を抽出できます。この例としては、PDF形式の金融レポートを読み込み、これにデータポイントのRDFを追加する、というものがあります。この場合、PDFドキュメントをネイティブ形式で読み込み、セマンティックのRDFを付けます。これらのデータポイントからは、ソースデータを簡単に辿れます。これは特にディスカバリー（発見）に基づく分析において便利です。

まとめ

今日、銀行にとって、データのガバナンスと品質管理を効果的に行うことはますます困難になってきています。これは、金融市場モデルが頻繁に変更されることにより、特定時点のデータを他のデータに合わせて調整し標準化すること（「congruence＝統合/適合」と呼ばれます）が困難になっているためです。

リスク管理、規制レポート、効果的な意思決定においては、データアーキテクチャやデータ集約が障害となっています。この問題への効果的な解決策として、データポイントモデリング（DPM）があります。しかし、リレーショナルデータベースを使ってDPMを行った場合、依然として大量のETLが必要です。RDFトリプルとドキュメントモデルを組み合わせることで、データモデリングの時間を数か月（あるいは数年）単位で短縮できます。

MarkLogicデータベースプラットフォームは、RDFならびにその生成元であるオリジナルの取引データの両方を格納できるという点でユニークです。ModelDRソリューションにより、情報をネイティブ形式で読み込むことができ、その後、新しく得られた情報をトリプルとして公開できます。補完的なこれらの技術を組み合わせることで、極めてアジャイルなソリューションが実現されます。これにより複数のデータモデルやソース全体から金融レポートを効率的に作成できます。

金融サービス業界におけるエンタープライズNoSQLソリューションの詳細については、[弊社ウェブサイト](#)をご覧ください。



マークロジック株式会社 MARKLOGIC K.K.

150-0043 東京都渋谷区道玄坂 1-12-1 渋谷マークシティウエスト 22 階 | +81 3 4360 5354
jp.marklogic.com | MarkLogic-JP@marklogic.com