

Delta Lake シリーズ レイクハウス

データレイクとデータウェアハウスの
「いいとこ取り」

 databricks



この eBook の概要

Databricks の Delta Lake の eBook シリーズは、データを扱う方が Delta Lake のフル機能を理解して利活用するための支援を目的として提供されています。

この eBook 「Delta Lake シリーズ : レイクハウス」では、レイクハウスのケイパビリティについて詳しく解説します。

学べる内容

Delta Lake のケイパビリティの詳細と、そのケイパビリティが性能を大幅に向上させる仕組みを理解できます。

目次

序章

Delta Lake とは

01

レイクハウスとは

02

レイクハウスと Delta Lake の内部機能

03

Delta Engine とは



Delta Lake とは

[Delta Lake](#) は、データの信頼性を高め、迅速な分析をクラウドのデータレイクにもたらし統合データ管理システムです。既存のデータレイク上で動作し、Apache Spark™ API と完全な互換性があります。

Databricks では、Delta Lake がデータレイクにもたらし信頼性、性能、ライフサイクル管理を実証してきました。Databricks を導入したお客様は、Delta Lake を活用することで、不正なデータ取り込み、コンプライアンスのためのデータ削除の煩雑さ、データ収集時のデータ欠落など、さまざまな課題を解決しています。

Delta Lake は、高品質データをデータレイクに迅速にもたらし、セキュアでスケーラブルなクラウドサービスで、チームによるデータの利活用を加速させます。



Chapter

01

レイクハウスとは？

01

レイクハウスとは？

Databricks ではここ数年、新しいデータ管理アーキテクチャ「レイクハウス」が、多くのお客様やユースケースにおいて独自に発展するのを見てきました。この章では、この新しいアーキテクチャと、かつてのアプローチと比較して優れている点について解説します。

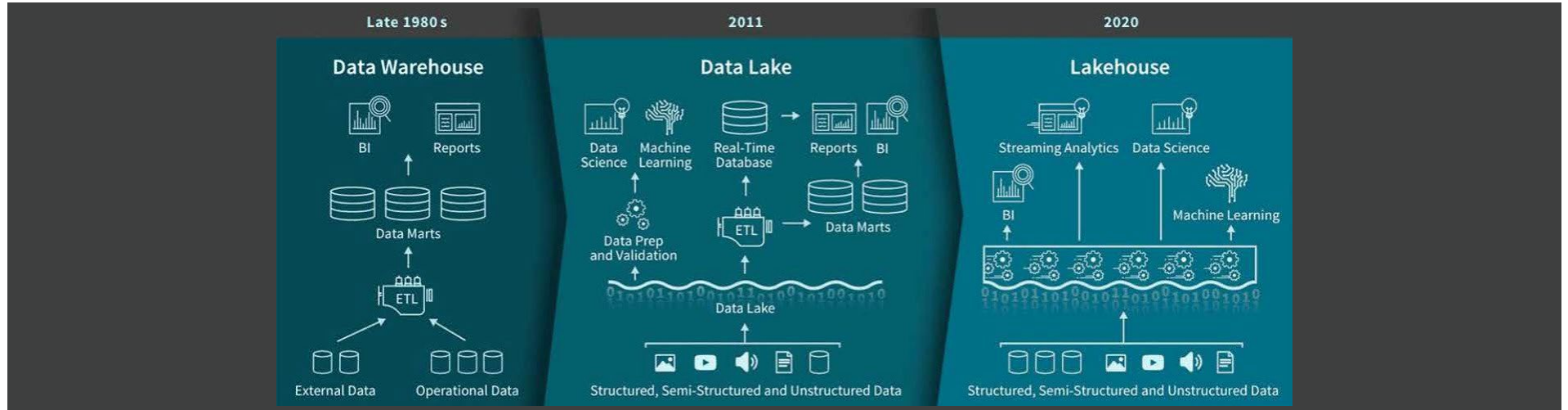
データウェアハウス（DWH）はこれまで、意思決定支援やビジネスインテリジェンス（BI）アプリケーションにおいて広く利用されてきており、これには長い歴史があります。データウェアハウスの技術は、1980年代後半の登場以来進化を続け、MPP アーキテクチャによって、より大規模なデータ処理が可能なシステムがもたらされました。

しかし、データウェアハウスには、構造化データの処理には適しているが、非構造化・半構造化データの処理には適していないという問題があり、近代ビジネスが必要とする高速で多様なデータの大規模処理においては課題がありました。データウェアハウスは、そういった最新のデータ処理を前提としたユースケースには適しておらず、コスト効率的にも最適なソリューションではありません。

多くの企業が、複数のソースから膨大なデータを収集するようになり、アーキテクト集団による単一システムの構想が生まれました。さまざまな分析プロダクトやワークロードのデータを単一のシステムに集約するという構想です。

10年ほど前から、多様な形式の未加工データ用レポジトリとして[データレイク](#)（DL）の構築が始まりました。データレイクは、データの格納には適していましたが、トランザクションのサポートやデータ品質の保証といった重要な機能が欠けていました。一貫性と分離性の欠如により、アペンドと読み取り、バッチとストリーミングジョブを混在させることはほぼ不可能でした。データレイクは、これらの理由から本来の目的の達成には至っておらず、データウェアハウスのメリットもほとんど活かされていないのが実情です。





一方で、柔軟で高性能なシステムに対するニーズが高まっており、多くの企業が、SQL分析、リアルタイムの監視、データサイエンス、機械学習など、多様なデータアプリケーションに対応するシステムを必要としています。AIを最大限に活用するには、非構造化データ（テキスト、画像、動画、音声など）の処理能力を高めることが不可欠ですが、これらはまさに、データウェアハウスが得意としないデータ形式です。

非構造化データの処理には、一般的に複数のシステムが使用されます。データレイクと複数のデータウェアハウスに加えて、ストリーミング、時系列、グラフ、画像データベースなどの専用システムが導入されます。複数システムの併用は、システム間のデータの移動やコピーを必要とし、複雑さと遅延の原因となります。

レイクハウスはDLとDWHの「いいとこ取り」

レイクハウスは、データレイクとデータウェアハウス両方の利点を取り入れた新しいデータアーキテクチャです。データウェアハウスに類似したデータ構造とデータ管理機能を、データレイクに使用される低コストのストレージに直接実装しています。レイクハウスは、安価で信頼性の高いストレージ（オブジェクトストア形式）が利用可能になった今、最新のニーズに対応するデータウェアハウス構築のための最善策といえるでしょう。

レイクハウスの主な特長は次のとおりです。

トランザクションのサポート

エンタープライズにおけるレイクハウスでは、多くのデータパイプラインがデータの読み取り・書き込みを同時に行います。ACIDトランザクションのサポートにより、SQLなどを利用したデータの読み取り・書き込みを複数のユーザーが同時に実行する場合でも一貫性が保たれます。

スキーマの適用とガバナンス

レイクハウスは、Starスキーマ、SnowflakeスキーマなどのDWHスキーマの適用と進化をサポートすることを期待されています。システムは[データの整合性を判断するケイパビリティ](#)および、堅牢なデータガバナンスと監査メカニズムを備えていなければなりません。

BIツールをサポート

レイクハウスでは、ソースデータに対して直接BIツールを使用できます。これにより、データの陳腐化の抑制・最新性の向上、遅延の低減を可能にし、データレイクとデータウェアハウスの両方で複製データを重複して保持することによるコストを不要にします。

コンピューティングとストレージを分離

具体的には、ストレージとコンピューティングが別々のクラスタを使用することを意味します。結果として、同時に利用するユーザー数やデータサイズの増大にあわせた拡張が容易になります。最新のデータウェアハウスの中には、このような特長を持つものもあります。

オープン性

レイクハウスでは、Parquetのようなオープンで標準化されたストレージ形式が使用されています。APIを提供し、機械学習やPython/Rライブラリなど、さまざまなツールやエンジンからデータへの効率的な直接アクセスを可能にしています。

構造化から非構造化まで、あらゆるデータをサポート

レイクハウスでは、画像、動画、音声、半構造化データ、テキストなど、データのタイプを問わず、最新のデータアプリケーションに必要なあらゆるデータの保存、調整、分析、アクセスが可能です。

多様なワークロードをサポート

レイクハウスは、データサイエンス、機械学習、SQL、分析など、さまざまなワークロードをサポートします。ワークロードの種類によっては専用のツールが必要な場合もありますが、その場合でも、同じデータリポジトリが使用されます。

エンドツーエンドのストリーミング

多くの企業がリアルタイムのレポートを当然のこととして認識しています。レイクハウスではストリーミングをサポートされており、リアルタイムのデータアプリケーション専用のシステムを別途用意する必要がありません。

上記の特長の他に、エンタープライズグレードのシステムでは追加機能が必要となります。例えば、セキュリティやアクセス制御の機能です。昨今のプライバシー規制に対応するためのデータの監査、保持、リネージなどのデータガバナンス機能は不可欠になっており、データカタログやデータ使用量に関するメトリクスなどのデータを抽出するためのツールも必要です。レイクハウスでは、このようなエンタープライズ機能を単一のシステムに対して実装、テスト、管理するだけですみます。



リサーチペーパー

Delta Lake: High-performance ACID Table Storage over Cloud Object Stores

Delta Lake : クラウドオブジェクトストアによる高性能 ACID テーブルストレージ

要旨

Amazon S3をはじめとするクラウドオブジェクトストアは、極めて大規模でコスト効率の高いストレージシステムであり、大規模なデータウェアハウスやデータレイクを保存するための主要な存在となっている。しかし、key-value 型ストアの特性により、ACID トランザクションや高い性能を実現することは困難である。オブジェクトのリスト化などのメタデータ操作は高コストで、一貫性の保証は十分でない。このリサーチペーパーでは、Databricks により開発された、クラウドオブジェクトストア上に展開するオープンソースの ACID テーブルストレージレイヤーである Delta Lake を取り上げる。Delta Lake は、Apache Parquet 形式に圧縮されたトランザクションログを使用し、ACID プロパティ、タイムトラベル、大規模な表形式のデータセットに対するメタデータ操作を大幅に高速化する（例：クエリにおいて数十億の表パーティションを素早く検索）。また、このデザインを利用して、データレイアウトの自動最適化、アップサート、キャッシュ、監査ログなどの高度な機能を提供している。Delta Lake テーブルへは、Apache Spark、Hive、Presto、Redshift などのシステムからアクセスできる。Delta Lake は、エクサバイト規模のデータを日々処理する数千社の Databricks の顧客に利用されており、最大のユースケースでは、エクサバイト規模のデータセットと数十億のオブジェクトの運用に活用されている。

著者

Michael Armbrust、Tathagata Das、Liwen Sun、Burak Yavuz、Shixiong Zhu、Mukul Murthy、Joseph Torres、Herman van Hövell、Adrian Ionescu、Alicja Łuszczak、Michał Szafranski、Xiao Li、Takuya Ueshin、Mostafa Mokhtar、Peter Boncz、Ali Ghodsi、Sameer Paranjpye、Pieter Senster、Reynold Xin、Matei Zaharia

※リサーチペーパーの全文は、[こちら](#)からダウンロードできます。





初期段階のレイクハウス

[Databricks のプラットフォーム](#)は、レイクハウスのアーキテクチャを備えています。Microsoft の Azure Synapse Analytics サービスは、[Azure Databricks と統合](#)することで、同様のレイクハウスパターンを可能にします。BigQuery や Redshift Spectrum などのマネージドサービスは、前述のレイクハウス機能の一部を提供しますが、主に BI や SQL アプリケーションを念頭に置いたものです。

独自のシステムを構築する場合は、レイクハウスの構築に適したオープンソースのファイル形式（Delta Lake、Apache Iceberg、Apache Hudi）を利用できます。

データレイクとデータウェアハウスを単一システムに集約することで、データを扱う各チームが複数のシステムにアクセスする必要がなくなり、業務が迅速化します。初期のレイクハウスでも、エンタープライズ向けの通常のデータウェアハウスに対しては十分なレベルの SQL サポートと BI ツールとの統合を提供します。マテリアライズドビューとストアードプロシージャも利用可能ですが、ユースケースによっては、従来のデータウェアハウスとは異なる他のメカニズムが必要になります。ストアードプロシージャは、従来の商用データウェアハウスとほぼ同一のセマンティクスを達成するシステムを必要とする「[リフト&シフトのシナリオ](#)」において特に重要です。

他のタイプのデータアプリケーションのサポートについて少し触れます。レイクハウスのユーザーは、データサイエンスや機械学習のような非 BI ワークロードのためのさまざまな標準ツール（[Apache Spark](#)、Python、R、機械学習ライブラリ）にアクセスできます。データ探索や絞り込みは、分析およびデータサイエンスに関する多くのアプリケーションで使用されます。Delta Lake は、レイクハウス上のデータの品質を、データ使用時まで継続的に改善するように設計されています。

ここで、技術面の特記事項を補足します。レイクハウスでは、ストレージレイヤーには分散ファイルシステムの使用が可能ですが、一般的にオブジェクトストアが使用されます。オブジェクトストアは、低コストで可用性の高いストレージを提供し、大規模な並列の読み取りに強みを発揮します。これは、最新のデータウェアハウスの必須要件です。

BI から AI へ

レイクハウスは、あらゆる業務に機械学習が取り入れられる時代に、企業のデータインフラをシンプルにしてイノベーションを加速させる新しいデータ管理アーキテクチャです。企業はかつて、運用システムから抽出した構造化データを製品開発や意思決定に利用していましたが、今日では、コンピュータビジョンやスピーチモデル、テキストマイニングなどの手法を用いて製品や意思決定に AI を活用しています。そこでデータレイクではなくレイクハウスが選ばれるのですが、その理由は何でしょうか。レイクハウスでは、非構造化データを扱う場合においても重要な、データのバージョン管理、ガバナンス、セキュリティ、ACID プロパティを提供できることが挙げられます。

レイクハウスは現在、コスト削減に成功している一方で、性能に関しては、長年の実績のある特定の目的に特化したデータウェアハウスなどのシステムと比較して劣る傾向があります。レイクハウスは、BI ツール、IDE、Notebook などのツールを好むユーザーも考慮して、UX および、主要なツールへのコネクタを改良し、さまざまなペルソナに訴求できるようになる必要があります。レイクハウスのテクノロジーは今後も成熟し、シンプルさとコスト効率、多様なアプリケーションへの対応といった特長を活かしつつ、さまざまな課題をクリアしていきます。

A photograph of two construction workers in safety gear. The worker on the left is wearing a white hard hat with a red logo, a black neck brace, and a blue work jacket. The worker on the right is wearing a red long-sleeved shirt, a red and white patterned scarf, and a grey safety vest. They are both looking down at a large sheet of paper (blueprints) held by the worker on the left. The background is a blurred construction site.

Chapter

02

レイクハウスと Delta Lake の内部構造

02

レイクハウスと Delta Lake の内部構造



Databricks は以前の [ブログ](#) で、企業におけるレイクハウス（LH）採用の増加状況について解説しました。このブログの内容は、技術系のオーディエンスから大きな反響がありました。多くの方がレイクハウスを次世代のデータアーキテクチャとして賞賛してくださったのですが、データレイクと何ら変わらないのではというご意見もいただきました。そんななか、Databricks のエンジニアと創業者が、データレイクとは一線を画すレイクハウスパラダイムを核とする技術的課題とソリューションについてのリサーチペーパー「Delta Lake: High-performance ACID Table Storage over Cloud Object Stores（Delta Lake：クラウドオブジェクトストアによる高性能 ACID テーブルストレージ）」を共同執筆しました。このペーパーは、大規模データベースの国際会議 VLDB 2020 で受理、発表されました。リサーチペーパーの全文は [こちら](#) からダウンロードできます。

「もし私が顧客に何が欲しいかと聞いていたら、彼らは、『もっと速い馬が欲しい』と答えていただろう」ーフォード・モーター創業者ヘンリー・フォード

この言葉の核心は、「人は、問題に対する本来のソリューションを考えるのではなく、自分の知る範囲でソリューションを思い描く」ということです。データストレージの世界では、長い間この傾向が続いていました。ベンダーは、新しいソリューションを模索するのではなく、データウェアハウスやデータレイクの「古馬」の改善を目指し続けたのです。

今から10年以上前に、クラウドがデータストレージの新境地を開きました。Amazon S3をはじめとするクラウドオブジェクトストアは、極めて大規模かつ費用対効果の高いストレージシステムであり、データウェアハウスやデータレイクを保存するために有用なプラットフォームとなっています。しかし、key-value型ストアの特性により、多くの組織で求められるACIDトランザクションを実現することは困難です。また、オブジェクトのリスト化などのメタデータ操作が高コストであること、一貫性が十分に保証できないことなどが、性能の妨げとなっています。

オブジェクトストアの特性に基づいた3つのアプローチ

データレイク

第一のアプローチはデータレイクです。テーブルをオブジェクトのコレクションとして保存するファイルのディレクトリで、通常は、Apache Parquetのようなカラム形式になっています。テーブルはオブジェクトの集合体で、多くのデータストアやシステムを追加する必要はなく、さまざまなツールからのアクセスが可能です。しかし、性能と一貫性に問題があります。トランザクションの失敗による隠れたデータ破損が発生しやすく、結果整合性により一貫性に欠けるクエリとなり、遅延時間は長く、テーブルのバージョン管理や監査ログなどの基本的な管理機能の利用はできません。

カスタムストレージエンジン

第二のアプローチはカスタムストレージエンジンです。Snowflake データウェアハウスのようなクラウド向けにカスタム構築されたシステムなどを指します。これらのシステムは、単一のデータソースを提供する、一貫性のある独立したサービスでメタデータを管理することで、データレイクの一貫性に欠ける問題を回避できます。しかし、全I/O操作がこのメタデータサービスに接続する必要があるため、クラウドリソースのコストが増大し、性能と可用性が低下するというマイナス面があります。また、Apache Spark、TensorFlow、PyTorchなどの既存のコンピューティングエンジンにコネクタを実装するには多くのエンジニアリング工数が必要になります。これは、データの処理や分析にさまざまなコンピューティングエンジンを使用するデータチームにとっての負担となります。これらのシステムは通常、従来の構造化データに最適化されているため、非構造化データによってエンジニアリングの課題が複雑化するというリスクもあります。さらに、独自のメタデータサービスがユーザーを特定のサービスプロバイダに固定する、すなわち、ユーザーが新たな手法を導入しようとする、そのための「移行作業」に多大なコストと工数を要するという問題があります。



レイクハウス

第三のアプローチはレイクハウスです。私たちは、クラウドオブジェクトストア上のオープンソースの ACID テーブルストレージレイヤーである Delta Lake を利用した斬新な高速化を追求しました。データストアの改善に留まらず、レイクハウスを介したデータの保存・利用法を根本的に変えようと考えたのです。データレイクの限界に挑む新しいシステムとして登場したレイクハウスは、データレイクとデータウェアハウスの優れた要素を取り入れた新しいパラダイムです。新たなシステムデザインによって構築されており、データウェアハウスと類似のデータ構造とデータ管理機能を、データレイクに使用される低コストのストレージに直接実装しています。レイクハウスは、安価で信頼性の高いストレージ（オブジェクトストア形式）が利用可能になった今、最新のニーズに対応するストレージエンジン構築のための最善策と言えるでしょう。

Delta Lake は、Parquet 形式に圧縮されたログ先行書き込みを用いて、どのオブジェクトが Delta テーブルの一部であるかという情報を ACID 方式で保持し、同時にクラウドのオブジェクトストアにも格納します。この設計により、複数オブジェクトの一括更新、オブジェクトの一部の置き換えを、直列化可能な方法で実行できます。オブジェクトからの読み取り・書き込みの高度な並列化は依然として可能です。また、ログは、大規模な表形式のデータセットのメタデータ操作を大幅に高速化します。さらに、Delta Lake は、タイムトラベル（ポイントインタイムのスナップショットのクエリ、更新のロールバック）、自動データレイアウトの最適化、アップサート、キャッシュ、監査ログなどの高度な機能も提供します。これらの機能の相乗効果により、クラウドオブジェクトストアでデータを扱う際の管理性と性能の両方が向上し、最終的には、データウェアハウスとデータレイクの主要なケイパビリティを併せ持つ、よりシンプルで優れたデータアーキテクチャを構築するレイクハウスのパラダイムへの扉が開くことになります。



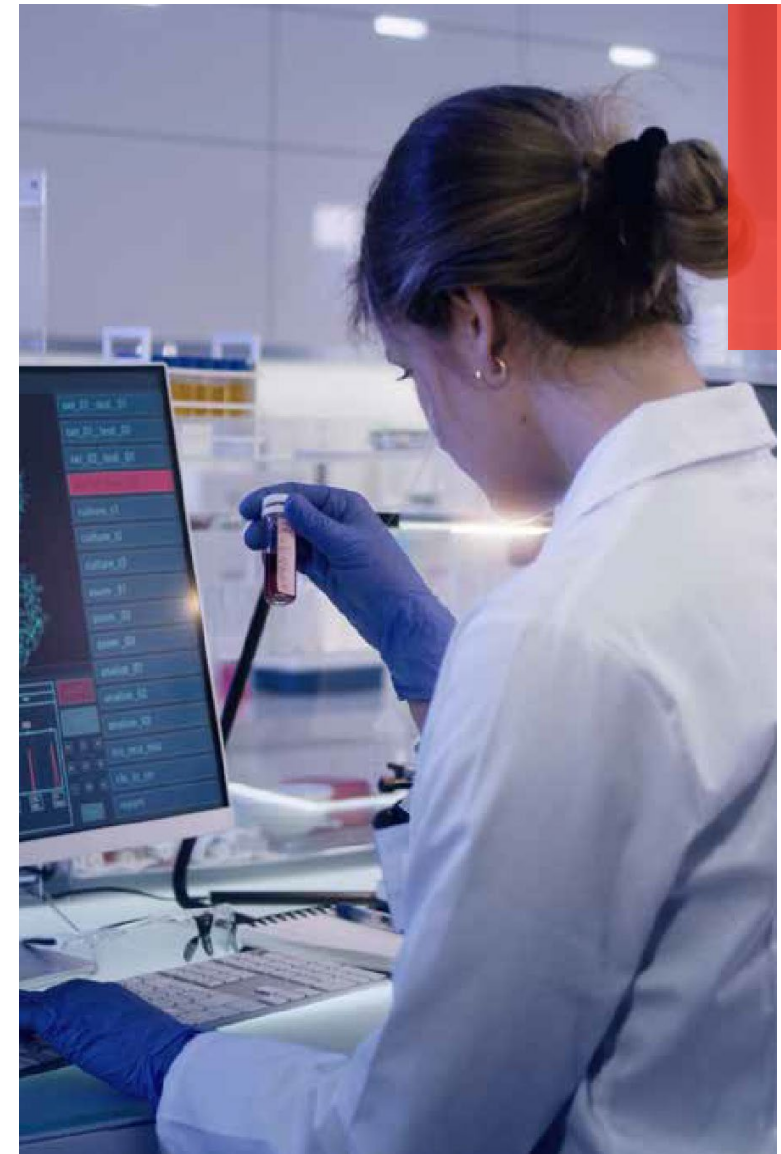
現在、Delta Lake は、数千社におよぶ Databricks のお客様とオープンソースコミュニティの多くの組織に使用されており、さまざまなデータソースやアプリケーションが関わる多様なユースケースにおいて、エクサバイト規模の構造化・非構造化データを日々処理しています。保存されるデータタイプには、エンタープライズ OLTP システムからの変更データキャプチャ（CDC）ログ、アプリケーションログ、時系列データ、グラフ、レポート用の集約テーブル、機械学習用の画像・特徴量データなどがあります。アプリケーションは、最も一般的な SQL ワークロードから、BI、ストリーミング、データサイエンス、機械学習、グラフ分析まで多岐にわたります。Delta Lake は、Parquet や ORC のような構造化ストレージ形式を使用していた多くのデータレイクアプリケーションや、従来型データウェアハウスのワークロードに適していることが実証されています。

これらのユースケースを見ると、お客様は、Delta Lake を活用してワークロードをクラウドオブジェクトストアに対して直接実行することで、データアーキテクチャを極めてシンプルなものにしていることがわかります。また、データレイクとトランザクション機能を備えたレイクハウスを構築して、メッセージキュー（Apache Kafka など）、データレイク、クラウドデータウェアハウス（Snowflake、Amazon Redshift など）による機能の一部または全部を置き換えるというケースもあり、この手法は増加傾向にあるようです。

[リサーチペーパー](#)では次の事柄を中心に解説しています。

- オブジェクトストアの特徴と課題
- Delta Lake の保存形式とアクセスプロトコル
- Delta Lake の現在の機能、メリット、制限
- 現在一般的に導入されているコアなユースケースと特殊なユースケース
- TPC-DS 性能を含む性能実験

このリサーチペーパーは、Delta Lake の理解を深め、低コストのクラウドストレージに保持されたデータに対して Delta Lake が DBMS のような幅広い性能と管理機能をどのようにして実現しているかを理解していただくのに役立ちます。Delta Lake のストレージ形式とアクセスプロトコルによる容易な運用、高可用性、オブジェクトストアへの高帯域幅アクセスについても解説しています。



A dramatic photograph of a space shuttle launching, with a massive plume of orange and white smoke and fire trailing behind it as it ascends into a dark sky. The shuttle is positioned in the upper right quadrant of the frame. In the foreground, the dark silhouette of a launch complex structure is visible.

Chapter

03

Delta Engine とは

03

Delta Engine とは

Delta Engine は、Apache Spark 完全互換のベクトル化クエリエンジンで、最新の CPU アーキテクチャに対応し、Databricks Runtime 7.0 に含まれている Spark 3.0 のクエリオプティマイザおよびキャッシング性能の最適化機能を連携させます。その相乗効果により、[データレイク](#)、特にDelta Lake で実現されたデータレイクでのクエリ性能が大幅に高速化され、[レイクハウスアーキテクチャ](#)の採用やスケーリングが容易になります。

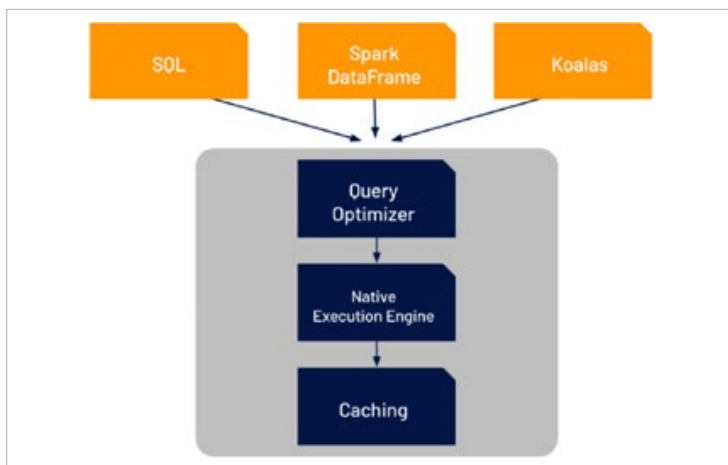
実行性能のスケーリング

ハードウェア業界ではここ数年、CPU クロック速度の向上が頭打ちになっています。その理由についてはここでは触れませんが、重要なのは、CPU のパワーだけに頼らずに、データの高速処理を可能にする新たな方法が必要であるということです。その 1 つに、並列処理できるデータ量の改善が挙げられます。しかし、この並列処理を可能にするには、データ処理エンジンを特別に構築する必要があります。

また、ビジネスが目まぐるしく変化するなか、データのモデリングに十分な時間を割けない状況になっており、そのことがクエリ性能の低下を引き起こしています。これは望ましいことではなく、ビジネスの俊敏性とクエリ性能を同時に最大化する方法が求められています。

高性能クエリを可能にする Delta Engine の登場

Delta Engine は、改良版クエリオプティマイザ、実行レイヤーとクラウドオブジェクトストレージの間に位置するキャッシングレイヤー、C++ で記述されたネイティブのベクトル化実行エンジンの3つのコンポーネントにより、SQL および DataFrame ワークロードでの Delta Lake を高速化します。



改良版クエリオプティマイザは、Spark 3.0の機能（コストベースのオプティマイザ、アダプティブクエリ、動的なランタイムフィルタ）を拡張し、より高度な統計で、スタースキーマのワークロードを最大18倍まで高速化します。

Delta Engineの**キャッシュレイヤー**は、キャッシュする入力データを自動的に選択し、よりCPU効率の高いフォーマットに変換して、NVMe SSDストレージの高速性を活かし、あらゆるワークロードでスキャン性能を最大5倍高速化します。

また、データチームが抱える問題を解決するためのイノベーションとして、Delta Engineは、**ネイティブな実行エンジン Photon**を実装しました。Databricks向けに完全に書き換えられた実行エンジンが、最新のクラウドハードウェアに対応してあらゆるタイプのワークロードの性能を改善し、同時にオープンなSpark APIとの完全な互換性を維持します。

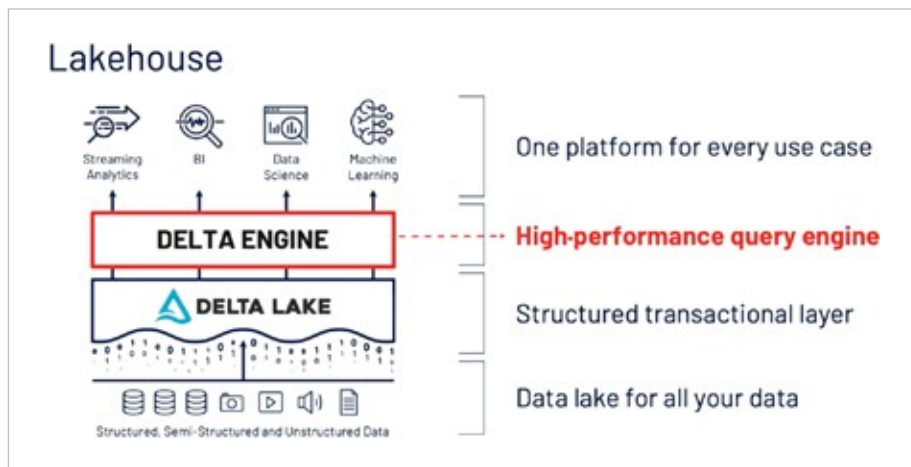


Delta Engine を使ってみる

前述の Delta Engine の3つのコンポーネント（クエリオプティマイザ、キャッシュレイヤー、ネイティブ実行エンジン）のご紹介を通じて、Databricks 内部の複数の箇所における改善が、データレイクでの分析ワークロードの性能をいかに高速化するかをご理解いただけたのではないのでしょうか。

Delta Engine のメリットは、時間・工数・コストの削減に加えて、レイクハウスパターンにおいてデータチームの高度なデータアーキテクチャの設計をサポートし、統合性とシンプルさ向上させることです。ぜひご活用ください。

Delta Engine については、Spark + AI Summit 2020 の基調講演「Delta Engine: High-performance Query Engine for Delta Lake」（Delta Engine: Delta Lake の高性能クエリエンジン）でも詳しく解説しています。録画は[こちら](#)からご覧いただけます。



次のステップ

この eBook では、Delta Lake とその機能が性能を向上させる仕組みについて解説しました。この eBook シリーズの他の特集および、次のステップに役立つ関連資料をご紹介します。

Delta Lake シリーズ

- [Delta Lake シリーズ：基礎と性能](#)
- [Delta Lake シリーズ：機能](#)
- [Delta Lake シリーズ：レイクハウス](#)
- [Delta Lake シリーズ：顧客ユースケース](#)

Delta Lake をさらに詳しく

- [Tableau とレイクハウスアーキテクチャによるデータレイクでの分析](#)
- [Databricks の Web サイト](#)
- [Databricks の無料トライアル](#)

データ + AI Summit Europe 2020

- [Photon を深掘り：ベクトル化の考え方](#)
- [ミートアップ & AMA（何でも聞いてセッション）：MLflow、Delta Lake とレイクハウスのユースケース](#)
- [Delta レイクハウスの性能向上のための一般的な戦略](#)
- [Spark 3.0 でレイクハウスモデルを実現](#)
- [Delta Engine で SQL クエリを飛躍的に高速化](#)

ビデオポッドキャスト・ポッドキャスト

- [レイクハウスへようこそ — Data Brew | 第2話](#)
- [Databricks の Data Brew | シーズン1：レイクハウス](#)
- [データだけでは十分ではない：データアーキテクチャの進化](#)