



Actian Zen

最先端のエッジ データ管理

技術概要

ホワイト ペーパー

目次

はじめに	1
エッジ インテリジェンスの時代	1
Actian Zen と最先端のエッジ データ管理の要件	2
主な設計原則	3
Key-Value ストレージのシンプルさと強み	3
SQL または NoSQL ? エッジでのデータ管理	4
エッジでのデータ管理のニーズ	6
現場サポートの必要性をゼロにする	10
Actian Zen アーキテクチャの詳細	10
MicroKernel データベース エンジン (MKDE)	11
NoSQL と Btrieve	13
SQL とリレーショナル データベース エンジン	14
Zen Control Center (ZenCC)	14
Zen によるデータ管理	15
Zen でサポートされているデータ型	15
Zen のデータ ファイル	16
Zen データの転送と共有	17
Zen データのバックアップと復元	17
Zen のセキュリティ	18
Zen のセキュリティ モデル	18
Zen の暗号化	19
Zen のセキュリティ監査機能	19
Zen の配置モデル	19
アプリケーション専用を組み込む	20
1 つ以上のアプリケーションをサポートするスタンドアロン	20
クライアント/サーバー モードと複数の配置インスタンス	21
Zen によるデータ統合	22
ETL がない Zen 環境	24
アプリケーションと Zen 以外のデータの統合	24
結論	26
Actian について-「Activate your Data™ (お客様のデータを有効活用)」	26

はじめに

10 年前、携帯電話を持っている人は米国では 20% だけでした。今日、この割合は 96% に達しており、中には複数のモバイル デバイスを持っている人もいます。世界全体の携帯電話の数は 2023 年までに 168 億台になると予測されていますが、この数字は IoT (Internet of Things:モノのインターネット) 上のデバイスの急増を考慮に入れていません。今日、平均的な大規模なメーカー、農業関連企業、防衛機関では、平均すると 50,000 台の IoT デバイスになる大規模な IoT センサー グリッドを持っています。このセンサー グリッドは、医療機器システムや高度道路交通システムから機械視覚システムや監視システムまで、あらゆるものをサポートする高速デバイスが豊富なインフラストラクチャの上にあります。それらすべての IoT デバイスを考慮に入れると、2025 年までにデジタル デバイスが 750 億台になるという予測は、合理的であるどころか、おそらく控え目でさえあるように見えます。結局のところ、80 億の人々が各自 1 ~ 2 台のデジタル デバイスを使用すると想定することは合理的だったかもしれませんが、世の中でデータを生成、捕捉、処理、保存、伝達する IoT デバイスの台数の上限を示唆する根本的な制約は何もありません。750 億台という数字は、「ようこそ、未来へ！」と語りかけるデジタル ハイウェイ側の広告看板にすぎないかもしれません。

しかし、この数字よりもはるかに重要なことは、それら急速に進化するデバイスに備わる機能です。IoT は、高速プロセッサ、巨大なメモリ キャッシュ、高速通信機能を備えているため、データ管理上の多大な課題を提起します。これらの課題には、機会を創出するものもあれば、脅威となるものもあります。

携帯電話について考えましょう。米国における電話のうち 5 台に 4 台がスマートフォンであり、音声通話はそれらのデバイスで送受信されるデータの 5% 未満です。実際に毎月、サービス プロバイダーと携帯電話の間では、音声通話のためにやり取りされるデータよりも多くのデータが、ソフトウェア更新の形でやり取りされています。現代アメリカの Z 世代の消費者は、毎月平均で 10GB のデータをやり取りしています。2025 年までには、5G の展開と共に、この数字は毎月 45GB にまで達すると予測されています。

しかも、この数字は先進国の消費者からサンプリングした数字にすぎません。2025 年までには、それまで接続されていなかったものも接続されるようになっていきます。また、有線接続または自社開発の狭帯域幅接続で接続されていたものが、なんらかの形で 4G または WLAN に移行します。これらのデバイスの 28% 以上が、暫定技術を飛び越して 5G に直接移行するものと予測されています。

それらすべてのデータ。それらすべてのアプリケーション。機会も脅威も指の間からこぼれる砂粒のように流れ去っていきます。今日の開発者は、機会や脅威が気付かれぬまま過ぎ去っていかないように、それらの課題を管理できるより良い方法を見つける必要があります。

エッジ インテリジェンスの時代

機会を捕らえ、脅威を撃退する場所は、「エッジ」と呼ばれる薄い領域にあります。そこで、ローカル デバイスとエッジ ゲートウェイ デバイス(大洋横断貨物船から地元のホーム デポ支店に至るまでの環境と考えてください)が、データの集約、クレンジング、メタデータの収集と分析、さらに高度な分析に関連するアクティビティの大部分を実行します。そうなるのには次のような十分な理由があります。

- 集中管理されたデータ ストアへのアクセスに必要な待ち時間は、特にローカルの AI がローカルのデータに基づいて意思決定を行おうとしている場合には非常に長くなる。

- 集中分析のために大量のデータを移動するコストは、処理と分析にローカルのリソースを使用することで軽減できる。
- セキュリティの問題とデータに関するプライバシーの問題により、ローカル データの保管と分析に対する需要が増大している。
- 類似したデバイスのグリッド インストールでは、例外に対するリアルタイムのローカル応答のために、同様のデバイス セット(たとえば PVT)に対してデータを集約および分析する必要がある。

今の段階では SF のように聞こえるかもしれませんが、機械学習が、部品の不良や顔認識のために機械視覚を実行するカメラなどのデバイスでローカルに推論アルゴリズムを実行するのは日常茶飯事になるでしょう。Alexa を複数のインテリジェントなセンサー デバイスと融合させる場合を考えてください。基盤となるハードウェア テクノロジーと経済は、このような融合をサポートするのに十分なクリティカル マスに達しつつあります。

このような現実があるため、台頭しつつある「エッジ インテリジェンスの時代」は、エッジという重要なインターフェイス全体を流れる情報を最も良く管理するにはどうすればよいかという問題を提起します。エッジでインテリジェンスを提供するアルゴリズムを開発することだけが問題ではありません。それらのアルゴリズムのパフォーマンスを改善し、管理するための基礎となるデータ管理戦略を、創造的に再考することが問題なのです。これは、エッジ インテリジェンスを推進するために、堅牢、安全、高速なデータ管理機能をエッジにどう組み込むかという問題でもあります。

本稿では、これらの問題に対する Actian の回答を紹介します。また、最新のエッジ データ管理戦略の要件を概説し、データ管理者が Actian Zen を使用して、製品の全範囲にわたるエッジベースのマシン間の接続および人とマシン間の接続を扱う開発者、製品設計者、サービス管理者の作業をサポートする方法も示します。その他の製品詳細、パフォーマンス ベンチマーク、およびその他の役立つユースケースについては agtech.co.jp/actian/product/ を、開発者向けの資料については agtech.co.jp/products/actian/docs_portal/zendocs/ を参照してください。

Actian Zen と最先端のエッジ データ管理の要件

中核となるデータ管理は、データを格納する必要がある任意の操作と定義することができます。従来のデータ センターやクラウドのデータ センターにあるようなデータベース、データウェアハウス、および Hadoop のようなデータレイクは、複数のプログラムやアプリケーションによって同時に作成、読み取り、更新、および削除される大量で多様なデータを処理できるように進化してきました。

しかし、エッジにおけるデータ管理は、従来のデータ センターやクラウドにおけるデータ管理とはまったく異なります。従来のデータ センターとクラウドのデータ センターは、比較的一貫性があり、適切に制御された環境になっています。これとは対照的に、エッジ環境は極端な異種性を呈しています。さまざまなデバイス、ゲートウェイ、インテリジェントな機器にまたがって開発、配置されるエッジ アプリケーションは、さまざまなプログラミング言語およびスクリプト言語、オペレーティング システム、ハードウェア プラットフォームに依存しています。これは、開発チームにとってはデータ変換、メタデータ管理、セキュリティ上の脆弱性、ベンダー管理および資産管理の領域における難題となります。

これらの難題にもかかわらず、エッジでのデータの処理、分析、およびローカルとクラウドのデータベースへのデータのクエリが増加するでしょう。これらのアクティビティをデータ センターに限定してしまうには、エッジで使用できる電力、メモリ、帯域幅があまりにも多くあります。

これらの理由で、次の7つの重要注意事項では、最先端のエッジ データ管理をサポートするデータベース製品に関する情報を提供します。開発者には以下が必要とされます。

1. 大量の永続的データを管理する能力(特に、散発的に切断の期間が生じるノマドワークの状況において)
2. IoT と従来のイントラネットおよびインターネット環境において、ピアツーピア構成とエッジ/クラウド構成でデータのステートフルおよびステートレス共有を管理する能力
3. 複数のオペレーティング システムで実行できるだけでなく、標準、組み込み型、仮想マシン (VM) の構成で実行できる能力
4. CLI、プログラミング言語、スクリプト言語などさまざまなインターフェイスによる組み込みアクセスのサポート
5. 性質の異なる、低速でデータを収集する多数のグリッドにより、高速のマルチチャネルのデータを収集、集約し、融合する能力
6. 広範なデータ型(整数や文字列データだけでなく、より複雑な JSON、BLOB、従来の構造体などの型)を扱う能力
7. 高度な分析、レポート、可視化のツールおよびプラットフォーム(市販のものとオープン ソースの両方)とのプラグ アンド プレイ互換性

Actian Zen は、最先端のエッジ データ管理の課題に取り組んでいるオンプレミス、クラウド、モバイル、および IoT アプリケーション開発者のニーズに応えるように設計されたデータベース製品の、安全でスケーラブルな単一のポートフォリオで構成されています。Actian Zen は、企業本社、支社/支店、およびリモート現場(モバイル デバイスや IoT など)の各環境に展開されたインテリジェント アプリケーションに永続的なローカル データと分散データを提供します。Actian Zen を使用することで、Microsoft Windows 10、Windows Server 2016、Windows Server 2019、Linux、macOS と iOS、Android、Raspbian Linux、Windows IoT Core、Windows Nano Server を実行する Intel ベースまたは ARM ベースのアーキテクチャ用のデータベース アプリケーションを開発および配置できます。このようなパワー、汎用性、柔軟性の組み合わせにより、ローカル データおよび組み込み分析に対する最新のアプリケーション要件をサポートするために必要な機能を開発者に提供します。

主な設計原則

前述の「最先端のエッジ データ管理」内のリストにある複雑な課題を考慮して、Actian Zen の設計は、シンプルさ、モジュール性、移植性、オープン スタンドールという 4 つの原則によって推進されてきました。これらの原則に基づいて、Zen の設計者が注力してきたターゲットは、高パフォーマンス、広範な開発環境および配置環境のサポート、エンジニアと開発者にとっての採用しやすさ、使いやすさ、現場サポートの必要性の最小化、および旧世代の Actian Zen リリースとの継続的な互換性でした。このセクションでは、これらの原則がどのようにアーキテクチャの特定の機能として現れるかを確認します。

Key-Value ストレージのシンプルさと強み

Zen データベース管理エンジンは、リレーショナル データベース管理システム (RDBMS) アーキテクチャではなく、Key-Value ストア (KVS) アーキテクチャをベースとして構築されていますが、エンタープライズ クラスの RDBMS システムと同様、ACID に完全に対応しています。これは、応答性の高いデータベース管理システム (DBMS) アーキテクチャを必要とする分散環境とインライン分析に焦点を合わせているエッジ インテリジェンスにとって、重要です。KVS アーキテクチャと RDBMS アーキテクチャはどちらも数十年前に考案されましたが、明確に異なる市場空間における顧客の要求と要件に基づく進化により分化したものです。具体的には、RDBMS

システムは大規模な統合システムに重点を置くようになったのに対し、KVS システムは小規模または分散型のパッケージアプリケーションに重点を置くようになりました。KVS ベースのデータベースの大半は NoSQL です。これらは本格的な RDBMS システムではありませんが、一連のテーブルとフィールドを使用してデータ構造をあらかじめ定義しないため、シンプルで高速です。

KVS アーキテクチャは、最先端のエッジ データ管理の視点から見ると、魅力的なパフォーマンス上の利点を備えています。この利点を活用して、幅広いプラットフォームや環境で効果的に運用でき、小規模なものから大規模なものまで調整できるデータベースを設計できます。

次の図 1 は、KVS アーキテクチャの仕組みを簡単に表しています。

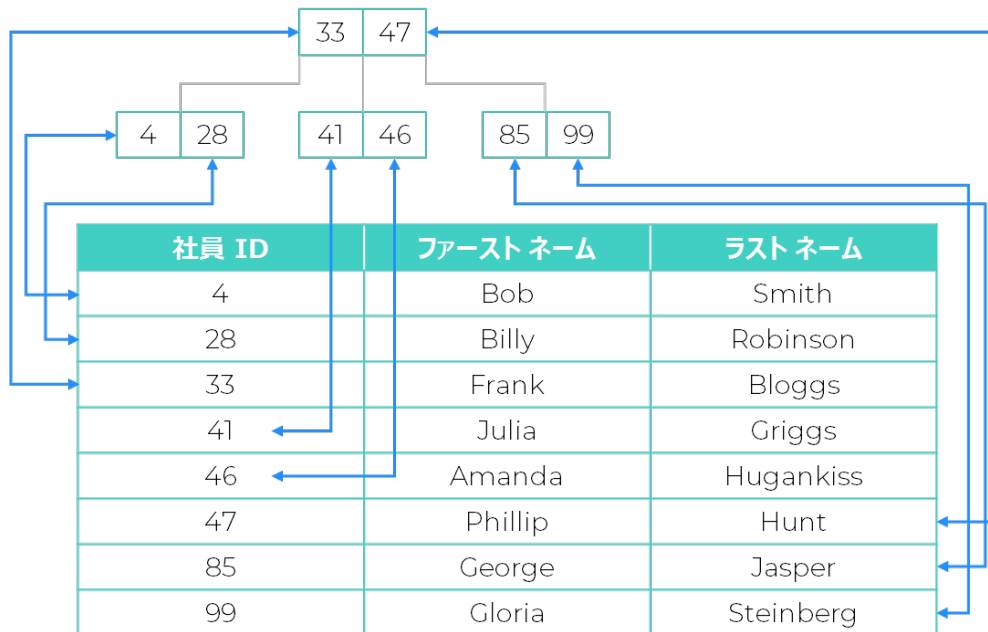


図 1: Key-Value ストアの構造と、それが B ツリー インデックス スキーマと連動する仕組み

KVS テーブルの各データ行には、キーと、そのキーとペアになっている 1 つ以上の値があります。図 1 では、社員 ID の番号がキーであり、それに対応する値がファースト ネーム列とラスト ネーム列にあります。これらの値には、ラージ バイナリ オブジェクト (BLOB) など任意のデータ型を指定できます。クエリに応答して、データベースエンジンのコード ロジックは、キー値を B ツリー インデックスとしてソートし、データテーブルまたはデータベースを構成する一連のペア間の迅速な移動を可能にします。ここでは、青線は、greater-less-than-equal-to アルゴリズムを使用したクエリがどのようにして目的の関連する値に到達するかを示しています。

SQL または NoSQL ? エッジでのデータ管理

エッジでのデータ管理には、読み取りと書き込みに関する速度とバランス、複数の柔軟なデータ型のサポート、複数のモードで実行できる機能 (時系列データ収集、既存のデータ セットに対するその場限りのクエリ、およびパブリッシャー/サブスクリバードモードなど、複雑なクライアント/サーバー モードの操作でトランザクションの読み取りと書き込みを処理できる機能) が必要です。これが、Actian が Zen の基盤となるデータ管理エンジンとして KVS を選択した理由の 1 つです。

[補足情報]すべての NoSQL データベースが同じというわけではありません

すべての NoSQL データベースが KVS であるわけではなく、すべての NoSQL データベースが同じであるわけではありません。図 2 は、NoSQL データベースの最も一般的な 4 つのタイプについて簡単に説明しています。Actian ではこれらの各オプションに基づく製品をご用意していますが、Actian Zen の基盤となるデータ管理エンジンとしては、KVS を選択しました。KVS がエッジでのデータ管理における速度とバランスに関する要件を満たしているためです。

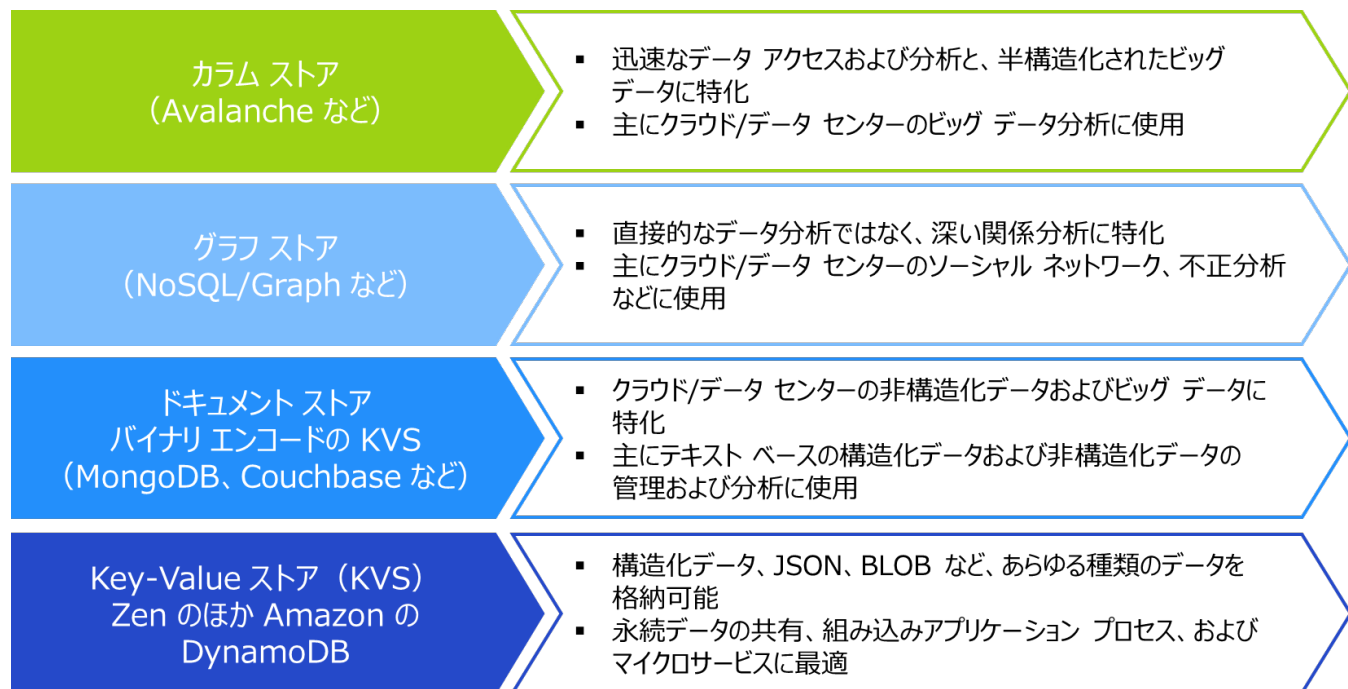


図 2: NoSQL データベースの最も一般的な 4 つのタイプ

他にも理由があります。KVS データベースの多くは NoSQL ですが、KVS は SQL のベースになることができます。Actian Zen は、KVS データ管理エンジンを使用して、SQL アクセスと NoSQL アクセスの両方をサポートします。NoSQL と SQL は、エッジ データ管理のシナリオによって異なる利点があるため、これは重要です。たとえば、データを長時間にわたってストリーミングし、連続的にデータをメモリに書き込む必要がある場合には、NoSQL にいくつかの利点があります。Zen では、NoSQL オプションを使用して自動タイムスタンプを行うことができます。一方、複数のリクエスターから発行されるその場限りのクエリや、保管されているデータの値を変更する可能性があるクエリを管理する場合、SQL には ACID 対応という重要な利点があります(ほとんどの NoSQL データベースにこの利点はありません)。次の図 3 では、データのアクセスと管理の手段として、NoSQL を使用する場合と SQL を使用する場合の主な利点と欠点を比較しています。

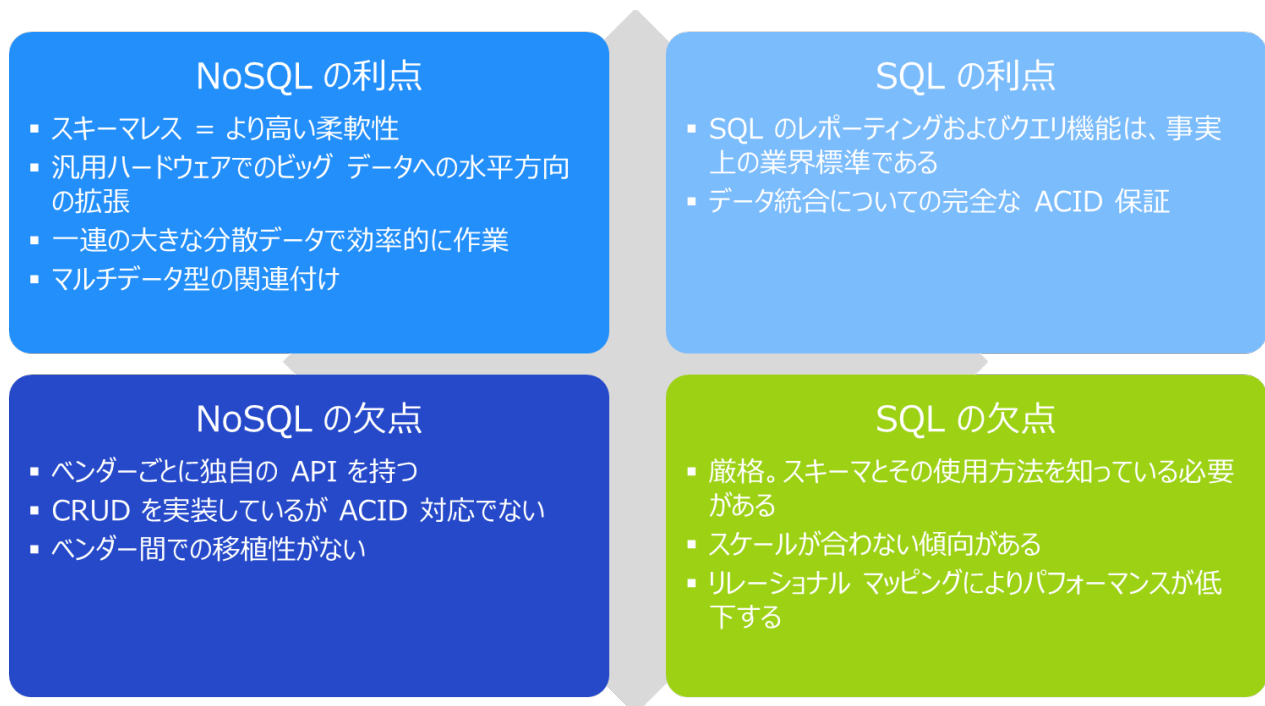


図 3: NoSQL と SQL の相対的な利点と欠点

エッジでのデータ管理のニーズ

エッジでは、データを一時的に格納する使用事例ははるかに多様です。データ速度の低い IoT センサーによってプッシュされる時系列データから、POS デバイスによってプッシュされる大量のトランザクション データや、携帯電話の Web UI からプッシュされるフォームベースのデータまで、その用途は多岐にわたります。この多様性により、特殊なファイル システムから、IndexedDB や SQLite などのデータをキャッシュするための低機能廉価版のデータベースまで、さまざまなデータ管理システムが作成されています。

これとは正反対の位置にあるのは、もちろん、複数のプログラムやアプリケーションによって同時操作で作成、読み取り、更新、および削除される大量で多様なデータを処理できるデータ管理システムです。データベース、データウェアハウス、および Hadoop のようなデータレイクは、クラウド データ センターと従来のデータ センターの両方で、この分野に特化して進化してきました。

これらのエッジ環境、クラウド環境、および従来のデータ センター環境の組み合わせを考えると、データベースやその他のデータ管理システムの数は数百にも及びます。それらデータ管理システムのいずれか、またはシステム群は、小規模システム、大規模な分散環境、スキーマ付きの構造化データ、またはドキュメント付きの JSON データのいずれかで構築されていようと、データ管理の特定のニーズを満たすことができます。ただし、急所は、これらすべての異なるデータ管理のユースケースに適用できる万能のソリューションが存在しないということです。

しかし、最先端のエッジ データ管理では、データがエッジからクラウドに、クラウドからデータ センターに移動し、そして再びエッジに戻るときに、データの処理と分析の要件に対応することがますます求められるようになってきています。当然のことながら、データ管理の要件はエッジ デバイスごとに異なります。主なエッジ デバイスにおけるデータの処理と分析に関する役割の詳細については、補足情報を参照してください。

[補足情報] エッジでのデータの処理と分析に関する要件

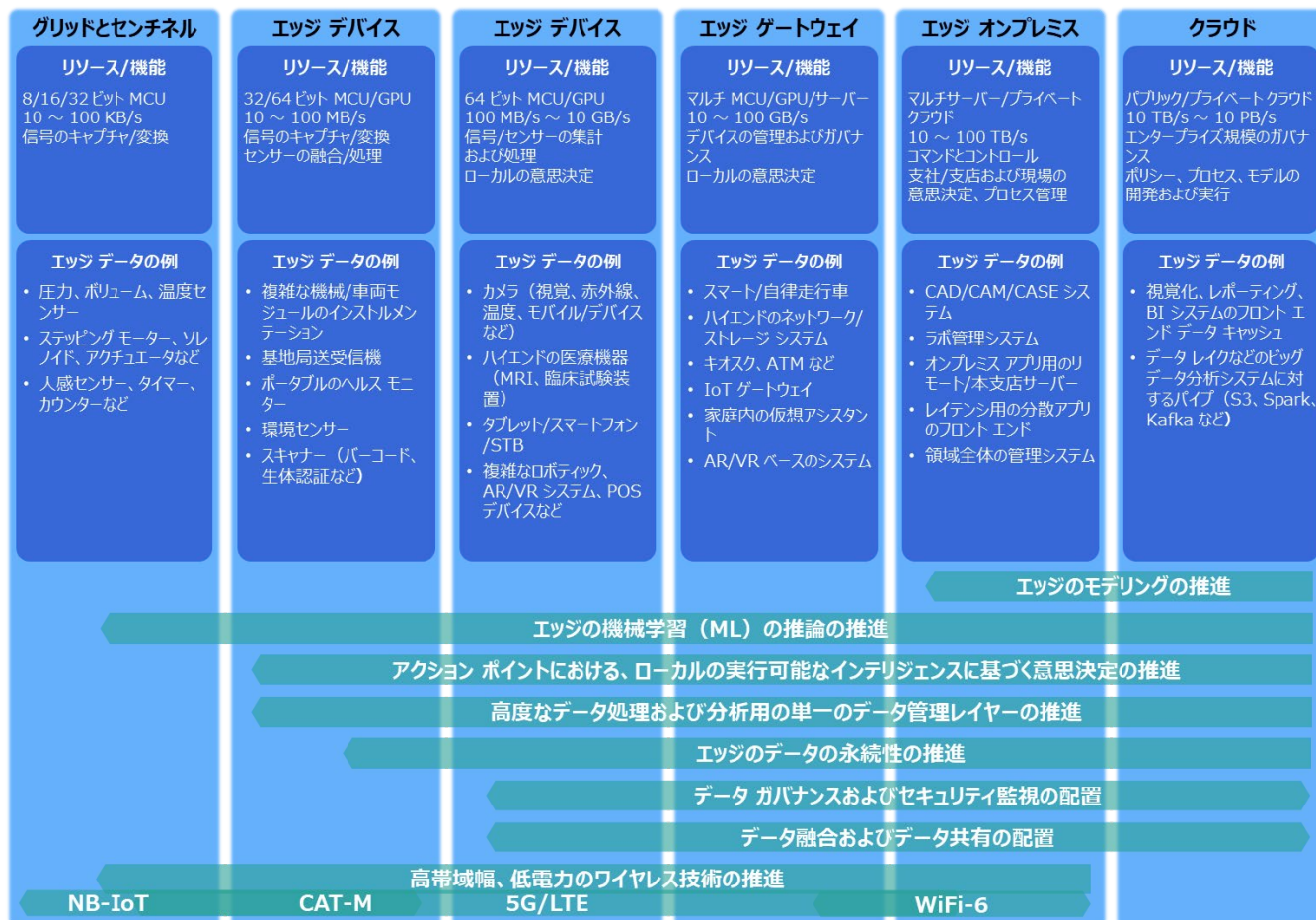


図 4: エッジ データ管理が必要な場所と、それが必要な理由と場所に対するニーズの変化

左から右に説明:グリッドとセンチネルはネットワーク エッジにあり、より洗練された、リソースの豊富なさまざまなデバイスのフロント エンド処理を行います。次いで、すべてのデバイスが並列で入力内容をエッジ ゲートウェイに転送します。さらに、エッジ ゲートウェイが自身よりも集中管理度の高いオンプレミスのエッジ システムにその入力内容を転送します。オンプレミスのエッジシステムは、成長するクラウド環境と停滞したデータ センター環境の中にあるものの一部を、待ち時間が少なく(低レイテンシ)、十分に活用されていない一連のエッジ リソースに移行する、新しいデータ センターのユース ケースと機会を表しています。

デバイスの説明の下のバーは、より堅牢なデータ管理のために要件を押し上げている主な傾向のいくつかを示しています。主要な推進要因としては、センサー フュージョンなどの高度な分析への移行や、エッジ デバイスまたはエッジ ゲートウェイでのアクション ポイントにおける意思決定などがあります。

これらさまざまなデバイスとユース ケースによって表されるデータ管理のニーズは多種多様です。エッジ分析アプリケーションの領域には、広範なユース ケースとデバイスの種類があります。ハイ エンドでは、64 ビット CPU、GPU アレイ、高解像度のビデオおよびオーディオ リソースを備えたデバイスで、機械学習 (ML) システムが機械視覚のユース ケースをサポートしています。リアルタイム アプリケーション用の ML では、低レイテンシが求められるだけでなく、多くの場合、少なくとも信号対雑音比の低減のための前処理が要求されます。ロー エンドでは、わずか数十キロバイトのメモリを搭載する 32 ビット マイクロコントローラーで、ML システムが教師なしルーチンを活用して低電力で動作していることがわかります。ここで、ジェスチャー、PVT センサーの変更、または他の何ら

かの明確なイベントにより、ML を実行する監視(センチネル)機能が駆動され、一連のアクションがトリガーされます。これらのアクションには、定義されたタスクを実行できる、よりインテリジェントでリソース集約的なシステムモジュールの起動が通常含まれます。

家庭内の仮想アシスタントや IoT ゲートウェイなど、既にユーザーになじみのある多くのインテリジェント エッジデバイスは、複数のユース ケースを組み合わせたプラットフォームを表しています。これらのデバイスは、ローカル エッジ サーバーでデータを集約し、分散アプリケーションを実行できます。ローカル エッジ サーバーは、クローゼットや飛行機の胴体の中で実行できるほどの小型のものから、(オンプレミスまたはプライベート クラウドの)複数のサーバー ラックを占有するほどの大型のものまであります。

モバイル ワークフォース アプリケーションには、他にもエッジ データ管理の課題があり、特にネットワークの切断(計画的か計画外かにかかわらず、地方の配送トラックで経験されるような切断)がある場合には、その課題があります。モバイル デバイスは、接続時にはデータを共有および配信できる必要がありますが、接続が切断されている場合にはローカル データのみを操作する必要があります。

モバイル シナリオは、分散アプリケーションをサポートするすべての関連プラットフォームにまたがる単一のデータ管理プラットフォームで最適に機能します。また、単一のデータ プラットフォームのメリットを享受できる別のシナリオは、製品ラインが多くの同一のコア コンポーネントや機能サービスに依存しているが、顧客が選択したサービスやデバイスの階層に応じて、機能が異なる方法で提示されるシナリオです。一連の医療用画像機器は、ポータブルな CT スキャナーから、固定型でハイ エンドの実験室用スキャナーまでの全範囲に及びます。これらすべての製品で単一のデータ管理プラットフォームを使用するには、そのデータ管理システムが、ロー エンドでは Android または組み込み Linux を実行しているスタンドアロン 64 ビット MCU で実行される一方、ハイ エンドでは組み込みのマルチコアの Intel 64 ビット CPU サーバーや Windows Server 2019(ローカルでの画像保管用に 2 TB 以上のメモリを搭載)で実行できる柔軟性が必要です。これらのスケーラビリティ上の留意点は、インテリジェントな農機具(APV から刈り取り機まで)からインテリジェントな機内エンターテイメント センター(ファーストクラス、ビジネス クラス、エコノミー クラスなどのクラスによって異なるオプションを提供する)まで、すべてに見られます。

これらのニーズに対応する製品とサービスは、従来のソフトウェア ベンダーやシステム インテグレーターの多くが作成しています。彼らがこの課題に取り組む方法は、プラットフォームごとに最適化された異なる製品とサービスを統合することです。最初に取り掛かるのはプラットフォームです。その後で、統合を行います。複数のプラットフォームにわたってデータ管理を容易にするために、API、ファイル システム、および関数呼び出しの統一されたセットを製品とサービスに搭載するために多くの作業が行われます。

Actian Zen を使用することで、開発者はこれまでとは異なるアプローチを取ることができます。Actian Zen を使用すると、開発者はシンプルでありながら強力なマイクロカーネル コアを最小の共通要素として利用し、そのコアを拡張して、エッジの幅広い要件に対応するための機能を追加することができます。図 5 は、[補足情報] エッジでのデータの処理と分析に関する要件に表示されている機能領域とユース ケースに対して Actian Zen の製品ポートフォリオをマッピングしたものです。



図 5: Actian Zen 製品ファミリー

Actian Zen Core エディション: スタンドアロンまたは断続的に接続される通信モードで動作して永続的でステータフルなデータ処理/分析を必要とするローカル アプリケーションを持つデバイスとインテリジェント マシンを強化できます。そのようなデバイスの例には、スマートフォン、タブレット、および広範な IoT デバイスがあります。開発者は、プログラムによるアクセスまたはスクリプトによるアクセスのために、Zen Core をアプリケーションに直接組み込むことができます。

Zen Edge エディション: より複雑な分散アプリケーションとさまざまな標準インターフェイス (ADO.NET、ODBC、JDBC など) をサポートするように、Zen Core エディションの機能を拡張したものです。Linux 環境およびその他の IoT 環境のパブリッシュ/サブスクライブ モデルで複数のデータ ストリームを同時に管理できます。開発者は、クライアント アプリケーションとは別に Zen Edge をインストールして実行でき、ユース ケースのニーズに応じて、ローカルまたはリモートの異なるデバイスで実行できます。Zen Edge は、独立および共有のデータ サービス上で、1 つまたは複数のアプリケーションへの同時ユーザー アクセスをサポートします。

Zen Enterprise Server エディション: 数千人のエンドユーザーまたはアプリケーションというニーズをサポートできるように Zen Edge の機能を拡張、スケールアップしたものです。より複雑な動作環境 (オンプレミス環境および仮想環境を含む) をサポートできます。Zen Enterprise Server エディションは、Zen Enterprise インスタンス上で

ローカルに動作するデータベース管理機能を追加できると同時に、ダウンストリームの Zen Core/Edge インスタンスの管理プラットフォームとしても機能します。

Zen Cloud Server エディション: Zen Enterprise Server エディションのすべての機能を提供しますが、パブリックまたはプライベートクラウド環境の仮想マシンおよびコンテナでも実行されます。

Actian Zen 製品は、Intel および ARM の 64 ビットアーキテクチャをベースとするメジャーなサーバー、デスクトップ、モバイル、および仮想環境のほとんどに簡単に導入できます。どのエディションも、NoSQL API および SQL クエリ、同じデータベース構造、管理および制御サービスに基づいて構築された共通のアーキテクチャをベースにしています。すべてのエディションは共通のファイル形式で書き込みを行います。また、組み込み方式や不定期に接続される方式で動作している場合でも、継続的な監視なしでリモートから実行するように配置できます。一般に入手可能な他の市販データベース製品は、Linux が組み込まれた ARM ベースのモバイル プラットフォーム、AWS および Microsoft Azure パブリッククラウド、VMware または Microsoft Hyper-V で構築されたプライベートクラウド、ローカルのサーバーとワークステーションでは動作しません。このような配置の柔軟性により、開発者は必要なデータ管理機能への共通のインターフェイスを使用して、アプリケーションをあるプラットフォームから別のプラットフォームに移植したり、開発者間でデータを安全に共有したり、オンプレミス サーバーに戻したり、クラウドにアップロードしたりすることが可能になります。

現場サポートの必要性をゼロにする

すべての Zen データベース製品には、データセンターや Web 開発者にとっては当たり前とされる機能が備わっていますが、中には、エッジ環境では必ずしも利用できない機能や、開発者が一から再発明することが多い機能も含まれています。必要な機能がデータセンターとクラウドデータベースとは関係のない方法で実装される場合もあります。たとえば、エッジにおけるコア要件はデータベース管理を回避することです。データベース管理は、従来の環境で非常に広範で重要なニーズであるため、複雑な組織データベースの管理をサポートするための専門職や大学のカリキュラムが存在するほどにもなっています。最先端のエッジ データ管理のニーズは、このようなデータ管理とは異なります。エッジのデータベースは、データベース管理者 (DBA) の介入を最小限に抑えて動作する必要があり、配置の開始、停止、構成、カタログ作成、監視などのために手作業を必要とする介入をなくす必要があります。

Actian Zen は、リモートからの構成、監視、診断に使用できる API である Distributed Tuning Interface (DTI) を使用して、データベース管理者不要のデータベースを配置できるようにします。このデータベース管理者不要のアプローチにより、テクニカル サポート担当者が現場で Actian Zen データベースを物理的に操作する必要がなくなります。

Actian Zen アーキテクチャの詳細

前述のように、すべての Actian Zen エディションは、単一のスケラブルで安全なモジュラー アーキテクチャをベースにして構築されています。その概要を図 6 に示します。

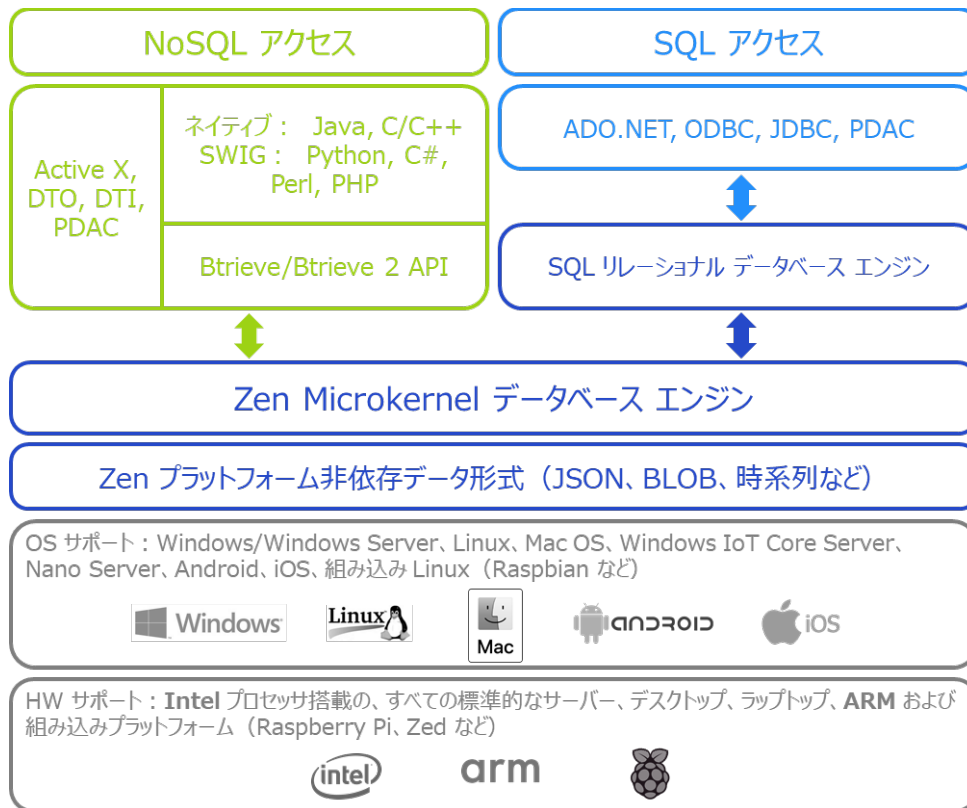


図 6: Actian Zen アーキテクチャ

実際の実装は、基盤となる物理リソースまたは仮想リソースを最大限に活用するため、レベルによって異なります。しかし、データベース自体は、すべてのアプリケーションが共通のアーキテクチャを使用できるようにすると共に、すべての開発者とエンドユーザーが共通の API および SQL に準拠できるようにします。以下のサブセクションでは、Actian Zen アーキテクチャの主なコンポーネントについて詳しく説明します。

MicroKernel データベース エンジン (MKDE)

Zen MicroKernel データベース エンジン (MKDE) は、すべてのデータ アクセス方法を含む、データベースの主要オペレーションを実行します。MKDE の主要オペレーションを使用すると、開発アプリケーションではキー値あるいはシーケンシャルまたはランダム アクセス方法に基づいて、レコードの取得、挿入、更新、または削除が行えるようになります。

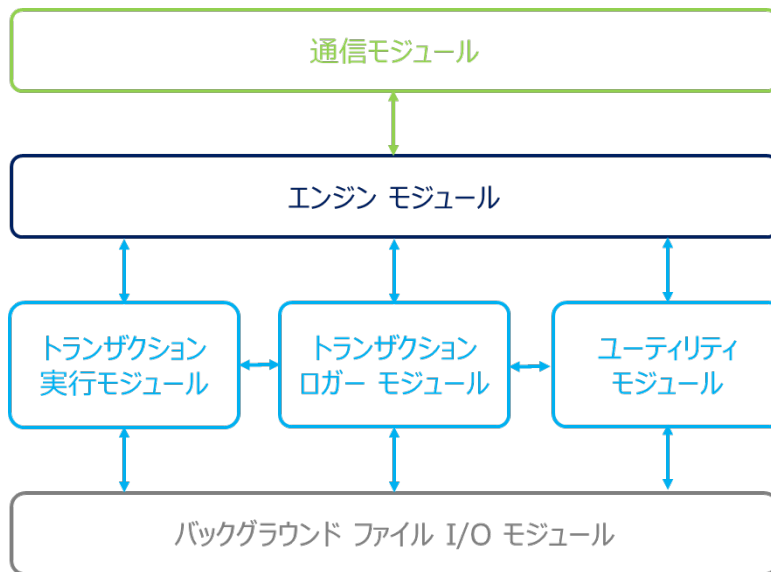


図 7: MicroKernel データベース エンジン アーキテクチャ

前述のように、MKDE に收容されるデータベースは共通のファイル形式で格納され、現在は 64 TB ものデータを保持することができます。これらの MicroKernel ファイル (Zen ファイル) には、レコードのほか、高速の CRUD 操作を可能にする B ツリー スタイルのインデックスが格納されています。これらには、スキーマレスな環境で、API 直接呼び出しを使用するか、高レベルの JSON または BLOB ファイルを使用して、簡単にアクセスできます。スキーマを使用する場合には、スキーマは独立したデータ辞書ファイル (DDF) にあります。現在のファイル形式は 64 TB が上限で、1 つのテーブルに数十億件までのレコードを格納できます。

MKDE は、複数の Zen データファイル (ローカル システムやリモートシステムに保管されているデータファイルなど) を簡単に操作できます。MKDE の機能には、ANSI SQL コマンド (コードに組み込むか、CLI を使用する) または NoSQL API (ほとんどすべてのプログラミング言語またはスクリプト言語を使用してアクセス可能) を使用してアクセスできます。図 8 は、NoSQL および SQL 呼び出しが Actian Zen 内でどのように実行されるかを示しています。

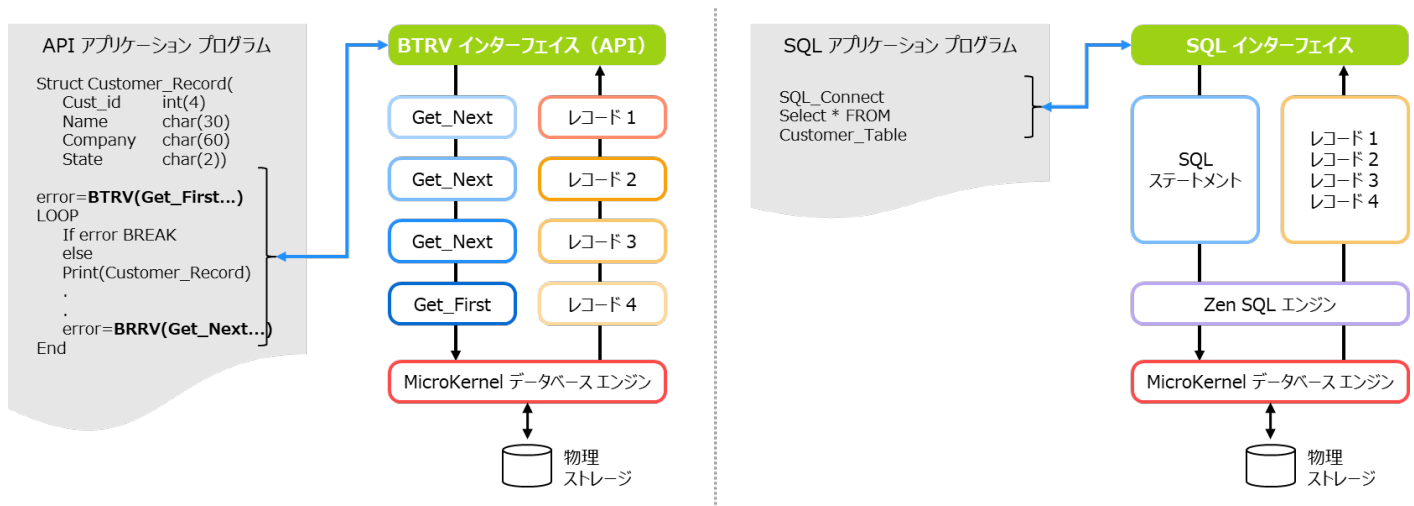


図 8: Actian Zen の NoSQL と SQL のアクセス方法の比較

最大限の柔軟性、シンプルさ、およびパフォーマンスを実現するために、Actian では、トランザクション処理がデータベースに直接適用される場合には Btrieve オプションを使用することをお勧めします。ただし、これは、

データをリレーショナル データベースとして使用したり、DDF として格納されている使用可能なスキーマを活用できる SQL 操作で使用したりすることを妨げるものではありません。

[補足情報] Actian Zen ソフトウェア開発キット (SDK)

Actian Zen ソフトウェア開発環境は、いくつかのソフトウェア開発キット (SDK) で構成されています。これらの SDK により、開発者はトランザクショナル アクセス方法 (NoSQL または Btrieve) の API、リレーショナル アクセス方法 (SQL) の API、またはこれら 2 つを組み合わせ使用して、アプリケーションを作成できます。SQL と NoSQL (Btrieve) のどちらの方法も、C、C++、C# や Java などほとんどすべてのプログラミング言語から使用できるほか、Python、Perl、PHP、JavaScript などの一般的なほとんどのスクリプト言語からも使用できます。

Zen の SDK は、Distributed Tuning Interface (DTI) 関数へのアクセスを提供します。この関数により、開発者とデータ エンジニアは、従来のデータセンターの環境では通常 DBA に委ねられる機能を処理できるようになります。DTI とそのオブジェクト指向パートナーである Distributed Tuning Objects (DTO) により、開発者は広範な、カスタマイズした便利なサーバー管理ツールとインターフェイスを迅速かつ簡単に開発できるようになります。たとえば、Windows 環境では、開発者は OLE オートメーション コントローラーを使用することも、あるいは C/C++、Visual Basic、Delphi などのプログラミング言語を使用して COM クライアントを作成することもできます。

Zen の SDK を使用したい開発者が作業を開始するには、agtech.co.jp/products/actian/docs_portal/zendocs/ で入手可能なドキュメント セットをご活用ください。API に関する詳細なドキュメントは agtech.co.jp/products/actian/docs_portal/ から入手できます。

NoSQL と Btrieve

Btrieve は、NoSQL の API であり、MKDE と直接通信します。ご利用いただける Btrieve API は 2 つあります。

- オリジナルの Btrieve API。単一の、パラメーター化された関数呼び出しによるデータベース アクセスを可能にします。
- Btrieve 2 API。新しい開発者が覚えやすい名前が付いた関数によるオブジェクト指向のデータベース アクセスを可能にします。

Btrieve API は以下との互換性を提供します。

- C/C++/C#、Delphi、COBOL、Visual Basic、Java プログラミング言語。
- JavaScript、Python、Perl、PHP スクリプト言語。
- いくつかの一般的な開発環境と SDK。JDK (Java Development Kit) 1.4 以降、Microsoft Visual 環境、Micro Focus、Android、Apple、Embarcadero など。

Zen SDK for Btrieve には、各インターフェイスを使用するために必要なすべてのヘッダー ファイル、オブジェクト指向の新しいクラスとメソッドのドキュメント、C と C++ 用の 32 ビットと 64 ビットの新しいライブラリが含まれています。また、SDK には、Perl、PHP、C# および Python 用の SWIG (Simplified Wrapper and Interface Generator) ファイルも同梱されているため、これらの言語の知識がある開発者は、これらのファイルを使い、C ライブラリおよび C++ ライブラリを介して Btrieve にアクセスすることができます。

Btrieve SDK には、JDBC に類似した一連の Java クラス ライブラリ (JCL) も含まれています。これらのライブラリを使用すると、オブジェクト指向のアプリケーションで Btrieve API セットのすべての機能を活用できます。また、MKDE と直接対話するので、リレーショナル データベース エンジンを通じて MKDE にアクセスする SQL のアクセス方法を使用する場合よりもパフォーマンスを改善できます。

SQL とリレーショナル データベース エンジン

Actian Zen は、データの定義、操作、および制御のために広範な SQL ステートメントと関連タスクをサポートしています。これらのステートメントとタスクは、ANSI と IBM SAA (Systems Application Architecture) の両方の標準に完全に準拠しています。

- **データ定義**ステートメントを使用すると、テーブルだけでなくビュー、インデックス、トリガー、ストアド プロシージャ、ユーザー定義関数を作成、削除、または変更できます。これらの SQL ステートメントにより、Zen リレーショナル エンジンは DDF 内のデータベース メタデータを更新します。また、基盤となっている MicroKernel ファイル構造も作成および操作できます。
- **データ操作**ステートメントを使用すると、標準のリレーショナル データ セットの理論により、データベースの内容にアクセスして変更を加えることができます。Zen では、テーブル データの取得と変更、トランザクションの定義、ビューと一時ファイルの作成、ストアド プロシージャとトリガーの実行を行うために、標準 SQL データ操作ステートメントを実行できます。
- **データ制御**ステートメントを使用すると、データベースのセキュリティ パラメーターを定義できます。辞書を作成する場合、その辞書のセキュリティを明示的に有効にするまで、その辞書のセキュリティは定義されません。Zen では、セキュリティの有効化と無効化、ユーザーとグループの作成と削除、グループにおけるユーザーの追加と削除、ユーザー パスワードの変更、テーブルと列への権限の付与と取り消しを行うことができます。

Zen SQL Relational Database エンジン (SRDE) は、すべてのデータベース メタデータを一連の DDF に格納し、環境で定義されている各データベースへの接続を制御します。同時に、Actian Zen の柔軟性により、これらのデータベースを NoSQL ストアのように Btrieve を併用してアクセスできます。

Zen Control Center (ZenCC)

Actian Zen Control Center (ZenCC) は、Zen Enterprise Server および Zen Cloud Server に統合されているユーティリティです。ユーザー (または、配置用にテストと QA を行っている開発者) はこのユーティリティから Zen Core、Zen Edge、Zen Enterprise Server および Zen Cloud エンジンへの接続、データベースとテーブルの作成と変更、データのクエリと更新、エンジンのパフォーマンスの調整、Zen ドキュメント ライブラリへのアクセスを行うことができます。

ZenCC は、ファイル エクスプローラーのモチーフを使用したグラフィカル ツールであり、Zen エクスプローラーという名前がふさわしいツールです。ZenCC を使用すると、Windows、Linux、または MacOS で実行されている任意の Zen Enterprise または Zen Cloud インスタンスから、Zen サーバーおよびクライアントを包括的に構成、管理できるようになります。図 9 は、ZenCC が提供するさまざまな機能を示しています。

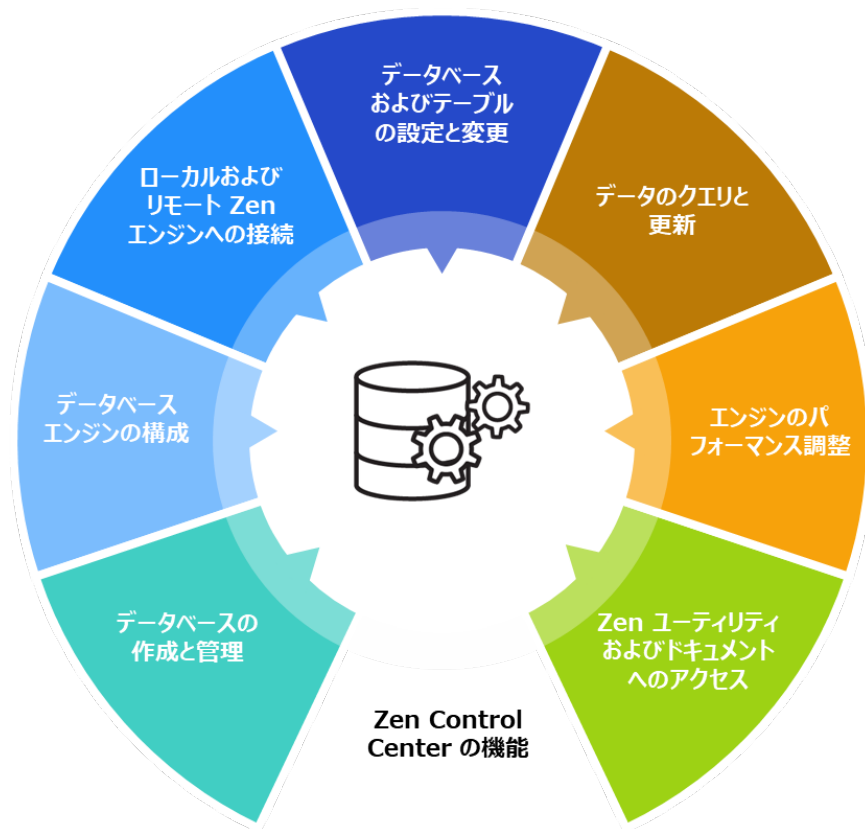


図 9: Zen Control Center(ZenCC)の機能

ZenCC を使用すると、ローカル クライアントや、ローカルまたはリモートの登録済みエンジンの、すべてのプロパティを設定できます。このエンジンには、ZenCC をローカルでサポートするにはリソースが足りない環境 (Windows Nano Server、Windows IoT Core、Raspbian など) で実行されているエンジンも含まれます。

また、ZenCC には、複数の SQL ステートメントを編集して実行できる SQL Editor も組み込まれています。SQL Editor に含まれているグラフィック指向の Table Editor を使用すると、テーブルの列、インデックス、および参照整合性 (RI) 制約の特性を追加、削除、または変更できます。

さらに、ZenCC UI は、SQL ステートメントの実行結果のテキストビュー、SQL ステートメントのアウトラインビュー、スプレッドシートに似たグリッド ビューを提供します。グリッド ビューでは、各フィールドがセル列として表示され、フィールドのデータを直接編集できます。

Zen によるデータ管理

Zen でサポートされているデータ型

Action Zen は、最先端のエッジ データ管理のシナリオで頻出するすべてのデータ型をネイティブでサポートしています。サポートされるデータ型は次のとおりです。

- 1 ~ 8 バイトの符号付きおよび符号なし整数

- 自動インクリメントする 2 ～ 8 バイトの ID 型
- 4 バイトおよび 8 バイトの浮動小数点値
- スルで終わるか、空白が埋め込まれた、シングルバイト文字列およびダブルバイト文字列
- 複数の日時形式
- 7 つの異なる COBOL 数値データ型
- BLOB/CLOB、GUID、LOGICAL などの特殊なデータ型

さらに、Zen は、JSON および BLOB データを含む、ほとんどすべてのユーザー定義、半構造化、または非構造化データを取り扱います。JSON ファイルの場合、Zen は、JSON のフィルターとクエリ演算子を使用して JSON ファイル内のさまざまなオブジェクトと構造を解析することで、高速アクセス機能とデータを考慮した検索機能を提供します。

時系列サポート(例: 統計的工程管理検査、科学的調査、その他の高速分析のケース)を必要とする IoT 環境の場合、Zen は SQL または Btrieve API を使用して入力されたデータに、セブタ秒またはナノ秒単位でのタイムスタンプを付加できます。これら高分解能のタイムスタンプでは、UTC または現地時間のいずれかを基準にすることができます。

最後に、Zen は自動インクリメント データ型をサポートしています。このデータ型は、ファイルへのデータレコードの追加に対して自動的にインクリメントした値を格納するデータ型であり、この値は各 Zen データファイルにおける各レコードの一意の値/ID として使用できます。これを後処理用の SQL クエリまたは Btrieve リクエストの基準として、またはデータの集約、消去、またはその他のクリーンアップ操作を呼び出すトリガーとして使用することで、手動によるデータベース管理の必要性をさらに減らすことができます。

Zen のデータ ファイル

前述のように、Zen ファミリのすべてのメンバーは同じ Zen データ ファイル形式を使用しています。プラットフォームやエンジンのタイプを問わない単一のファイル形式がサポートされているので、複雑な変換や ETL (抽出、変換、格納) 処理を必要とせず、データベース ファイルを Zen プラットフォーム間で移行できます。

各 Zen ファイルには単一のデータベース テーブルが格納されており、個別に管理できます。Actian Zen バージョン 13/14 では、ファイル(つまりテーブル)の最大サイズは 64 TB です。小規模な環境で貴重な記憶域を節約するために、Zen はレコードとページという 2 種類のデータの圧縮を提供しています。これらのデータ圧縮は、別々に使うこともできれば、同時に使うこともできます。圧縮により、データ ファイルのサイズが削減されるだけでなく、データ型に応じたパフォーマンスの改善が見られ、異なるプラットフォーム上の Zen インスタンス間でデータを移動する際のパフォーマンスへの影響を相殺できます。圧縮率は 1/8 にもなることがありますが、ファイル圧縮全体は圧縮されるデータの型に大きく依存します。

Zen のすべてのデータ ファイルとデータ辞書はオペレーティング システムで表示できるため、簡単に配置およびバックアップできます。また、Actian Zen エンジンは、最新(v13)のファイル形式から、1992年に最初にリリースされた v6 ファイル形式までの間の読み取り/書き込みの互換性を提供します。継続的な下位互換性は Actian Zen の特徴であり、今日構築したアプリケーションで何年先もデータにアクセスできることを期待できます。

Zen データの転送と共有

Actian Zen のすべてのエディション (Zen Core から Zen Cloud Server まで) は、ネットワーク越しのデータの転送と共有をサポートしています。Zen アプリケーションが実行されているマシンからネットワーク ファイルにアクセスして管理するには、そのマシンで適切なクライアント リクエスターを使用します。アプリケーションで行った Zen 呼び出しは、Zen クライアントを経由して、Actian Zen データベース エンジン (ローカルまたはネットワーク越しのリモート) に送信されます。Actian Zen エンジンはクライアントに応答を返送し、クライアントはアプリケーションに応答を渡します。接続に使用される TCP/IP プロトコル (v4 または v6) は、すべての動作環境でサポートされており、中断が発生した場合に特定の環境で自動的に再接続を行うことができます。ワイヤ暗号化はすべてのリモート通信で利用できるため、パブリック ネットワーク セグメントを横断する可能性のある重要データや個人データを保護するために必要なセキュリティが確保されます。

Zen データのバックアップと復元

エッジでもデータの消失は発生する可能性があるため、バックアップ戦略が、最先端のエッジ データ管理でも重要です。前述のように、Actian Zen ファイルはすべての OS から見えるため、既存のバックアップ戦略に簡単に組み込むことができます。標準オペレーティング システム ユーティリティ、サードパーティ製ソフトウェア、またはシンプルなスクリプト ソリューション (ユーザーがデータにアクセスしていない場合) でも、Zen ファイルの完全バックアップまたは増分バックアップと回復を確実に行うことができます。

ユーザーまたは自動プロセスが使用中のデータベース ファイルをバックアップするには、次の 3 つのソリューションから選択できます。

- **Continuous オペレーション モード** は、MKDE 内の特別なファイル処理メカニズムです。ユーザー/アプリケーションを中断せずに、データ ファイルに対するすべての変更を一時データ ファイル (デルタ ファイルと呼ばれる) に保存してデータ ファイル (開いている場合) を閉じるようエンジンに指示します。データ ファイルは実質的にある時点で凍結されているため、バックアップは通常の方法でデータ ファイルにアクセスすることができます。バックアップが完了したら、Continuous オペレーション モードをオフにして、デルタ ファイルに保存されているすべての変更を元のデータ ファイルに戻すようエンジンに要求します。この要求は MKDE の内部で処理されるため、このオプションはすべてのプラットフォームとエンジンのタイプで利用できるほか、開発者が API またはスクリプトを介して開始できます。
- **Zen Backup Agent** は、バックアップ ソフトウェアからの前処理および後処理のコマンドライン呼び出しで起動することができます。Backup Agent は、オンにすると、現在開いているすべてのファイルに対して Continuous オペレーション モードを自動的に呼び出し、その後、Backup Agent のセッション中に新しいファイルが開かれないかどうかを監視します。Backup Agent が完了したら、バックアップ ソフトウェアから Backup Agent を再度呼び出して、すべてのファイルを通常の方法に戻します。Backup Agent は Zen Enterprise Server と Zen Cloud Server に同梱されています。¹
- **Microsoft のボリューム シャドウ コピー サービス (VSS)** は、Windows Server 上で実行される Zen Enterprise Server または Zen Cloud Server に実装された Zen Backup Agent の代わりに使用できます。Zen VSS Writer コンポーネントはインストールされると自動的に有効になり、VSS が完全なボリューム スナップショットを取得するまでデータベースを静止 (フリーズ) させます。その後、当該のバックアップ ソフトウェアでは、VSS が取

¹ Zen Workgroup for Windows では、Backup Agent は別売のオプション製品として提供されます。

得したスナップショットをマウントし、VSSがトランザクションにおいて一貫性のあるスナップショットを持っていることを把握したうえで、そのデータをコピーできます。

どのバックアップソリューションを選択した場合でも、Actian Zen データは常に Zen ファイル レベルでバックアップされます。データを復元する場合は、固有のニーズに応じて、データベース全体を復元することも 1 つのファイルのみを復元することもできます。

アーカイブ ログ

アーカイブ ログは、オペレーティング システムの定期的バックアップを補完するのに使用できるメカニズムです。アーカイブ ログを使用すると、最後のバックアップ以降の全データベース操作のログをファイルごとに保存することができます。システム障害が発生した場合、バックアップからデータ ファイルを復元し、ログ ファイルから変更をロールフォワードして、システムをシステム障害発生前の状態に戻すことができます。

Zen のセキュリティ

アプリケーションレベルでは、Actian Zen および Actian Zen ファイルへのアクセスは、オペレーティング システムや Active Directory アカウントを介したユーザー認証によって、もしくは名前付けした Zen データベース ユーザーとして保護することができます。操作レベルでは、Actian Zen は、SQL ベースのアプリケーションに対してはリレーショナル テーブル レベルとリレーショナル列レベルの整合性を、また、MKDE を呼び出す Btrieve アプリケーションに対してはファイル レベルの整合性を保証できるセキュリティ機能を提供します。Zen のアドオンである Actian Audit for Zen は、基盤となる MKDE レベルでアプリケーションに依存しない詳細な監査証跡を提供することで、各データベース イベントの発生元を監視および識別できるようにします。

Zen のセキュリティ モデル

Actian Zen の、リレーショナル データベース エンジンに対する保護方法と MKDE エンジンに対する保護方法は、少し異なります。

リレーショナル データベース エンジンについては、ローカル データベースごとに SQL 認証および許可の設定を行うことができます。Zen Enterprise Server または Zen Cloud Server をインストールした Windows プラットフォームの場合は、Windows ドメインおよび AD との統合を提供します。どちらの場合も、マスター ユーザー、グループ、およびグループ内の各ユーザーに対して、パスワード保護された権限を設定できます。権限の対象は、テーブル、テーブル内の列、ビュー、およびストアード プロシージャというように段階的に拡張できます。各データベースのユーザーやグループを管理するには、ZenCC GUI または SQL スクリプトを使用します。

また、MKDE セキュリティ モデルは、基盤となるオペレーティング システム環境によって提供されるユーザー認証および許可メカニズムを活用します。Actian Zen には、OS のユーザー アクセス権からデータベースの権限を分離し、データベース レベルのユーザーの資格情報と権限を使用する高度なセキュリティ オプションも用意されています。

Zen の暗号化

すべての Zen データ ファイルは、所有者名 (オーナー ネーム) による暗号化をサポートすることで、保存されているデータを保護します。Zen ファイル形式の以前のバージョンでは AES-192 ビットおよび AES-128 ビットの暗号化のみがサポートされていましたが、最新の Zen ファイル形式 (v13) ではディスク上のデータの AES-256 ビット暗号化がサポートされています。

Actian Zen では、Blowfish (40 ビット、56 ビット、および 128 ビットのキーを使用して実装) という有名かつ実績のあるパブリック ドメイン アルゴリズムを使用して、Zen クライアントとデータベース エンジンとの間のネットワークトラフィックを暗号化することもできます。このタイプの暗号化は、データがネットワークの「ワイヤ」上を移動するときに保護するため、ワイヤ暗号化と呼ばれていますが、ワイヤレスを含むあらゆるネットワーク インフラストラクチャで機能します。ワイヤ暗号化が要求されていないときは、アプリケーションによって送信されるデータへの不正アクセスに対する追加の抑止力を提供します。

Zen のワイヤ暗号化は、特定のセキュリティ モデルとは関係ありません。どの Zen セキュリティ設定も、ワイヤ暗号化をオンにしてもしなくても使用できます。

暗号化を使用する他のすべてのセキュリティと同様に、抑止力が大きいほど、暗号化アルゴリズムと復号化アルゴリズムの実行にある程度のプロセッサ時間が必要になるため、パフォーマンスが低下します。ファイル暗号化はファイルごとに有効にすることができますが、ワイヤ暗号化はクライアントごとまたはサーバーごとに無効にしたり、常時有効にしたり (Wi-Fi、WAN、または完全には信頼できないパブリック ネットワークの場合)、必要に応じて有効にしたりすることができます。

Zen のセキュリティ監査機能

Zen のアドオン製品である Actian Audit for Zen は、Zen の組み込みデータベース ユーザー アクセス管理に加えて、成功および失敗したログイン試行やデータベース操作などのデータベース イベントをログに記録し、ユーザー ID、ネットワーク ステーション ID、日時、アプリケーション名、データベース名、テーブルの種類、およびテーブル アクティビティを保存します。更新または削除されたすべてのレコードの変更前および変更後のイメージを記録します。また、一部の操作に対してだけですが、「元に戻す」機能もサポートします。Actian Audit for Zen は基盤となる MKDE レベルで機能するため、アプリケーションに依存せず、監査証跡内のすべてのデータベース イベントの発行元を監視および識別できます。

Zen の配置モデル

Actian Zen は、オフィス、データ センター、エッジ、およびクラウドに配置できます。サーバー、ラップトップ、携帯電話のほか、MRI や車両から IoT ゲートウェイ、カメラ、センサーなどまでの複雑な機器で稼働します。動作環境と基盤となるハードウェア プラットフォーム (11 ページの図 6 を参照) によって目的の Zen 製品をサポートするのに十分なリソースがある限り、Actian Zen を実行できます。

Zen の配置と構成は、アプリケーションに Zen をどのように組み込んで使用するかによって異なります (たとえば、Zen をローカルのデータ管理エンジンとして使用するか、クライアント/サーバー モードで使用するかなど)。このセクションでは、主な配置オプションについて説明します。

アプリケーション専用に組み込む

図 10 に示すように、Zen データベースの最小エディションである Zen Core のフットプリントは約 4MB です。これには、Zen データベース エンジンの主要な機能を提供する 3 つの組み込み可能なライブラリ モジュールが含まれます。これらのモジュールは、MKDE、SRDE、および Zen クライアント モジュール (Zen Edge、Zen Enterprise Server、および Zen Cloud Server の各エディションへの直接アクセスを可能にする) で構成されています。開発者およびエンジニアは、これらのモジュールをアプリケーションに直接組み込むことで、アプリケーションから、アプリケーションに組み込んだ Zen Core エンジンへの、または組み込んだクライアント モジュール経由によるリモート Zen サーバーへの、Btrieve API 呼び出しまたは SQL クエリを実現します。これにより、データ テーブルとそのビューがアプリケーション内でプロセスとして使い回されます。これらのデータ テーブルとビューは必要に応じて追加のメモリを消費しますが、消費できる追加メモリは、これらが搭載されているデバイスのメモリ制限によってのみ制約されます。アプリケーションごとに独自のデータ テーブルとサポート プロセスを備えて、データを分離することができます。

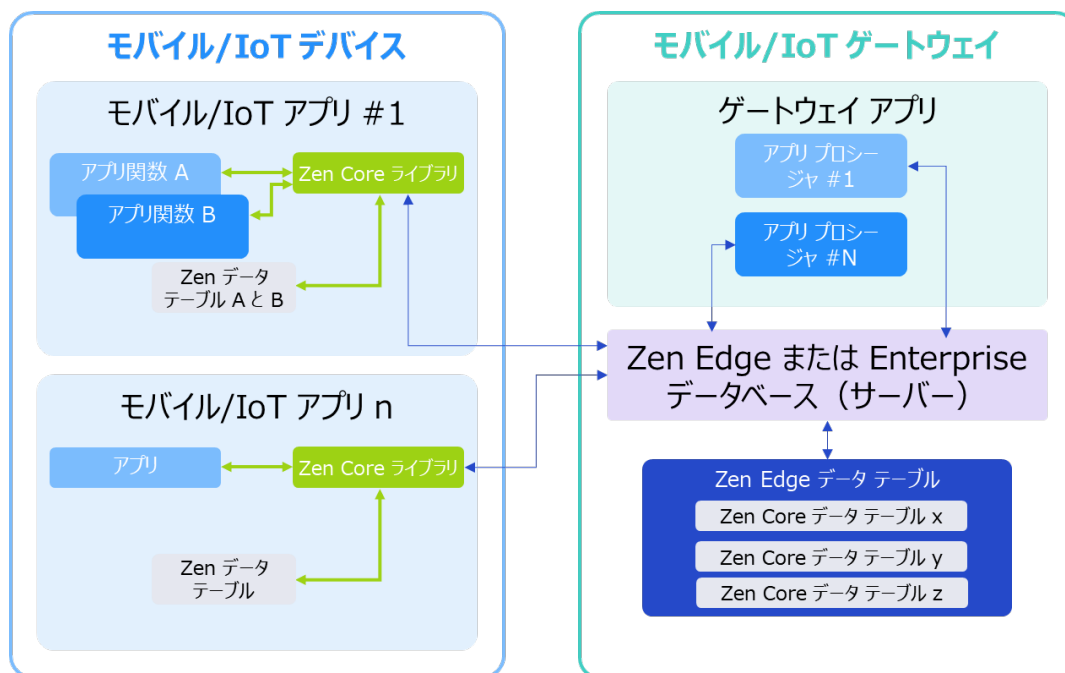


図 10: クライアント/サーバー構成の Zen Edge Server または Zen Enterprise Server に接続されたスタンドアロン アプリに組み込まれた Zen Core

開発者はアプリケーションに Zen Core を組み込むことができるため、エンドユーザーが Zen Core を意識することはありません。Zen Core を組み込んだアプリケーションの操作方法は、当該の動作環境で実行される他のファイルと同じです。

1 つ以上のアプリケーションをサポートするスタンドアロン

アプリケーションに組み込んだ配置モデル (前出) では、デバイスのローカル アプリケーション ロジックは組み込み型のデータ管理機能を含むので、ローカルの Zen Core エンジンを利用します。同様の配置モデルは、Actian Zen の Zen Edge エディションを使用すれば利用できます。ただし、この場合は、アプリケーション ロジックが必ずしもローカルの Zen データベース エンジンに関連付けられるとは限りません。

Zen Edge を使用する開発者は、Zen クライアント アプリケーション ロジックを分離して、他のデバイスでリモートで実行できます。このデバイスは、同じデバイス上で Zen データベース エンジンが実行されていてもいなくてもかまいません。Zen Edge は、サポートされているプラットフォームにあって CRUD (NoSQL) または SQL 操作を要求する Zen クライアントからの複数のデータベース接続要求を処理できます。このような配置の柔軟性は、たとえば、アプリケーションが複数のエッジ プラットフォームに分散されており、ピアツーピア方式でデータを共有する必要があるシナリオで活用できます。この柔軟性を活用できる別のシナリオとしては、重要なデータやメタデータの管理と分析の結果を、上流で集約するか、Zen クライアント アプリケーション ロジックを実行している下流デバイスに配布するシナリオがあります。

あるいは、Zen Edge では、異なるタイプの複数のアプリケーション間でデータを共有および融合できます。このシナリオでは、ステートフル (アプリケーション ロジックに基づく場合) とステートレス (元々の Zen データベースの機能を使用する場合) のどちらを使用してもかまいません。

これらのシナリオは、Zen Workgroup、Zen Enterprise Server、および Zen Cloud Server でも、Zen Edge の場合と同じ方法でサポートされます。これら大規模なエンジンでは、サポートされている API を介してデータにアクセスする複数のローカル アプリケーションがサポートされます。さらに、これらの大規模エンジンは膨大な数の Zen クライアントからのリクエストにも対応できるため、これらのエンジンを使用すると、ダウンストリーム クライアント (アップストリーム データのパブリッシャーまたはダウンストリーム データのサブスクライバーとして機能) としての Zen Core および Zen Edge にアクセスできるようになります。Actian Zen を使用した場合の結論としては、非常に異なるプラットフォーム上でさまざまなプログラミング言語で実行されているアプリケーションにわたってデータを分散、共有、集約、および配布できるプラットフォームを手に入れることができるようになります (ただし、どのアプリケーションも、ローカル リソースによって課せられる制約内で実行されています)。

クライアント/サーバー モードと複数の配置インスタンス

Zen Enterprise Server エンジンと Zen Cloud Server エンジンは、クライアントとサーバーの間で並行してやり取りされるデータの読み取りと書き込みを同時に処理できる Zen サーバーのエディションからのアップストリーム サポートを必要とする数千台のクライアント (Zen Edge や Zen Core またはサードパーティ製のデータ サービス/リクエスト) をサポートできます。データにアクセスするユーザーとプロセスは Zen Server や Zen Cloud Server 上で直接稼働することもできれば、環境内でサポートされている任意のデバイス (デスクトップ PC、クラウド サーバー、エッジ上の IoT デバイス、モバイル デバイスなど) 上で稼働することもできます。前述のように、どの Zen エンジンもすべての API を同時にサポートするので、一部のクライアントからの急増するデータ入力、その他のクライアントからのリレーショナル クエリとレポート、さらに他のクライアントからのシステム管理呼び出しを同時に実行することができます。

このパフォーマンスと柔軟性の組み合わせは、Zen サーバーがエッジにゲートウェイとして配置されている場合 (図 11 を参照) に非常に役立ちます。複雑な環境では、いくつかの異なるダウンストリーム IoT グリッドが作成されることは、簡単に想像できます。グリッドの一例は、圧力、容積、温度の読み取り値を測定する、Kbps 以下のデータ レート センサー用のシンプルなグリッドです。グリッドの別の例は、機械視覚を実行し、高速で膨大なデータ セットを生成する高度なカメラ システムと連動するグリッドです。Zen Cloud から Zen Enterprise、Zen Edge、Zen Core までの Zen サーバーが混在している場合、さまざまなレベルでデータを集約することで、他のレベルのサーバーまたはプロセスによる詳細な分析を行うことができます。

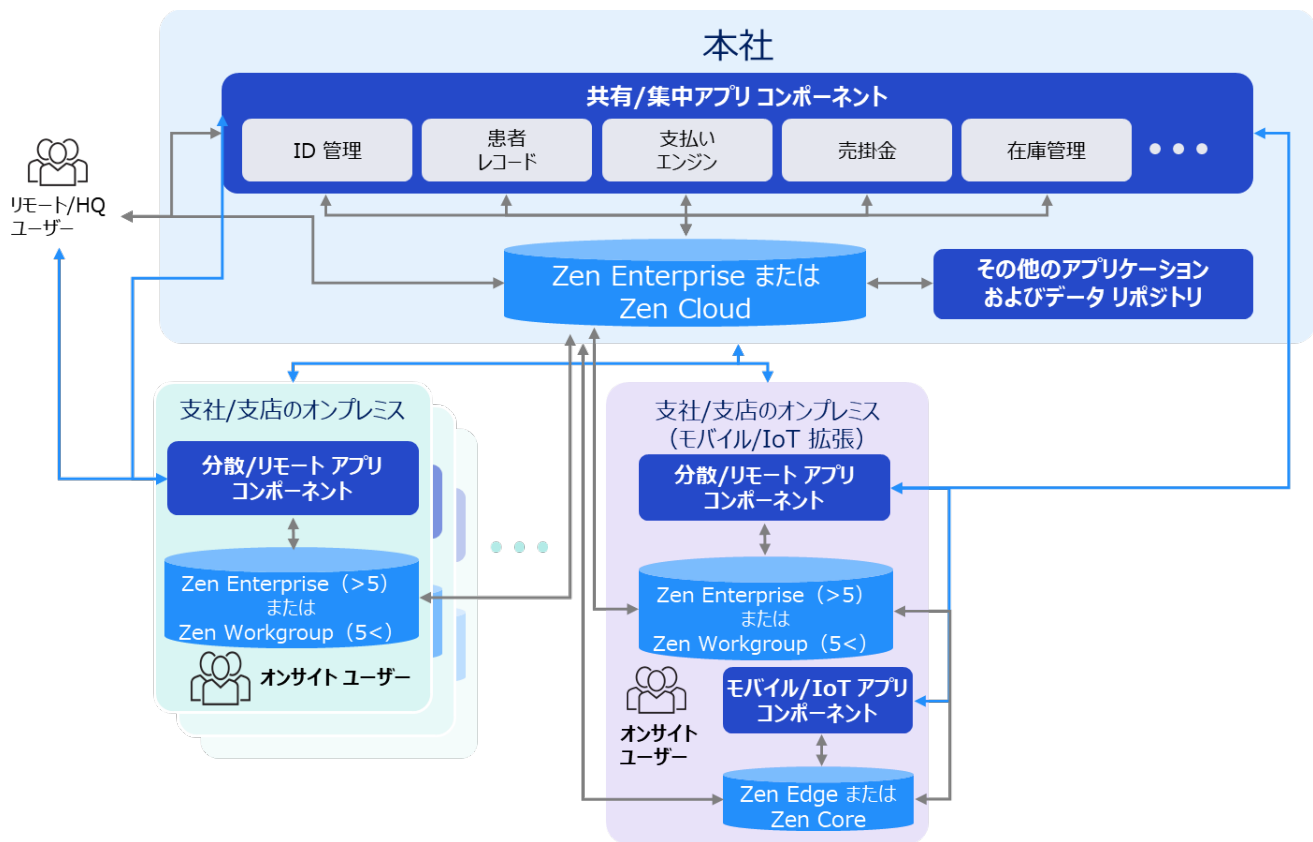


図 11: カスタム/垂直型マルチユーザー アプリケーションのデータ管理

Zen によるデータ統合

最先端のエッジ データ管理の世界で現在使用されているさまざまなシステム、プラットフォーム、プログラミング言語、およびデータ形式のことを考えると、どうしてもデータ統合の課題に対処する必要があります。遅かれ早かれ、Zen 以外のデータ形式を読み書きしたり、Zen 以外のアプリケーションにデータを渡したりすることが必要になります。多くの場合、これを行うために自動化ツールを活用しています。あるいは、独自のアプリケーション内または他のサービス内で ETL 機能を実行するカスタム コードを記述します。

ETL アクティビティは、IoT システムとモバイル アプリケーションの開発と統合に関連する時間とコストのかなりの部分を占めます。これを回避する方法はありません。データは常にデバイス間を移動する必要があるため、ある程度の抽出と格納がどうしても必要になります。データがいつ、どこへ、どのような方法と理由で移動するかという問題に対する回答は、イベント駆動型の視点からすると、ビジネス固有のユース ケースに左右されることが多く、そのニーズがアプリケーション ロジックやデータ管理の課題を非常に複雑にしています。

Action Zen は、以下の主要機能により、抽出および格納の操作に関連する時間と労力を大幅に削減できます。

- 抽出または格納する対象を迅速に決定するための、標準のクエリおよびサポート機能
- 自動的なタイムスタンプ付加、レプリケーション、およびバックアップ
- 接続が切断された場合の自動再接続

Zen は ETL の抽出と格納の部分的高速化できますが、最も時間がかかる統合タスクは ETL の**変換**部分です。さらに、データをシステム A からシステム B へ、システム B からシステム C へ移動するにつれて、変換に関連するパフォーマンスの低下と潜在的なセキュリティの脆弱性は大きくなります。

すべてが Zen 製品である環境を利用できるときは多くの場合、ETL の変換部分をほとんど排除できます。下記の図 12 は、異なるポイントでそれぞれのデータ管理システムが使用されている典型的な製造環境を示しています。使用されるデータ管理システムとしては、リソース制約のあるデバイスに対してはフラット ファイルが、モバイル デバイスに対しては SQLite が、必要なさまざまな機能に対してはそれらをサポートするその他のデータベースがあります。

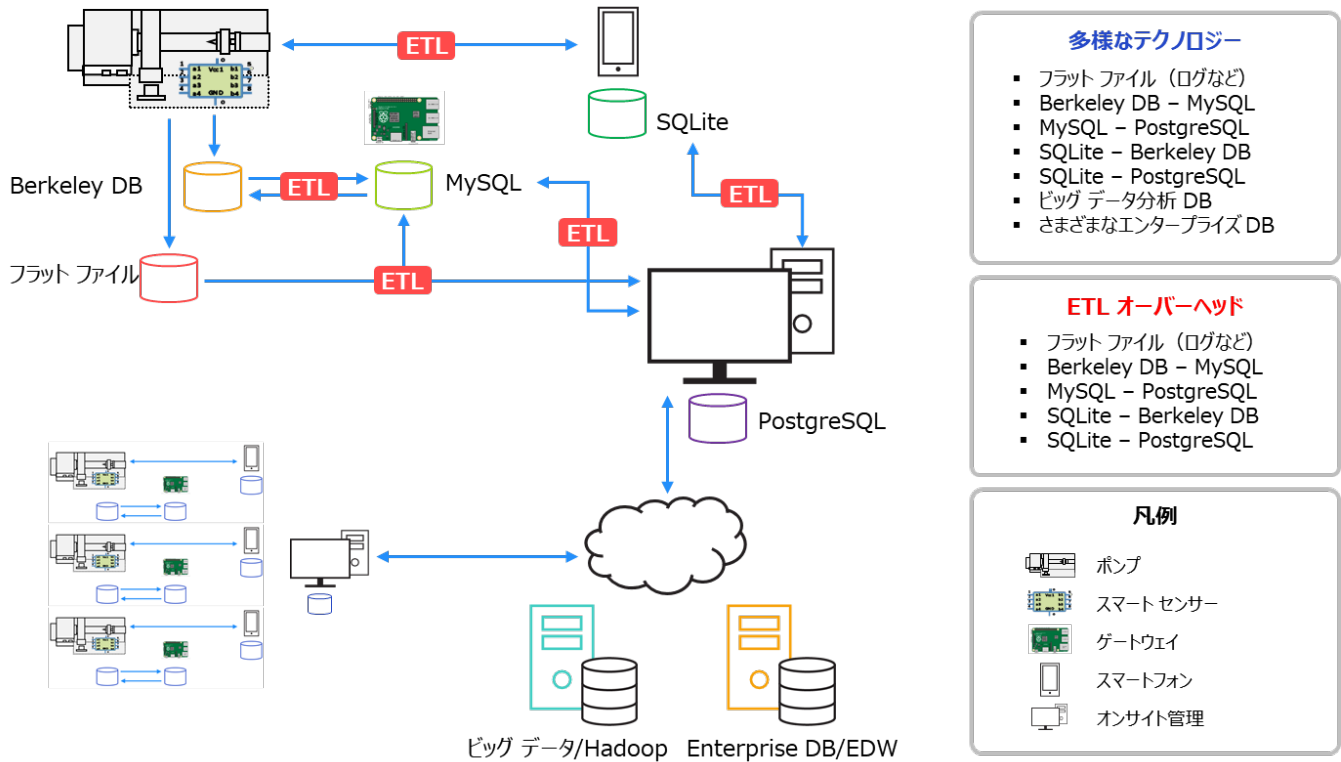


図 12: ETL がある異種環境

このシナリオの複雑さを図 13 に示すシナリオと比較してください。後者のシナリオでは、それらのフラット ファイル システムおよびデータベース システムのそれぞれを Zen のエディションに置き換えています。サポートされているすべてのプラットフォームに共通である Zen のデータ形式により、ETL の変換部分に代わるシンプルなファイル コピー操作が可能になります。さらに、すべての Zen エディションにおけるクライアント機能とサーバー機能の相互運用性により、任意のクライアント上のアプリケーションが任意の Zen サーバーのデータに直接アクセスできるようになります。これにより、データが作成されている場所に直接、リアルタイムでアドホックなアクセスをすることができます。

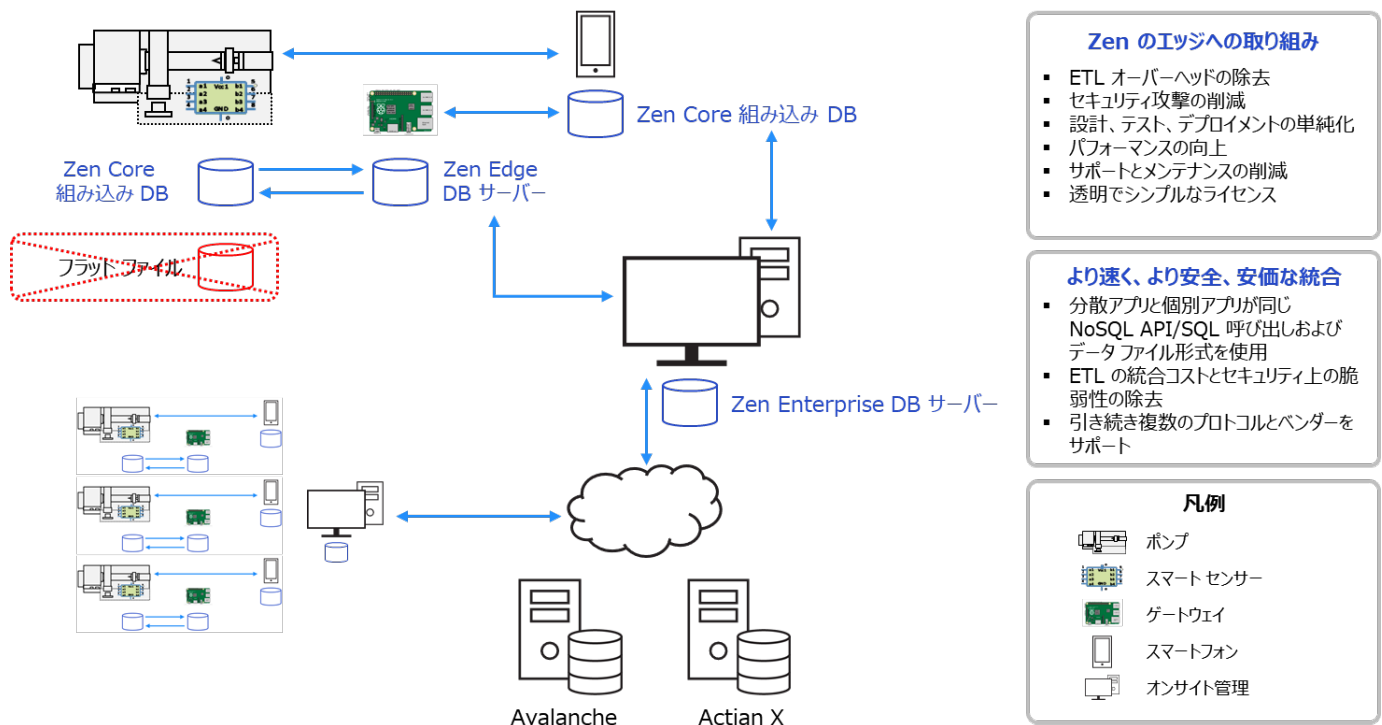


図 13: ETL がない同種環境

注意が必要な重要ポイントは次のとおりです。ETL がある場合、最新のデータにアクセスできるまでの時間差がどうしても生じます。全体を通して Zen を使用すると、その時間差がなくなり、任意のデバイスの最新データをドリルダウンしてポーリングできるようになります。Zen をインストールしたデバイスは、低速のバッチ ETL プロセスがデータを移行したり同期したりするのを待たずに、Control Center から最新の命令とアルゴリズムに直接アクセスできます。

ETL がない Zen 環境

わかりやすくするため、またパフォーマンスへの影響と統合コストの大部分が ETL で発生するため、Zen のみで構成される環境を「ETL がない(環境)」という用語で表します。Zen を同種環境(特にエッジ構成の場合)で使用することで、数多くの利点が得られます。

- さまざまなフラットファイル標準およびデータベース標準の要件を満たすためにデータを復号化、再編成、再フォーマットするコードによって発生するセキュリティ脆弱性を回避できます。
- 復元、復号化、および暗号化に関するデータの抽出および格納についてアプリケーション ロジックを実行することで発生するパフォーマンスのボトルネックを回避できます。
- 複数のアーキテクチャにわたるソリューションの設計と開発に関連する統合コストを回避できます。

アプリケーションと Zen 以外のデータの統合

Zen のみの環境を配置できない場合でも、標準 ETL 統合は Zen によって改善されます。Actian Zen では、次のような一般的な ETL ソリューションで使用されている代表的なインターフェイスがサポートされています。

- **ADO.NET:** Zen には、ASP.NET、Windows、およびコンソールアプリケーションからのデータアクセスとコマンド実行をサポートする ADO.NET SDK が用意されています。Zen ADO.net SDK の動作が保証されているのは、Windows 2019 まで、および ADO.NET データプロバイダー 4.3 以降です。Zen では、.NET Core 2.0 以降および Windows ユニバーサル プロジェクトを対象とする .NET アプリケーションをサポートする、ADO.NET SDK Standard をご用意しています。
- **JDBC:** Zen アプリケーションと Java アプリケーションとの間の通信については、Zen は、Windows、Linux、macOS 環境用 OpenJDK 内の JRE 8 をベースにした JDBC 4.1 ドライバー (バージョン 2 までの下位互換性あり) をサポートしています。
- **ODBC:** Zen は、任意の標準 DBMS と連携して動作する ODBC コネクタ (コアレベル 1 および 2 用の ODBC v3.51) を提供します。また、アプリケーションの API 言語や基盤となる動作環境に依存しない SQL アクセスを可能にします。

これらのメカニズムにより、ほぼすべてのアプリケーションやデータベース、JSON、XML ファイルなどの間でデータを転送できます。

結論

最先端のエッジ データ管理は、旧世代のデータ管理システムが満たすことができなかったさまざまな課題を提示します。このような課題としては、さまざまなデータ型の操作から、常時接続設定と不定期接続設定の複雑な組み合わせで通信する広範なシステムとセンサーの操作までが含まれます。膨大な量のデータと膨大な数のデバイスがあり、それらを取り込もうとする者にとっては膨大な数の機会があります。しかし、これを成功させるためには、新しいタイプのデータ管理ソリューションが必要です。

Actian Zen は、最先端のエッジ データ管理の要件を満たしている独自の位置にあります。Actian Zen には、ほとんどの開発者、データ エンジニア、システム アーキテクト、およびその業務パートナーのニーズに対応できる、データベース エディションのポートフォリオが備わっています。Actian Zen では以下を実現しています。

- 単一のモジュラー アーキテクチャ。モジュラー アーキテクチャのデータベース エディションは、小規模な組み込み可能なバージョンから本格的なクラウド サーバーまでの全範囲を稼働できます。また、モジュラー アーキテクチャの機能は、組み込み型のスタンドアロンアプリケーションでの永続的なデータセットのサポートから、数千ものアップストリームおよびダウンストリームのパブリッシャー/サブスクライバー間で同時実行される非常に動的なクライアント/サーバー通信のサポートにまで及びます。
- あらゆるデータ型 (JSON、BLOB、ドキュメント、従来の構造体など) のデータへの SQL または NoSQL によるアクセス。このようなアクセスを行うには、必要に応じて暗号化と圧縮の両方をサポートする MicroKernel データベース エンジン (MKDE) を使用することで、どのエディションに対しても不変な単一のファイル形式を採用します。
- C/C++/C#、Java、JavaScript、Python、Perl、Visual Basic、COBOL など、最先端のエッジ データ管理に最も関連するプログラミング言語とスクリプト言語を使用して、MKDE および基盤となる Zen データ ファイルにアクセスできます。
- ハードウェアおよび動作環境に依存しないデータベース アクセス (32 ビットおよび 64 ビットの Intel および ARM プラットフォームで実証済み)。
- OS およびプラットフォームに依存せず、ローカル システムとリモート システムの間で共有できるデータ ファイル。復号化、復元、または時間のかかるデータ変換は必要ありません。
- ローカルおよびリモートでの管理、バックアップ、共有、および監査に対する、シンプルでありながら包括的なサポートが組み込まれています。
- 現場およびリモートでの配置へのデータベース管理者不要アプローチ。このアプローチは、データベースの自動的なチューニングおよび構成のための組み込み API と、リモートでの構成と管理を容易にする組み込みユーティリティによって実現されます。

Actian Zen は御社の技術スタッフが求めている機能を提供します。御社の技術スタッフはそれらの機能を使用して御社全体の業績を改善できます。

Actian について—「Activate your Data™ (お客様のデータを有効活用)」

Actian は、ハイブリッド データの管理、分析、および統合を行う企業であり、世界各国に散在する数千のお客様に競争上の優位をもたらすデータを提供しております。Actian は、革新的なハイブリッド データのテクノロジーやソリューションを配置することで、基幹系システム同士がオンプレミスまたはクラウド、ある

いはその両方で通信し、かつ最高の状態で統合されることを保証します。全世界の数千の先進的な組織が、データに関する最も困難な課題を解決するソリューション プロバイダーとしての **Actian** を信頼して、現在および将来の事業経営方法の変革に取り組んでおられます。

Actian 社の詳細情報や取扱製品については、<https://www.actian.com> (英語) をご覧ください。



株式会社エージーテック

本社：〒101-0054 東京都千代田区神田錦町 1-17-5 Daiwa 神田橋ビル 6F

TEL：03-3293-5300（代表） FAX：03-3293-5270

カスタマーセンター TEL：03-3293-5283 Mail：info@agtech.co.jp