

ビッグデータ可視化プラットフォーム Polyspector™

Polyspector™ High-Speed Big Data Visualization Platform

李 新肖 松崎 秀則

■ LI Xinxiao

■ MATSUZAKI Hidenori

高度情報化社会の進展に伴い、様々なデータを収集して活用するビジネスが大きく発展している。データのトレンドや潜在的な問題点を、迅速かつ直感的に把握するには、データの絞り込み条件を変えながら可視化することが有効である。

東芝は、それぞれが独自技術の可視化専用データベースエンジンと最適集約アルゴリズムを組み込んだ、ビッグデータ可視化プラットフォーム Polyspector™を開発した。可視化専用データベースエンジンが、必要な部分データを高速に抽出し、最適集約アルゴリズムが、表示解像度に応じて、抽出したデータを必要最小限まで削減することで、高速可視化を実現した。1台のパソコン(PC)でのインタラクティブ操作で、億単位のデータの可視化を検証した結果、画面操作のレスポンス時間が1~3sに抑えられ、性能ワーストケースでの可視化画面のレスポンス速度を従来の100倍以上に改善できた。

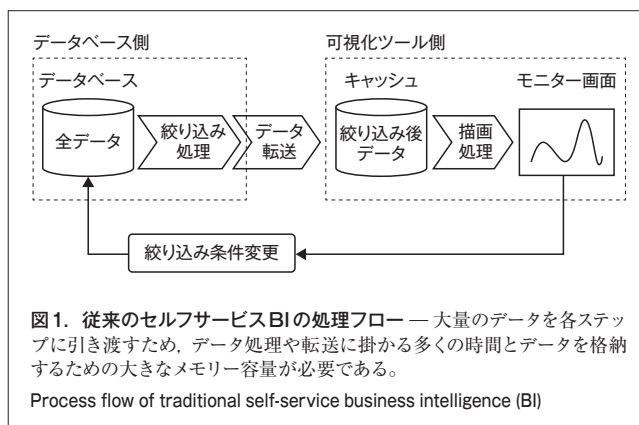
In line with the progress of the advanced information society, demand has recently been expanding for the collection and utilization of big data in various business fields. An effective means of rapidly and intuitively grasping trends and potential problems in such data is to visualize the data according to narrowed-down conditions.

In this context, Toshiba has developed Polyspector™, a big data visualization platform incorporating its proprietary database engine for dedicated visualization and an optimal aggregation algorithm. Polyspector™ makes it possible to perform big data visualization processing at high speed through both high-speed extraction of partial data necessary for processing by means of the database engine and reduction of the extracted data to the minimum level necessary for visualization by means of the optimal aggregation algorithm. We have conducted verification tests with more than a hundred million data processed by interactive operation using a personal computer, and confirmed that the response speed of the visualized display in even the worst-case performance is improved more than a hundredfold compared with that of conventional tools while reducing the response time required for the display updating operation to only 1 to 3 seconds.

1 まえがき

高度情報化社会の進展に伴い、様々なデータを収集して活用するサイクルでは、新たな付加価値を創出するビジネスが大きく発展してきている。また、収集したデータを活用する過程では、探索的データ分析を支援する可視化システムのインタラクティブ操作で、収集したデータのトレンドや潜在的な問題点を直感的に把握できる、セルフサービスBI(ビジネスインテリジェンス)ツールの導入が進んでいる。

従来のセルフサービスBIツールでは、図1に示すように、可視化をデータの絞り込みや、データ転送、描画処理などの複数のステップに分けて行っている。まず、ユーザーが設定した検索条件を、SQL(Structured Query Language)文などでデータベース側に伝え、データベース側でデータの検索や絞り込みを行い、結果を可視化ツール側に転送する。その後、可視化ツールは、転送されたデータをいったんメモリーなどのローカル媒体に格納(キャッシュ)し、それから描画処理をして画面上に表示する。したがって、ローカル媒体に格納されたデータの範囲内であればデータ転送の必要がないが、この範囲を超えると、再び検索条件設定から画面表示までの処理を行うの



で、可視化に多くの時間を要する。

また、データ分析は、試行錯誤で、範囲選びやノイズ除去を行いながら結論を導き出す重複作業である。したがって、データ量の増大は、①転送時間の無視できない増加や、②転送されたデータの一時格納(大容量のメモリー消費)によるシステムの全体性能低下、③逐次描画のために生じるデータ量に比例した処理時間増大などの問題を引き起こし、ユーザー操作時のレスポンス性能の低下をもたらす。

そこで、東芝は、それぞれが独自技術の可視化専用データベースエンジンと最適集約アルゴリズムを組み合わせることで、大量データの高速可視化処理を実現した、ビッグデータ可視化プラットフォーム Polyspector™ を開発した。ここでは、その特長について述べる。

2 高速可視化処理

2.1 可視化専用データベースエンジン

データを列ごとに分け、各列データ内で時間粒度や空間粒度による階層的構造を作り、更に各階層内部をその粒度（時間や空間）での範囲ごとに分割してソーティングする。そして、可視化画面に使われる列において指定された範囲に関係するデータの列、粒度、及び範囲を特定することで、データ絞り込みを高速化している。

2.2 最適集約アルゴリズム

インタラクティブ操作で、視点や絞り込み条件を変更する際、可視化専用データベースエンジンで取得したデータは、更新された条件に最適化されていないケースが多い。

このアルゴリズムは、ビッグデータにおいて、モニター画面の表示解像度がデータ数よりはるかに少ないことを利用し、各データが画面上にマッピングされる位置を予測して画面上で重なると判断されたデータをまとめる⁽¹⁾。

モニター画面の同じ位置にマッピングされる複数のデータをまとめ、カラーや形で表現することで、データが理解しやすくなる。また、表示解像度に応じてデータ量を減らすことで、転送時間、一時格納でのメモリー量、及び描画時間を減らすことができる⁽²⁾。

2.3 高速可視化の処理フロー

処理フローを図2に示す。まず、インタラクティブ操作での視点や絞り込み条件の変更に応じ、可視化専用データベース

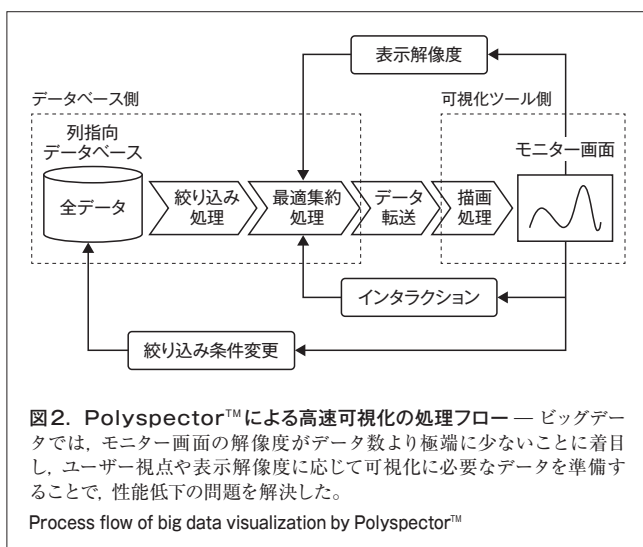


図2. Polyspector™による高速可視化の処理フロー — ビッグデータでは、モニター画面の解像度がデータ数より極端に少ないことに着目し、ユーザー視点や表示解像度に応じて可視化に必要なデータを準備することで、性能低下の問題を解決した。
Process flow of big data visualization by Polyspector™

エンジンが、絞り込んだデータを取得する。続いて、最適集約アルゴリズムが、画面の表示領域の解像度（縦と横の画素数）に応じてデータの最適集約を行う。最後に、集約されたデータを可視化ツール側に転送し、描画処理を行う。この一連の処理は、可視化ツール側でのインタラクション操作で可視化データ範囲（ズームイン/ズームアウト）や絞り込み条件が動的に変更されるたびに実行される。

高速可視化処理の例として、図3を用い、パラレルコーディネートによる可視化を説明する。パラレルコーディネートは、多次元データを直感的に把握するために使われる。各次元は、並行した軸で表現され、行ごとのデータは、各軸上の対応する位置を通る折れ線（又は曲線）として表現される。

4次元（温度、湿度、照度、快適度）データの高速可視化は、行数が大量であることを想定し、以下の三つのステップで実現される。

- (1) テーブルデータの各行、例えば (10, 60, 300, 80) において、各列の範囲を画面上の高さ (100画素) で規格化したときの値 (20, 60, 5, 30) を各行のインデックス (各軸の縦方向の相対座標) とし、それらを連結した複合インデックス “20-60-5-80” を作る。
- (2) 同じ複合インデックスの行をカウントしてまとめ、複合インデックスを主キーとするテーブル (複合インデックス, カウント数) を集約結果とする。
- (3) 集約結果のデータを用い、複合インデックスを折れ線に、カウントをラインのカラーにマッピングしたパラレルコーディネートを描画する。

例えば、図3で、集約結果 (20-60-5-80, 1,000) はピンク色の折れ線に対応し、(60-90-50-10, 10) は青色の折れ線に対応する。これにより、ピンク色と青色の2本の折れ線で、合計1,010個のデータを表現できる。

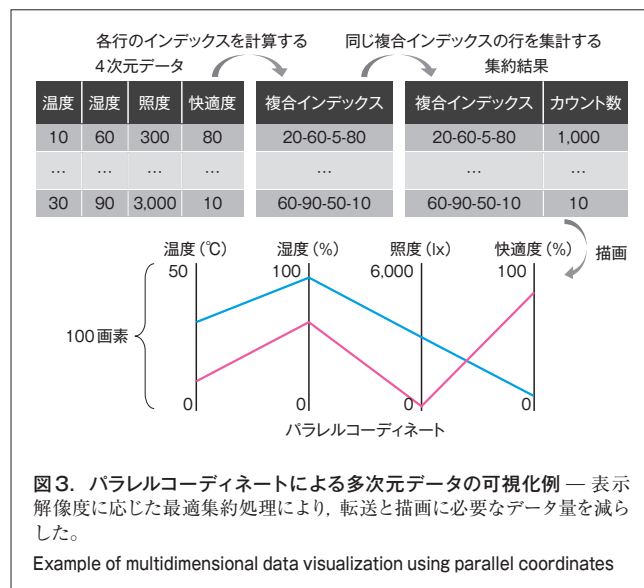


図3. パラレルコーディネートによる多次元データの可視化例 — 表示解像度に応じた最適集約処理により、転送と描画に必要なデータ量を減らした。
Example of multidimensional data visualization using parallel coordinates

3 可視化プラットフォーム

Polyspector™は、図4に示すように、ワーカー、フロントエンド、及び可視化ツールの、モジュール化された3要素で構成される。様々なデータソースへの接続を想定した汎用のセルフサービスBIとして、各モジュールを疎結合することで円滑なデータ受渡しを行う。ワーカーとフロントエンドの間はメッセージキューで隔離して汎用性を高めており、フロントエンドと可視化ツールはネットワークによって接続される。

3.1 ワーカー

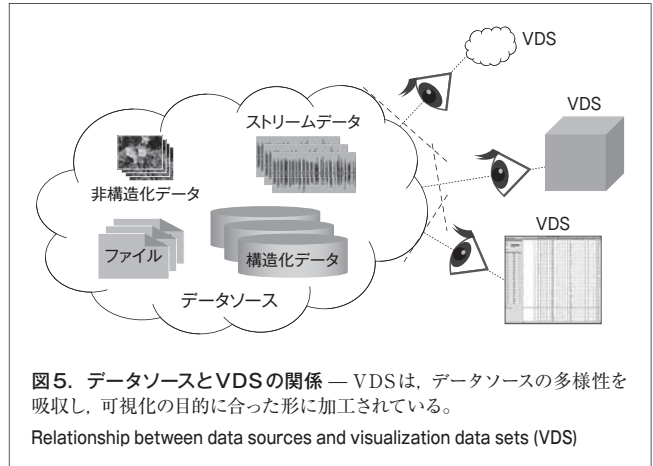
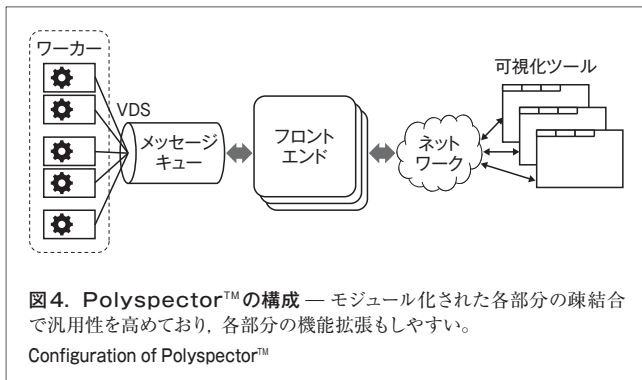
高速可視化の実現に中心的な役割を果たすもので、データソースへ接続してデータ取得とデータ集約処理を担当し、可視化ツールが使える形にデータを加工する。

可視化対象は、目的がはっきりしないままの様々なデータソース（データベースや、ファイル、画像データ、ストリームデータなど）であり、活用や分析を行う際は、ワーカーが目的に合った形にデータを加工して、可視化ツール側に渡す。

ワーカーから出力されるデータはVDS (Visualization Data Set) と呼ばれ、フォーマットは利用の目的によって主に4種類に分けられる。

- (1) テーブルフォーマット この種類のデータを描画できるチャートを基本チャートと呼び、汎用の基本チャートには、ラインチャートや、バーチャート、散布図、エリアチャートなどがある。
- (2) ツリー構造フォーマット 少ないデータ量でツリー構造（ツリーマップなど）を視覚化できる。
- (3) タイムシリーズフォーマット ストリーミングデータなど、時間軸にソーティングされたデータで、圧縮や高速化がしやすい。
- (4) 独自（自由）フォーマット フォーマットを解釈するロジックが必要で、対応できるチャートにしか使われない。

ワーカーは、可視化ツール側のUI（ユーザーインターフェース）上での絞り込み条件や描画領域サイズ（解像度）などの指示に従い、データソースから必要な範囲のデータを取得して描画領域サイズのデータに集約し、VDSとして可視化ツールに



転送する。VDSは、図5に示すように、描画領域サイズに応じたユーザー視点でデータソースを見たスナップショットであり、その視点で見えない情報は除外されている。

3.2 可視化ツール

Webブラウザにロードされて実行され、様々なチャートを提供する他、画面解像度や、ユーザー操作で選んだデータ範囲を特定し、それらの情報をフロントエンド経由でワーカーに送る。また、ワーカーからVDSを受け取り、可視化画面の生成や更新を行う。

3.3 フロントエンド

ワーカーと可視化ツールの間に配置され、データ管理や、アクセス権限管理、可視化ツールとワーカー間のデータ授受など多岐にわたる役割を果たす。データ管理は、複数のワーカーが提供するVDSを収集して、可視化可能なデータリストを可視化ツールに提供し、アクセス権限管理は、どのユーザーをどのワーカーにアクセスさせるかといった権限を統一的に管理する。

4 可視化の応用例

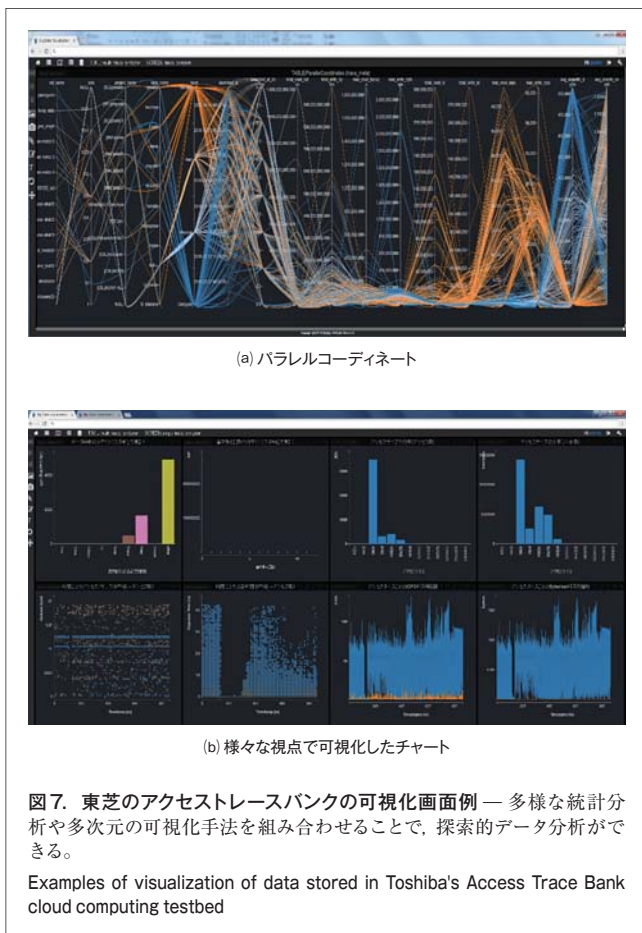
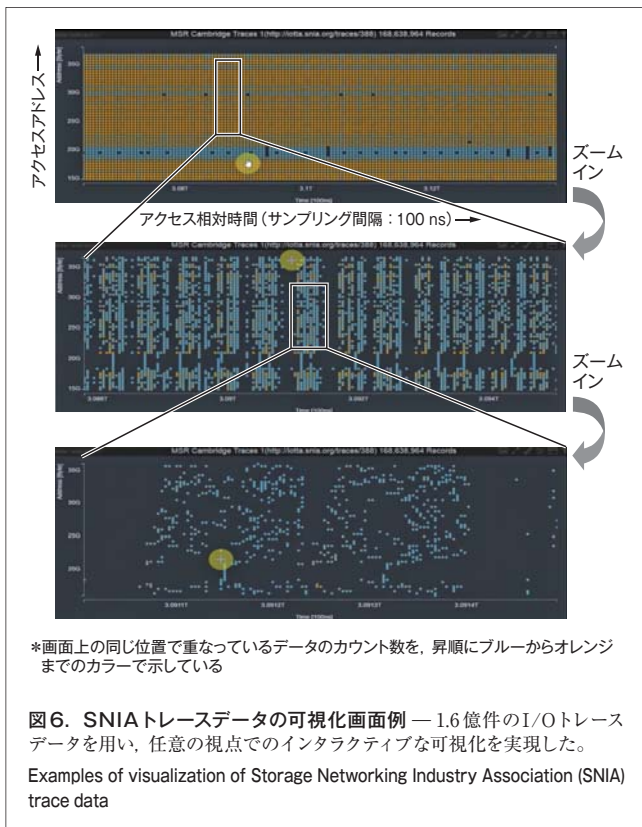
4.1 SNIAのI/Oトレースデータの可視化

SNIA (Storage Networking Industry Association) のI/O (Input/Output) トレースデータ⁽³⁾1.6億件を可視化した画面を図6に示す。全体の俯瞰（ふかん）から、インタラクティブ操作によるズームイン/ズームアウトで時間やアドレスを絞り込み、注目する範囲のアクセスパターンを視覚的に分析できる。

1台のPCでシステムを構築して検証した結果、ユーザー操作から画面更新までの時間が1～3sに抑えられた。性能ワーストケース（全データ範囲のオーバービュー）での可視化画面の操作レスポンス速度が従来の100倍以上に改善し、インタラクティブな可視化操作が実現できている⁽¹⁾。

4.2 アクセストレースバンクの可視化

当社内で使われているクラウドテストベッドであるアクセストレースバンクは、ストレージI/Oトレースデータを収集（1 T



(テラ: 10^{12} バイト/日) し、アクセス日時や、プロジェクト名、書込み総量、ピック書込み速度、書込みサイズなどを変量とした多次元データ (28次元 \times 100レコード/日) にまとめてデータベースに蓄積する。蓄積された多次元データで、アクセスパターンの検出などのマイクロな視点から、統計情報による全体俯瞰などのマクロな視点まで、ストレージI/Oを分析できる (図7)。

図7(a)は、パラレルコーディネートで28次元データを可視化した画面で、全体のデータ分布を把握しながら、次元間の相関関係をインタラクティブ操作で解明できる。書込みタイプや書込みサイズなどの複数の視点で統計データを計算し、バーチャートや、ラインチャート、ヒストグラムなどで可視化したものが図7(b)である。チャート間が連係し、一つのチャートの範囲選択で他のチャートもその範囲に再描画されるので、複雑なパターン分析ができる。

5 あとがき

データのトレンドや潜在的な問題点の直感的な把握にはセルフサービスBIツールが効果的であるが、ビッグデータ時代では、可視化における高速化が重要である。

そこで、当社は、それぞれが独自技術の可視化データベースエンジンと最適集約アルゴリズムを組み込んだ、ビッグデータ可視化プラットフォームPolyspectorTMを開発した。億単位の実データを用いたインタラクティブ操作による可視化の検証で、性能ワーストケースのレスポンス速度を、従来の100倍以上に改善できた。

今後、PolyspectorTMを多分野に適用し、探索的データ分析の活用範囲を拡大していく。

文献

- (1) Li, X. et al. "Advanced aggregate computation for large data visualization". Proc. 2015 IEEE 5th Symp. Large Data Analysis and Visualization (LDAV). Chicago, IL, 2015-10, IEEE. 2015, p.137-138.
- (2) Li, X. et al. "Distributed Aggregate Computation between Server and Client for Interactive Visualization". Proc. 2015 IEEE 5th Symp. LDAV. p.135-136.
- (3) SNIA. "I/O Trace Data Files". SNIA IOTTA Repository. <<http://iota.snia.org/traces/>>, (accessed 2017-05-10).



李 新肖 LI Xinxiao, Ph.D.

技術統括部 研究開発センター コンピューターアーキテクチャ・セキュリティラボラトリー主任研究員、博士 (工学)。ビッグデータ可視化技術、データマネジメント、及びデータ分析の研究・開発に従事。Computer Architecture & Security Systems Lab.



松崎 秀則 MATSUZAKI Hidenori

技術統括部 研究開発センター コンピューターアーキテクチャ・セキュリティラボラトリー主任研究員。ビッグデータ可視化技術及びNAND型フラッシュメモリー応用ストレージシステムの研究・開発に従事。情報処理学会会員。Computer Architecture & Security Systems Lab.