D-034

データストリーム処理手法を用いた Web アクセス解析システム
A Web Access Analysis System Using Data-Stream Processing Technique

今井 照之† 海老山 知生† 喜田 弘司† 藤山 健一郎† 中村 暢達†
Teruyuki Imai Tomoo Ebiyama Koji Kida Ken-ichiro Fujiyama Nobutatsu Nakamura

1. はじめに

Web アクセスログを収集しアクセス数の推移や訪問者の動線を解析する Web アクセス解析システムがプロモーションの効果測定，サイトの問題点の洗い出し，コンテンツ推薦サービスなどに活用されている．従来の Web アクセス解析システムでは，数千から数万ページのログを日単位に蓄積しバッチ的に解析しているため，解析結果は日単位の遅延がある．このためプロモーションの反応を即座に解析するといった遅延の少ない解析はできなかった．

本稿では，データを蓄えずに発生毎に処理するデータストリーム処理手法を用いることで，高速に Web アクセスログを解析する方式を提案する．

2. Web アクセス解析システムとその課題

従来の Web アクセス解析システムは，日々のアクセスログをデータベースで集計することで様々な角度から統計解析するシステムである．例えば，訪問者毎にページ遷移を追跡し，迷子になっている状況を検出することでサイト構造の問題を検知できる．

ところが，これらの解析は，日単位の遅延がある．今日の Web サイトでは，blog や SNS 等，非常に更新頻度の高いコンテンツが増えているので，低遅延な解析が望まれている．また低遅延な解析が実現すれば，メールによるプロモーションの効果を即座に把握する，訪問者の興味・関心に関する仮説，検証のサイクルの細かい検証など，より高度な活用が考えられる．

遅延の少ない Web アクセス解析システムを実現するには，従来のデータベースを用いた方式では限界がある．ログの登録，インデクシング等の処理は負荷が高く夜間にバッチ処理することが通常であるため，日単位の遅延が発生する．したがって，ログの発生から解析結果への反映までの遅延が少ない高速 Web アクセスログ解析が課題である．

3. データストリーム処理による高速解析

3.1 本システムの構成

本稿では，Web アクセス解析システムをデータベース (DB) 系とストリーム系の 2 系統に分ける構成を提案する (図 1)．DB 系では，従来と同様，全てのログを使った複雑な解析をクエリの要求の度に行う．ストリーム系では，高速性を要求される解析要求を事前にデータストリーム解析サーバに登録しておき，データストリーム処理により高速に解析する．

†NEC サービスプラットフォーム研究所
Service Platforms Research Laboratories, NEC Corporation

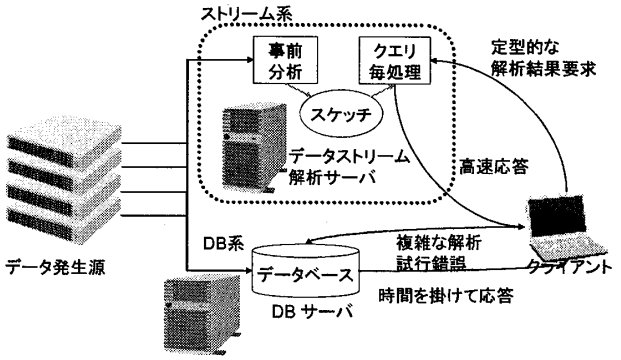

図 1 本システムの構成

以下，ストリーム系について述べる．まずデータストリーム処理手法について述べ，次に Web アクセス解析システムへの適用方法を述べる．

3.2 データストリーム処理手法

データストリーム処理手法は，データを貯めることなく，発生したデータをその収集過程で流れ作業的に解析処理を行う手法である (図 2)．

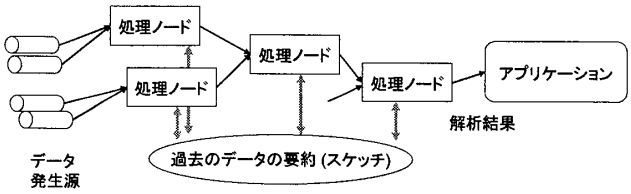

図 2 データストリーム処理手法

データストリーム処理では，解析処理全体を小さい処理単位 (処理ノード)のネットワークとして構成する．各処理ノードはデータにより駆動される．すなわち，処理ノードは，前の処理ノードからデータを受け取る度にこのデータを加工し，その結果を次の処理ノードへ渡す．この一連の動作を全ての処理ノードが繰り返すことで解析が行われる．このように，処理ノードはパイプライン的に動作するため，スループットを高めることが出来る．

各処理ノードはパイプのように，過去のデータは使用せず，受け取った最新のデータのみを対象として解析する．しかし，統計処理やデータ比較など，過去のデータが必要な場合がある．その場合，過去のデータや解析結果を要約して記憶するスケッチと呼ばれる共有メモリを使用する．

3.3 データストリーム処理手法の適用

データストリーム処理手法を用いてデータを低遅延に解析するには，処理ネットワークを適切に設計する必要がある．例えば，本来並列処理できる処理ノードを直列に接続すると遅延が大きくなる．データストリーム処理を用いて低遅延な解析を実現する要件を以下に挙げる．

要件1　同期の必要な解析間は処理ノードをつなぎ，非同期に可能な解析を行う処理ノードはつながない，
要件2　処理ノードで共有するデータは，小さいサイズに要約する．
要件3　並列に可能な解析は，独立した処理ノードとし，依存関係のある解析は直列につないだ処理ノードでパイプライン化する，

これらの要件を踏まえ，データストリーム処理手法を用いた Web アクセスログ解析システムの処理ノードのネットワークについて述べる (図 3).

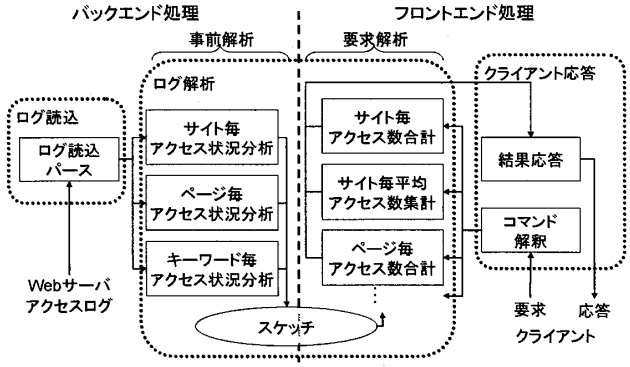

図 3 処理ノードの構成

Web アクセス解析システムにおけるストリーム系は，以下の処理から構成される．

- ログ読込：Web サーバから発生するログを発生毎に読み込み，必要なデータを抽出 (ログ読込・パース) する．
- ログ解析：読み込んだデータを逐次解析する．この解析には，クライアントからの要求に依存せず事前に解析できる部分 (事前解析) と，クライアントからの要求に応じた解析 (要求解析) がある．例えば，ページ毎のアクセス数をカウントする場合，時間毎にカウント (事前解析) しておき，クライアントから要求されたカウント期間に応じてアクセス数の合計を要求解析として回答する．
- クライアント応答：クライアントからの要求を解釈し，ログ解析にクエリを発行してその結果を応答する．

要件 1への対応：　ログ読込はデータが発生する度に実行され，クライアント応答はクライアントからの要求に応じて実行されるため，非同期に動作する．一方，ログ解析の内，事前解析はログ読込と同期し，要求解析はクライアント応答と同期している．そこで，処理ノードのネットワークを，ログの発生毎にログ読込から事前解析までを処理するバックエンド処理と，クライアントからの要求毎に要求解析とクライアント応答を処理するフロントエンド処理で構成する．

要件 2への対応：　フロントエンド処理とバックエンド処理の間では，事前解析の結果をスケッチとして共有する．事前解析の結果は膨大なサイズになるため，解析結果の時系列を近似式で要約するアルゴリズム[1]を用いる．

要件 3への対応：　バックエンド処理では，ログ読込と事前解析には依存関係がある一方，事前解析内の各種解析 (カウンタ，ランキングなど) は独立に実行できる．そこで，ログ読込と事前解析はパイプライン化し，各事前解析処理は並列に行う．一方，フロントエンド処理では，コマンド解釈，要求解析，結果応答には依存関係があるが，要求解析内の各種解析は独立に実行できる．そこで，コマンド解釈，要求解析，結果応答はパイプライン化し，各要求解析処理は並列に行う．

3.4 考察

本システムの遅延について考察する．データストリーム処理では，ログ発生後すぐに処理を開始するため，バッチ処理のような処理開始待ちが発生しない．また，過去のデータへはアクセスせず最新のデータのみを逐次処理するため，各処理で扱うデータセットが小さく，過去の生データではなく要約されたスケッチを記憶することによるオンメモリ処理と，パイプライン化・並列化により高速な解析が行われる．このように，本システムは低遅延な Web アクセス解析を実現している．

一方，本ストリーム系では，解析方法 (カウンタ，ランキング等) は予め登録する必要があり，クライアントは決められたパラメタ (カウントやランキングの期間など) のみ指定できる．しかし，この制約は実用上問題ない．例えばプロモーションの効果把握ではアクセス数の変化というように，必要な解析の種類が決まっていることが多いためである．また，登録されていない解析も，DB 系を用いて従来と同様に可能である．

4. 試作システム

我々は，データストリーム処理手法を用いた Web アクセス解析システムを試作した．解析対象とするアクセスログは，実運用している大規模 ISP の Web サーバのログであり，1 秒当たり 100 件，1 日当たり 1GB 程度のデータ量である．試作システムで行う解析は，サイト単位，ページ単位にアクセス数のカウント，アクセス数ランキング，検索キーワードランキングである．

ユーザは解析クライアント (図 4) により，これら解析結果の時系列変化をグラフで見ることが出来る．いずれの解析もデータ発生から数秒以内の遅延で最新の解析結果を得ることが出来た．

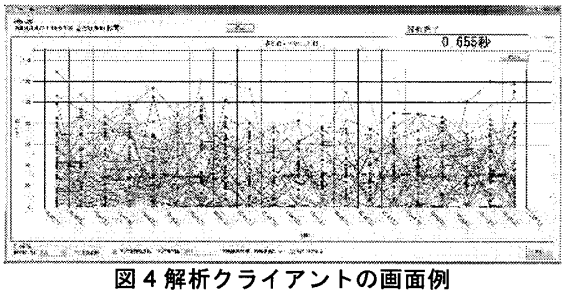

図 4 解析クライアントの画面例

謝辞

本研究の一部は，総務省からの委託研究「ユビキタスサービスプラットフォーム技術の研究開発」の成果である．また，NEC ビッグローブの協力を受けて実施した．

参考文献

[1]海老山 他，データストリーム処理を高速・省メモリで行うためのスケッチ生成方式，FIT2009