

大規模データ分析のためのオブジェクト SaPO の提案

田 仲 正 弘^{†1} 田 浦 健 次 朗^{†2} 鳥 澤 健 太 郎^{†1}

大規模データ分析には、高効率な分散並列処理、数年のスパンで常時流入するデータの継続的な分析、データ分析において本質的な操作である多種多様な分析の試行錯誤のサポートや分析プログラムのアップデートへの対応、計算機環境の移行に伴うマイグレーションの短時間化、容易化など、プログラム実装や運用上の数多くの課題がある。本稿では、大規模データ分析のプログラミングと運用を容易にするためのオブジェクト SaPO (Self-adaptive Persistent Object) を提案する。

SaPO: An Object for Large-scale Data Analysis

MASAHIRO TANAKA,^{†1} KENJIRO TAURA^{†2} and KENTARO TORISAWA^{†1}

Large-scale data analysis poses a wide variety of issues in both development and operations including efficient distributed processing, continuous processing on streaming data over several years, support of trial and errors that are essential for various analysis, and reducing time and cost of migration of computation environments. In this paper, we propose SaPO (Self-adaptive Persistent Object) as the solution, which is an object with some features for simplifying development and operations for large-scale data analysis.

1. はじめに

近年、言語処理や科学技術計算を初めとする様々な分野で、大規模データ分析が注目を集めている。情報通信研究機構が開発した大規模 Web 情報分析システム WISDOM X^{*1} や対災害 SNS 情報分析システム DISAANA^{*2} でも、Web から収集した数十億件の Web 文書や、常時流入する Twitter のテキストに対して、各種の深い言語処理を適用することで、質問応答を初めとする様々な分析を可能としている。

しかし、こうした大規模データ分析を行うシステムを開発し、継続的に運用することは容易ではない。本稿では、大規模データ分析の課題を概説するとともに、その解決として、オブジェクト SaPO (Self-adaptive Persistent Object) を提案する。

2. 大規模データ分析の課題

前述の WISDOM X や DISAANA では、数十種類

のプログラムや各種のデータベースを連携させ、テキストの深い意味的分析を行っている。そうした深い意味的分析のための処理は、高い計算負荷を生じる一方で、高スループット、低レイテンシが求められる。

例えば WISDOM X では、Web 数十億ページを情報源としているが、各種の分析プログラムは日々改良が加えられていること、また up-to-date な情報を常に提供していくことなどを考慮すると、1日あたり1億ページ程度のスループットが必要である。また DISAANA においては、緊急を要する災害関連情報をリアルタイムで分析するために、低レイテンシが重要であることは言うまでもない。

こうした要求を満たすには、構文解析器等の分析プログラムの実行プロセスを、与えられた入力を処理した後も終了させず、プーリングして使い回すことにより、起動オーバーヘッドを隠蔽したり、高速なストリーム通信によって、レガシーなものも含む多数の分析プログラムを効率的に連携させることが必要となる。さらには、様々なスペックからなる多数の計算機に、スループットを最大化、あるいはレイテンシを最小化するように、自動的に分析プログラムを配置するといった機能も重要である。

以上は比較的短い時間的スパンでの課題であるが、システムを長期的に公開したり、商用で提供するような場合には、他にも様々な課題が生じる。例えば、ク

^{†1} 情報通信研究機構
National Institute of Information and Communications Technology

^{†2} 東京大学大学院情報理工学系研究科
Graduate School of Information Science and Technology, The University of Tokyo

*1 <http://wisdom-nict.jp>

*2 <http://disaana.jp>

ラストの耐用年数が来た際に、新しいクラスタの上にシステムをインストールし、数百テラバイト単位のデータをコピーすることは容易ではなく、膨大な作業と長いダウンタイムが生じる。また、ハードウェアのメンテナンスの際には、分析プログラムを中断させる必要が生じるが、多様な分析プログラムが複雑に関係しあっている状況では、一旦中断した処理を再開することは容易ではない。さらには、分析プログラムの更新時には、データを再度分析する必要があるが、そうした操作をシステマティックかつ容易に行うメカニズムの必要性も高い。

3. SaPO

SaPO (Self-adaptive Persistent Object) とは、オブジェクト指向プログラミング言語から生成・操作される、分析プログラムと処理対象データを持つオブジェクトである。多数のオブジェクトの並列実行により、分散並列処理をモデル化するという点で、アクターを基盤に用いながら関連モデルや MapReduce など様々な処理モデルを実現する epiC²⁾ に似るが、大規模データの処理には長時間を要することから、PJava¹⁾ 等のオブジェクト指向データベースと同様、SaPO はプログラムの開始と終了のサイクルを超えて永続化される。また、計算機間の移動、複製、処理対象データの分割、マージ等のメソッドを提供する (図 1)。

SaPO は内部にデータベースと分析プログラムを持ち、SaPO のメソッド呼び出しによって分析を実行する。その主たる動作は、以下のようなものである。

- (1) 外部から与えられるデータを分析し、その結果を内部のデータベースに格納する。
- (2) 外部からのメソッド呼び出しに応じて、内部のデータベースを参照し、分析処理を行って返す。ここで外部とは、別の SaPO も含む。分析プログラムの配置、マイグレーションなどの本来複雑な処理を、SaPO 内部で自動化することで抽象化する。また、他の SaPO と協調して動作するため、自らのデータベースや分析プログラムについて、その構造やプロファイルを取得するリフレクション用メソッドを提供する。

上記の特徴を持つ SaPO を単位としてアプリケーションを構築することで、前述の課題を解決しつつ、様々な分析プログラムを組み合わせることができる。例えば、DISAANA を SaPO を用いて構築する場合、DISAANA 全体に対応する SaPO は、常時流入する Twitter 情報をデータベースに保持する SaPO と、ユーザからの要求に従ってデータベースから質問への回答を取り出す SaPO から構成する。また、この 2 種類

の SaPO はそれぞれ、下位モジュールとして多数の SaPO を持つ。上位の SaPO は、自身はデータベースを持たず、外部から Twitter 情報や質問の回答となるデータを与えられると、そのデータを分割し、下位の SaPO のデータベースに格納する。上位の SaPO に対する分析メソッドの呼び出しは、実際には下位の SaPO に委譲され、下位の SaPO で分析メソッドが並列実行される。下位の SaPO による分析メソッド実行の結果は、上位の SaPO によって集約され、呼び出し元に返される。

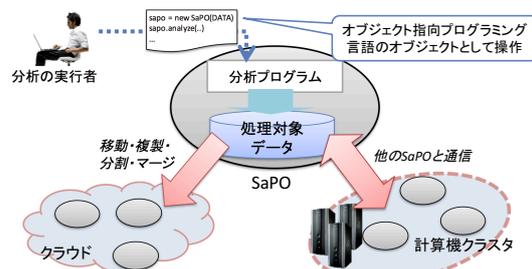


図 1 SaPO の概要

SaPO の要素技術の一部は、著者らが開発したミドルウェア RaSC^{*2)} によって既に実現されている。今後、SaPO が配置された計算機環境に合わせて、並列実行数や、データをオンメモリデータベースに格納するかどうかなどを自動で決定する自己適応機能、自らが持つデータの生成過程を記録し、分析プログラムのアップデート時に必要に応じてデータを再生成する機構、分析処理の柔軟な中断・再開をサポートする機構などを、SaPO の基本機能として実現していく。

4. おわりに

本稿では、大規模データ分析の課題と、その解決のための SaPO と呼ぶオブジェクトの構想を紹介した。今後、WISDOM X や DISAANA 等の開発と実運用を通じて、大規模データ分析のための技術を、SaPO ベースの新たな基盤ソフトウェアとして集約していく。

参考文献

- 1) Atkinson, M.: Persistence and Java - A Balancing Act, *Conference on Objects and Databases*, pp.1-31 (2001).
- 2) Jiang, D., Chen, G., Ooi, B.C., Tan, K.-L. and Wu, S.: epiC: An Extensible and Scalable System for Processing Big Data, *VLDB Endowment*, Vol.7, No.7, pp.541-552 (2014).

*2 <https://alaginrc.nict.go.jp/rasc/>