

DAQ-Middleware におけるコンポーネント間通信の高効率化に関する研究

Research on a High Speed Communication Method Between Components for DAQ-Middleware

前田 寛幸[†]

Hiroyuki Maeda

長坂康史[†]

Yasushi Nagasaka

1. はじめに

近年、ネットワーク技術の発達に伴い、ネットワークを介して様々なデータがやり取りされるようになった。物理学実験に対するデータ収集 (DAQ) システムにおいても、これらのネットワーク技術を利用することで、大容量の測定データを高速に収集し、保存することが可能となってきた。しかし、同規模の実験であっても、それぞれの実験用にデータ収集システムを開発することが多く、開発者の負担も大きくなってきている。

本研究室では、これらの負担を軽減する目的で、データ収集システムの汎用フレームワークとして、オブジェクトベースの DAQ-Middleware を他研究機関と共同で開発している。しかし、このフレームワークはデータ収集部分の通信にオブジェクト通信を利用しているため、通信効率が悪いという課題がある。

そこで本研究では、この DAQ-Middleware における通信を高速化することによって、システム全体の効率化を目的とする。

2. DAQ-Middleware

2.1 概要

DAQ-Middleware とは、ネットワーク分散環境でデータ収集ソフトウェアを容易に構築するためのソフトウェア・フレームワークである^{[1]-[4]}。

この DAQ-Middleware は、RT (Robot Technology) ミドルウェアをベースにしており、データ通信を行うそれぞれのコンポーネント間では、オブジェクト通信を行っている。また、このフレームワークでは様々なコンポーネントを組み合わせることができ、ネットワーク分散コンポーネント技術により、ユーザが求めるデータ収集システムを容易に開発することが可能となる。

図 1 に DAQ-Middleware の基本構成を示す。図 1 の PC はそれぞれ、ユーザインターフェイス用計算機 (UI) と DAQ 用計算機 (DAQ#1) を示している。ユーザは DAQ コンポーネントを制御する PC (UI) にある DAQ オペレータへ制御コマンドを送ることで全体のシステムの制御を行う。図 1 の PC (DAQ#1) では、基本 DAQ コンポーネントと呼ばれる Gatherer、Dispatcher、Logger、そして、Monitor が DAQ オペレータからのコマンドにより動作する。また、DAQ コンポーネント間の通信はネットワーク透過であるため、1 台の計算機上でこの 4 つのコンポーネントと DAQ オペレータを動作させることも可能である。

2.2 通信方式

DAQ-Middleware では、データ転送の際にオブジェクト通信を行うため、CORBA (Common Object Request Broker

Architecture) を用いている。そのため、仕様の異なる様々なコンピュータやプログラミング言語で利用することができるという利点がある。また、プラットフォームに依存しない分散処理環境の構築が可能である。

しかし、この CORBA 通信は複雑さゆえに、一般的な Socket 通信よりもオーバーヘッドが大きい。そのため、システムの負荷に繋がり、処理時間増加の原因となる。よって Socket 通信よりも通信効率が悪くなるという問題がある^[5]。

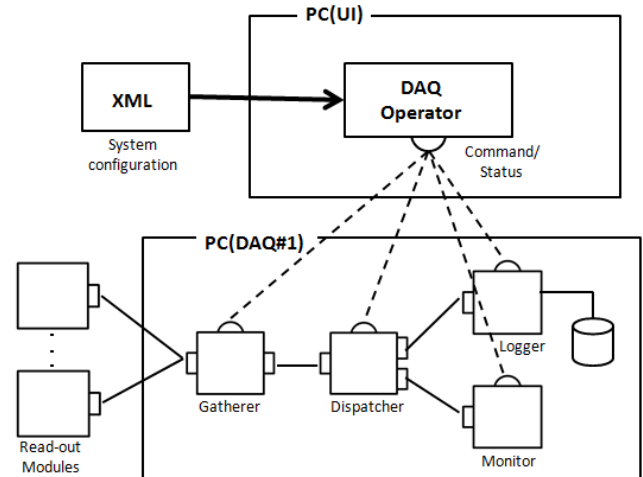


図 1 DAQ-Middleware の基本構成

3. 提案方式

3.1 概要

DAQ-Middleware では、CORBA を用いてデータ転送を行っているため、データサイズによってはオーバーヘッドが増大し、通信効率が悪くなる。そこで、コンポーネント間の通信効率を高めるため、CORBA での通信とともに、Socket 通信も選択できる機構を開発する。また、システム内の各コンポーネント間通信は、それぞれの特徴に合わせて、通信方式を選択できるようにする。

3.2 実装

DAQ-Middleware のコンポーネントである DAQ コンポーネントの概略を図 2 に示す。図に示すとおり、DAQ コンポーネントは、データ入力用ポートである InPort、出力用ポートである OutPort、データ処理を行う Logic、そして、コンポーネントに対する制御情報を受け取る Service Port で構成される。DAQ システムは、このコンポーネントを組み合わせることで構築することができる。DAQ コンポーネントの特徴は、任意の数のデータ入力ポートと出力

[†] 広島工業大学大学院工学系研究科

ポートを設定できる点である。また、図に示すサービスポートから DAQ オペレータの命令を受け取り、制御を行うことができる点である。新たな DAQ コンポーネントの開発は、コンポーネントのコアロジックと呼ばれる部分に機能を実装することで可能となる。

本研究では、DAQ コンポーネントのコアロジックに Socket 通信を行う機構を追加する。その際に、Socket 通信を行うためのクラスを作成し、既存のクラスにそれを継承させることで新たな通信方式を実現する。また、Socket 通信機能をコアロジックに実装することで、従来の CORBA と干渉しない機構を開発することができる。さらに、これらの通信方式を切り替えることができるようにした。この切り替えには、コンフィグレーションファイルを利用する。このファイルには、使用する DAQ コンポーネントの名前やその IP アドレス、コンポーネント間の接続情報など DAQ システムの動作に必要な設定が XML 形式で記述されており、DAQ オペレータがこの設定ファイルを読み込むことで、DAQ システムを動作させることができる。

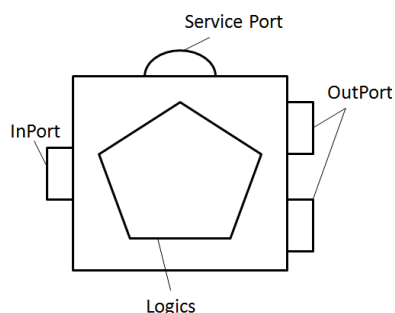


図2 DAQ コンポーネント

4. 性能評価

4.1 セットアップ

提案システムの性能を評価するため、CORBA と Socket 両方の通信速度を測定した。測定に用いた 2 台の PC は同じハードウェアで構成され、クロスケーブルによって直接接続した。表 1 に PC のスペックを示す。測定に使用した転送ファイルは、2 bytes～1 Mbyte のバイナリファイルである。測定結果は、CORBA と Socket 共に 100 回ずつ測定し、その平均としてまとめた。

表 1 PC スペック

OS	Scientific Linux 6.2
CPU	Intel(R) Xeon(R) CPU E5606 2.13 GHz
Memory	2048 MBytes
NIC	Intel Corporation 82598EB 10-Gigabit AT2 Server Adapter

4.2 結果と考察

測定結果を図 3 に示す。この図は、通信データサイズにおける転送速度の変化を表している。この図からわか

るように、データサイズが 256 bytes 以下のときは、通信方式の違いによる差はほとんど見られない。しかし、512 bytes 以上のデータ通信に対しては、Socket 通信と CORBA の転送速度の差が大きくなっている。両者の転送速度の差は、転送データサイズが 1024 bytes のとき最大で約 3.5 倍、また、全体平均で約 2.5 倍となった。

このことから、DAQ-Middleware において、通信方式を CORBA から Socket 通信に変更することでより効率的な通信を行えることが確認された。

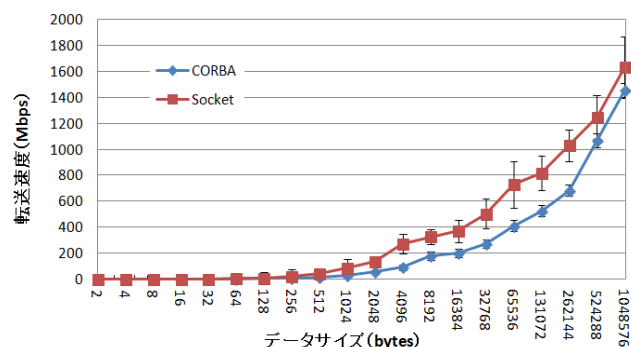


図3 CORBA と Socket の通信速度比較

5. おわりに

本研究では、DAQ-Middleware におけるコンポーネント間通信の高効率化を目的とし、Socket 通信の実装を行った。また、CORBA での通信とともに、Socket 通信も選択できる機構を開発した。これにより、各 DAQ コンポーネント間の通信は、それぞれの特徴に合わせて、通信方式を選択できるようになった。

また、通信効率を明らかにするために、CORBA と Socket 通信のデータサイズにおける転送速度の変化を測定した。測定結果から、Socket 通信は、CORBA よりも転送速度が平均して約 2.5 倍効率が良いことを示した。

今後は、通信方式の切り替えを自動で行うことで、ユーザに意識させずに通信効率を高める方法について研究を行う。

参考文献

- [1] Y. Yasu, K. Nakayoshi, E. Inoue, H. Sendai, H. Fujii, N. Ando, T. Kotoku, et al., "A Data Acquisition Middleware," in *Proc. IEEE/NPSS Real Time Conference*, pp. 1-3, May 2007.
- [2] Y. Yasu, K. Nakayoshi, H. Sendai, and E. Inoue, "Functionality of DAQ-Middleware," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 2, pp. 487-490, Apr. 2010.
- [3] K. Nakayoshi, H. Sendai, Y. Yasu, E. Inoue, T. Kotoku, N. Ando, Y. Nagasaka, S. Ajimura and M. Wada, "DAQ-Middleware: Progress and status," *J. Phys.: Conf.*, Ser. 331 02202318, 2011
- [4] H. Sendai, K. Nakayoshi, Y. Yasu and E. Inoue, "Performance measurement of DAQ-Middleware," *J. Phys.: Conf. Ser.* 331 022039, 2011
- [5] N. Ando, T. Suehiro, K. Kitagaki, T. Kotoku and W. Yoon, "RT-middleware: distributed component middleware for RT (robot technology)," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3933-3938, Aug. 2005.