2J-07 高信頼かつ高性能な非同期メッセージキューの実現手法

田中 俊介　小山 貴夫　日下 貴義　吉谷 文徳　松田 栄之
株式会社NTTデータ 情報科学研究所
e-mail: {shun, kym, kusaka, yositani, matu}@rd.nttdata.co.jp

1. はじめに

OLTP(OnLine Transaction Processing)システムではACID性質(原子性・一貫性・隔離性・耐久性)を満たす必要がある.TPモニタもしくは非同期メッセージキュー(以下キューと略す)はACID性質の一部を提供するミドルウェアである.TPモニタは同期型であるため，サーバ停止時にはクライアントでのメッセージ送信要求は異常終了する．キューは非同期型であるため，サーバ停止時にもメッセージ送信要求を実行可能である．メッセージはサーバが再起動した時点で転送される[1].

キューでは転送するメッセージをログとしてディスクに記録するため，ディスクI/Oがボトルネックとなり，スループットが上がらないケースが多い.

本報告では，高信頼かつ高性能なキューの実現方式を提案する．また，本方式と従来方式の性能比較結果について報告する.

2. 従来のキュー

2.1. 従来のキューの処理方式

2台のサーバ間の通信を，従来のキュー[i)]を利用して行う場合を図1に示す.

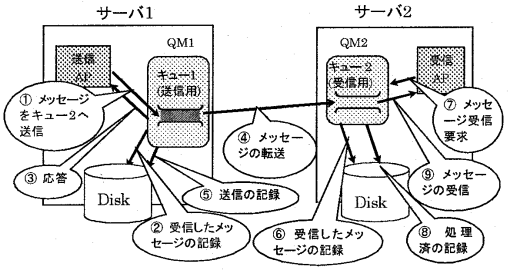

図1 従来のキューでの処理

始めに，送信APが同一サーバ内のQM(キューマネージャの略)にメッセージの送信要求を行う．送信要求を受けたQM1は，受信したメッセージをログに記録し，送信APに応答を返す．その後，QM2にメッセージを転送する．メッセージを受信したQM2はログ記録を行う．受信APが受信要求を行うと，QM2がメッセージを渡し，ログに処理終了を記録する.

2.2. 従来のキューの問題点

最新鋭のサーバ機では，CPUの速度に比べるとディスクI/Oが遅いため，OLTPシステムではディスクI/Oがボトルネックになる場合が多い.

従来のキューでは，1つのメッセージを送信側QMと受信側QMの両方でログ記録を行う.ディスクI/Oが多く発生するため，スループットが上がらないケースが多い.

3. 新しいキュー実現手法の提案

3.1. 目的

キューにおいて，トランザクションの耐久性を維持しながら処理性能(メッセージ転送のスループット)を高めることを目的とする.QMのメッセージ転送処理に関してログ記録の回数を減らすことで処理性能向上を目指す.

3.2. 処理方式

本報告のキューでは，1つのメッセージは，送信側QMもしくは受信側QMのどちらか一方のみでログの記録を行う.ログの記録は2つのQMで交互に行う.送信APからのメッセージ転送要求に対して，1つ目のメッセージは送信側QMのみでログ記録を行い，2つ目のメッセージは受信側のQMのみでログ記録を行う.3つ目のメッセージは再び送信側のQMのみでログ記録を行い，以後、送信側QM，受信側QMで交互にログ記録を行う(図2参照).

2種類のメッセージ転送処理の各々について図2に示す.

3.3. 信頼性(障害時の処理)

信頼性に関しては以下のようなアプローチを取る.

受信側サーバに障害が発生した場合の処理を図3に示す.QM2でログ記録を行う処理(図2の偶数番目メッセージの処理)では，QM1はQM2でのログ記録が完

A Design of a Reliable and Scalable Asynchronous Message Queue
Shunsuke Tanaka, Takao Koyama, Takayoshi Kusaka, Fuminori Yoshitani, Shigeyuki Matsuda
Laboratory for Information Technology,
NTT DATA CORPORATION

i)従来のキューの処理方式は，代表的なキューの実装であるIBM社のMQSeries®[2]の処理方式に基づいて記述した.
MQSeries®はIBM社の商標.

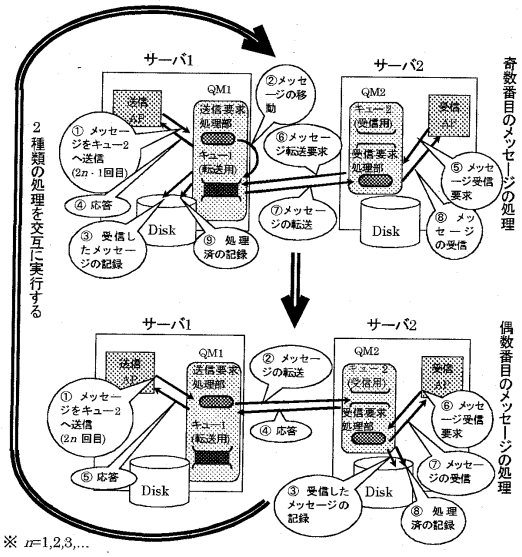

図2 本報告のキューの処理方式

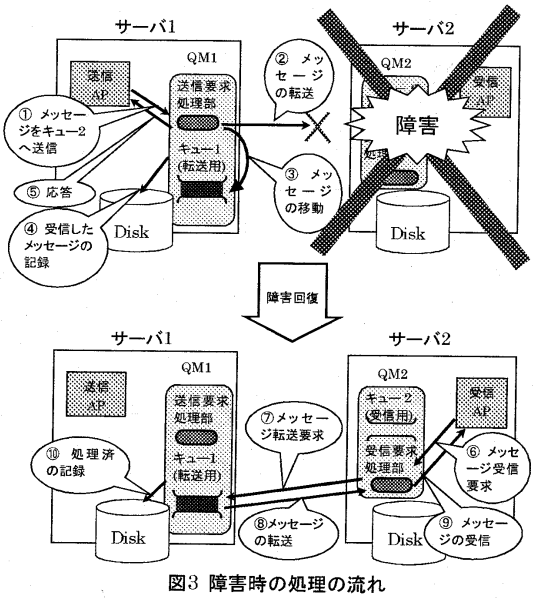

図3 障害時の処理の流れ

了したことを確認してから送信 AP に応答を返す．サーバ 2 に障害が発生した場合，QM1 から QM2 へのメッセージ転送は失敗し，QM1 はタイムアウトによって失敗を検知する．そのとき，QM1 は自サーバのディスクにログ記録を行い，ログ記録が完了してから送信 AP に応答を返す．以上のように，受信側サーバ障害時には全てのメッセージを送信側 QM でログに記録するため，本方式ではトランザクションの耐久性を実現可能である．

送信側サーバに障害が発生した場合に関しては，従来のキューと同様に，特別な対策は必要でない．これは送信 AP がメッセージ転送要求を行えないことから，QM で処理を継続する必要がないためである．

4. 評価

従来のキューと同等の処理方式を持つ QM と本方式の処理方式を持つ QM のプロトタイプを実装した．2 種類の QM のそれぞれに対して，2 台のサーバ[ii]間でメッセージ転送を行い，送信 AP ・受信 AP の数,メッセージサイズを変化させた場合のスループットを測定した．

測定した結果を図 4 に示す．図 4 から，本方式の方が従来方式よりも高いスループットが得られることが確認できた．特に，AP 多重度 2 以上の場合にその差が顕著に現れるという結果が得られた．

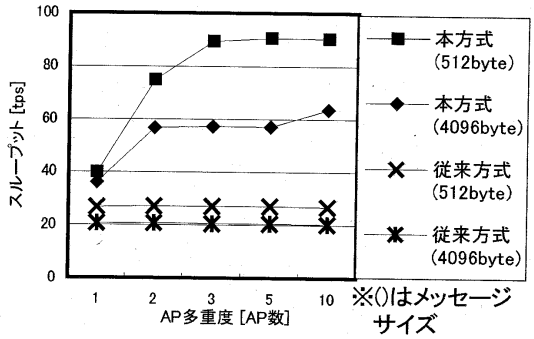

図4 性能比較の実験結果

5. おわりに

従来の高信頼非同期メッセージキューでは，全てのメッセージを送信側マシンと受信側マシンの両方でログに記録するため，ディスク I/O が多く発生し，スループットが上がらないという問題がある．本報告では，各メッセージを送信側もしくは受信側のどちらか片方のみでログ記録を行う処理方式を提案した．また，提案した方式に基づいてプロトタイプを実装し，従来の処理方式との性能比較から，本方式の優位性を確認した．また，本方式が性能の確保と共に十分な信頼性が実現できることを示した．

【参考文献】

[1] Philip B. and Eric N., トランザクション処理システム入門, 日経 BP 社, Mar. 1998

[2] MQ Series® Planning Guide, IBM, Apr. 1999

ii) 実験は以下のマシンを 2 台使用して実施した．

Sun Ultra1, CPU:UltraSparc167MHz×1, Memory:256Mbyte, Disk:FastSCSI 内蔵 Disk 2.1G×1, Ethernet:100Base-T, OS:Solaris 2.6, コンパイラ:gcc 2.8.1