

メディア処理向けクラウド基盤「虹雲」

赤間 浩樹 松田 基弘 毛受 崇 長谷川 知洋 内藤 一兵衛 山室 雅司
 日本電信電話 株式会社 NTT サイバースペース研究所

1 はじめに

我々は情報統合管理の更なる効率化を目指し、研究開発を進めている。

1.1 データベースとデータストリームの統合

追記・参照型 DMS^[1]は、分散データストリーム管理システム(DSMS)をデータベース管理システム(DBMS)と統合した分散システム向けミドルウェアである。図1に示すようにデータの流れの方向への機能分散(追記部,フィルタ部,ビュー部)と,処理量増加の方向への分散(Scale-out)の2軸の分散構成を採り,万能なフィルタ部が自律的に追記部からデータを取得して処理を行うことによって,システムの無停止での進化(規模成長と機能成長)を可能にする。

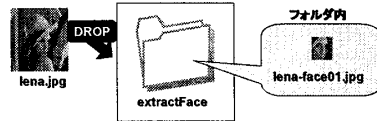


図2 フォルダ名に応じた処理の実行例

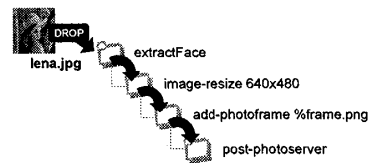


図3 フォルダ階層に従った処理連鎖の例

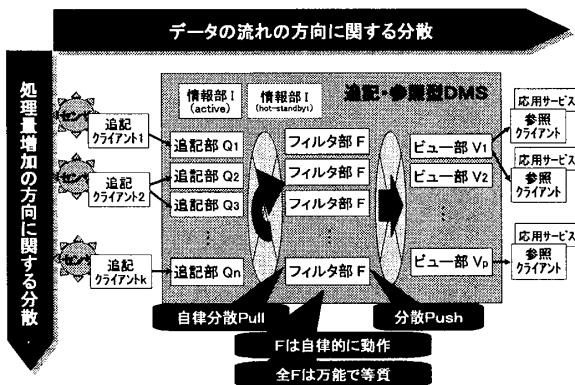


図1 DMS アーキテクチャ

	UNIX pipe	POLDER
1 ユーザ・インタフェース	・GUIで専門化向き。△	・GUI(デスクトップと統合)で初心者にも容易。○
2 データフロー制御	・基本は一直線。○	・Tree状にデータフローが分岐することが可能。◎
3 大量ファイルへの手軽な適用	・shスクリプトのfor活用。△	・複数のファイルを選んでそのままDROPすれば全てに適用。◎
4 デバッグ	・shスクリプトの編集により実施。△	・処理連鎖での途中での停止、途中からの開始が容易。○
5 プログラムの保存・再利用	・shスクリプト化して保存。×	・フォルダとしてそのまま。何もなくても保存されているし、いつでも再利用可能。◎

図4 UNIX pipe との比較

1.2 メディア処理とメディア管理の統合

フォルダ・プログラミング環境 POLDER^[2]は、フォルダ・メタファを使ったメディア処理部品群に対する Web マッシュアップ環境である。各種のメディア処理部品群を既存のフォルダによるメディアデータの管理に統合する。「フォルダの外側からデータが投入されてフォルダ名に応じた処理が実行され(図2), その結果がフォルダの中に生成され,それが下側のフォルダに連鎖していく(図3)」という基本原則を使ってプログラミング初心者にも簡単な処理構成・実行環境を提供する。POLDER は,データフロー処理の視覚化環境と捉えることもできる。図4に UNIX の pipe との比較を示す。

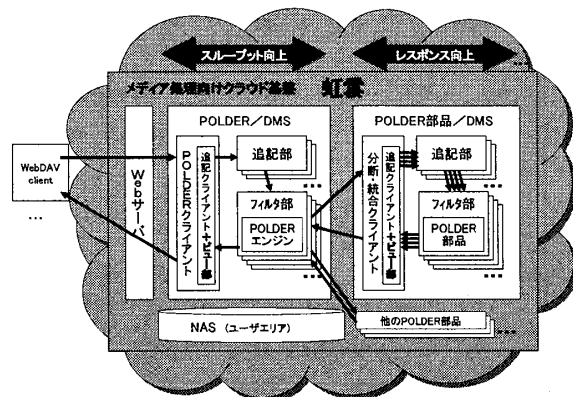


図5 メディア処理向けクラウド基盤「虹雲」

2 メディア処理向けクラウド基盤「虹雲」

POLDER は,追記・参照型 DMS(以下 DMS と省

NIJIGUMO: a cloud for multimedia contents processing
 Hiroki AKAMA, Motoki MATSUDA, Takashi MENJO,
 Tomohiro HASEGAWA, Ichibe NAITO, Masashi YAMAMURO
 NTT Cyber Space Labs.

略)の一応用サービスとして実装される。これを POLDER/DMS と呼ぶ。さらに POLDER から実行される各 POLDER 処理部品群の一部も DMS を使って実装される。これら POLDER と複数の DMS から構成されるメディア向けクラウド基盤を「虹雲」(図5)と呼ぶ。虹雲はクラウドの分散処理のパワーを多数の利用者に簡単なインタフェースで提供する。

2.1 POLDER/DMS でのスループットの向上

利用者の WebDAV クライアントから PUT されたファイルは Web(WebDAV)サーバを経由して POLDER/DMS が受け付ける。POLDER クライアントは追記クライアントとビュー部を兼ねている。PUT の情報は追記部を経由してフィルタ部の POLDER エンジンに渡され、POLDER 実行完了後にビュー部を経由して利用者クライアントに通知される。投入データや処理中データ、処理結果データなど、各種ファイルは系内の NAS の各ユーザ域に存在する。DMS は POLDER サービスに対して複数のサーバを活用した同時実行性を高め、スループットを向上させる役割をもつ。

2.2 POLDER 部品/DMS でのレスポンス向上

POLDER エンジンからフォルダ名に応じて呼び出し実行される各種部品を POLDER 部品と呼ぶ。POLDER 部品には多様なものが存在するが、巨大なコンテンツに対する時間がかかる処理に対して DMS の分散処理を使って応答時間の高速化をはかることができるケースが存在する。その場合には、POLDER エンジンから呼び出される部品の最初でコンテンツの分断を行い、そこで生成された断片データに対して POLDER 部品の並列実行を行い、再び結果の統合を行って、その結果を POLDER エンジンに返却する。この場合には、分断数に応じたレスポンス時間の向上が期待できる。

3 評価

前述の効果を確認する実験を行った。

【実験1】 画像サイズ変更処理 (image-resize 2048x2048)を複数の利用者が同時に実行する場合もしくは1人の利用者が複数のファイル(100 個)を一気に DRAG&DROP した場合における POLDER/DMS によるスケラビリティを図6に示す。台数に応じて全体完了までの処理時間が短縮し、スループットが向上している。

【実験2-1】 大きなコンテンツ (mpeg2 映像約 10 分)に対する重要シーン検出処理^[3]における処理時間を図7に示す。DMS を使ったことで 18 倍程度レスポンスが高速化している。

【実験2-2】 大きなコンテンツ (テキスト 7MB 分)に対して形態素解析および構文解析処理を行う処理時間を図8に示す。DMS を使ったことで 16 倍程度レスポンスが高速化している。

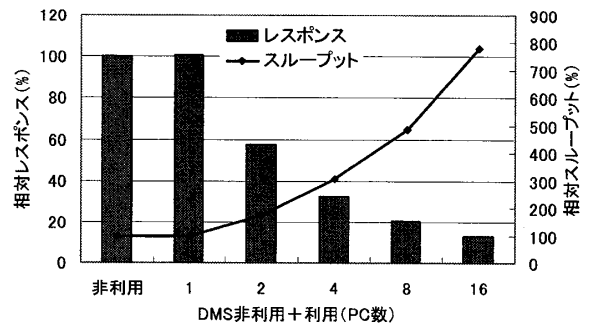


図6 スループット向上

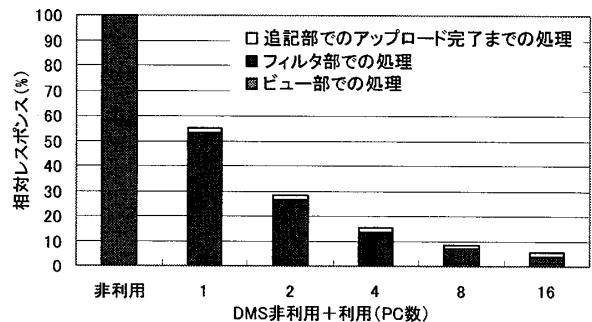


図7 レスポンス向上(映像処理)

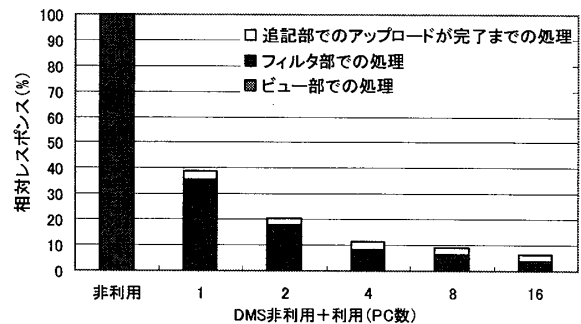


図8 レスポンス向上(テキスト処理)

4 まとめ

POLDER サービスにおいて、追記・参照型 DMS はスループットの向上およびレスポンスの向上の双方に寄与する。レスポンスが向上するケースは、分断と統合にかかる時間が相対的に短いケースでなければならない。POLDER/DMS 用の DMS と POLDER 部品/DMS 用の各 DMS は同一の DMS を使った異なるサービスとして存在可能で、マシンリソースの有効な活用も可能である。

参考文献

- [1] 赤間 他: 追記・参照型データ管理システムの設計と評価, 情報処理学会論文誌, Vol.49, No.2, pp.749-764 (2008).
- [2] 赤間 他: フォルダ・プログラミング環境「POLDER」, 情報処理学会論文誌, Vol.50, No.8, pp.1798-1809 (2009).
- [3] 鳥井 他: 動きを用いた動物体アップショット検出, 信学会総合大会論文集 2005 情報・システム(2), pp.172 (2005).