

Keio-MMP におけるパフォーマ構築キットへの展望†

1N-4

斎藤 尚則*‡ 西尾 信彦** 萩野 達也**

*(株)富士通ソーシャルサイエンスラボラトリー **慶應義塾大学環境情報学部

1 はじめに

筆者らは現在、分散マルチメディア環境 Keio-MMP のコンダクタ/パフォーマ・アーキテクチャ[1]上で動作する実験的な JPEG 伸長パフォーマ [2] の実装を進めている。

コンダクタ/パフォーマ・アーキテクチャは、Real-Time Mach 3.0 マイクロカーネル [3] 上の複数のサーバ群により連続メディア処理を実行する機構であり、処理のタイミングと資源管理を統括するコンダクタの下で、各種メディアに特化した複数のパフォーマが協調的に動作する。メディアデータに対する処理は複数のパフォーマによってパイプライン的に実行される。このメディアデータの流れをストリームと呼び、ストリームの中間部においてメディアデータの加工を担当するパフォーマをフィルタパフォーマと呼んでいる。

各種のメディアに特化した複数のサーバとして実現することにより [4]、コンダクタ/パフォーマ・アーキテクチャ上では、連続メディア処理システム構築の際に高い柔軟性および拡張性が期待される。筆者らの実装している JPEG パフォーマはフィルタパフォーマの典型的な一例であり、パフォーマ構築の際に参考となるいくつかの特徴がみられる。本稿では、フィルタパフォーマのインタフェースおよび実現例について報告し、あわせて将来におけるパフォーマ構築キットへの展望について述べる。

2 フィルタパフォーマの実現

ストリーム中のフィルタとして機能するパフォーマの実装について、JPEG 伸長パフォーマの例にもとづいて報告する。

2.1 同期ストリームと非同期ストリーム

パフォーマの連結により形成されるストリームは、処理の開始/終了のタイミングをコンダクタが支配する同期ストリームと、コンダクタの管理する時間制約と無関係に動作する非同期ストリームに分類される。非同期ストリームは先行制御やジッタの吸収を目的としている。現時点では同期ストリームについてのみ実装している。

ストリーム上を流れるメディアデータは MDU(Media Data Unit) と呼ばれるデータ単位に分割して取り扱われている。MDU は当該メディアデータを構成する要素として意味をもって扱うことのできる最小単位である。ス

トリームを構成する各パフォーマは MDU を単位として処理をおこなう。

2.2 JPEG 伸長パフォーマの実装例

フィルタパフォーマの一例として、ソフトウェアによる JPEG 画像の伸長をおこなうパフォーマの実装をおこなった JPEG の伸長処理をおこなうエンジンとしては Independent JPEG Group による libjpeg(Release 4A)を採用し、以下に示すパフォーマインタフェースを付加した。

標準の libjpeg では伸長処理に必要となる全情報を decompress_info_ptr 型の構造体で管理している。伸長処理の副作用により同構造体の内容が破壊され、リエントラントな伸長ルーチンとして利用することができないため、伸長ルーチンの起動時に構造体の内容を毎回デフォルト値で初期化している。

JPEG パフォーマは以下のパフォーマインタフェースをコンダクタに提供する。このパフォーマインタフェースは他のパフォーマにおいても利用可能である。

● open_connection()

コンダクタからのセッション開始要求を受けつける。JPEG パフォーマはパフォーマの初期化をおこない作業領域となるメモリオブジェクトを確保する。

● set_value()

パフォーマのステータスを設定する。またはデフォルト設定値を変更する。

● get_value()

パフォーマのステータスを得る。

● process_and_map()

メディアデータの加工をおこない、Mach の vm_map システムコールを用いて処理済みのメディアデータを下流のパフォーマのアドレス空間にマップする。

● close_connection()

パフォーマの終了処理をおこなう。未実装

メディアデータ処理の同期は、適切なタイミングでコンダクタが process_and_map() を呼ぶことによって実現されている。

JPEG パフォーマでは、set_value() によって、JPEG 伸長時のパラメータとなる量子化テーブルおよびハフマンテーブルの内容を書き換えることができる。get_value() はパフォーマのデフォルト設定値および set_value() による設定値を参照する手段を提供するが、現時点では実装されていない。

現在の実装では、JPEG パフォーマの取り扱うメディアデータは JFIF(JPEG File Interchange Format) 形式のイメージに限られている。Parallax XVideo の生成する JPEG Movie フォーマットを理解するファイルサービスパフォーマ(実装中)と組み合わせて使用するため、

† "Toward a performer construction kit in Keio-MMP" †

SAITOH Takanori*, NISHIO Nobuhiko**, HAGINO Tatsuya**

*Fujitsu Social Science Laboratory, Ltd., 8 Higashida-cho, Kawasaki-ku, Kawasaki-shi, Kanagawa, 210 Japan, **Keio University, 5322, Endo, Fujisawa-shi, Kanagawa, 252 Japan

† この研究は、情報処理振興事業協会 (IPA) が実施している開放型基盤ソフトウェア研究開発評価事業「マルチメディア統合環境基盤ソフトウェア」プロジェクトのもとに行なわれた。

‡ 開放型基盤ソフトウェア湘南藤沢キャンパス研究室の研究員として IPA に登録されている

set_value() に対応するパフォーマの内部処理に修正をくわえている。

2.3 CSB によるパフォーマ間 IPC

パフォーマ間における MDU の転送を上述のように Mach の `vm_map` システムコールで実行する (同期ストリーム) ことはコストがかかり、また処理の並列性を高めることが困難であるため、CSB (Cyclic Shared Buffer) と呼ばれる共有メモリ [1] によるパフォーマ間 IPC が提案されている。複数のサーバの協調的動作によって連続メディア処理をおこなう場合、サーバ間通信のコストにより全体のパフォーマンスが低下するため、データのコピーをとらなわなないサーバ間通信の手段として CSB が提案された。

各フィルタパフォーマは、ストリーム中で隣接する他のパフォーマとの間で CSB を共有している。CSB がパフォーマに提供している機能は MDU の入出力の手段のみであるため、CSB を介したメディアデータの転送においては、パイプライン上の同期処理を保証するための以下のプロトコルをとる。

- `send_csb_for_proc()`
上流側パフォーマが処理を完了し、一個の MDU を下流側パフォーマに CSB 経由で送信する
- `receive_csb_for_proc()`
下流側パフォーマが CSB 経由で上流からの MDU を受信する
- `send_csb_released()`
下流側パフォーマが CSB 上の一個 MDU に対して処理を完了し、上流側パフォーマがそれを再利用することが可能となったことを通知する
- `receive_csb_released()`
上流側パフォーマが解放されたフリーの MDU を受けとる

通信は Mach の port 経由でなされ、送信はすべて非同期におこなわれる。フィルタパフォーマは固有のメディア処理を実行するためのエントリを呼びだし、MDU に対する加工をおこなう。処理されたデータは下流側の CSB に置かれ、次のステージに送られる。

3 パフォーマ構築キットへの展望

パフォーマの構築に際して簡便なプログラミングインタフェイスを提供するためのパフォーマ構築キットの作成を検討している。パフォーマ構築キットの実現については:

- 各パフォーマに固有のメディア処理に依存しないコードを抽出し、「空の」パフォーマのテンプレートを作成する
- CSB によるパフォーマ間通信インタフェイスをライブラリ化する

などを検討している。構築キットには入らないフィルタ処理以外の、各種パフォーマ固有の処理について以下に述べる。

3.1 QOS 制御の手段

各パフォーマが QOS 制御をおこなう方法は、それぞれが取り扱うメディアデータの特性に強く依存している。

また各メディアデータに対してメディアスケーリングをおこなう場合にも適用の方法は様ではない [5]。

このため各パフォーマには、コンダクタに対し当該パフォーマの提供する QOS 制御メソッドの次元およびメディアスケーリングの粒度を通知するメソッドが必要となる。

3.2 メディアデータに関する付加的情報の扱い

各パフォーマには、メディアデータのフォーマットやメディア処理時のパラメータなどの通知を受ける方法が用意されていなければならない。これらの付加的情報をを用いた動的 QOS 制御の一例として、JPEG 画像圧縮の量子化テーブルを積極的に変更することによりストリーム中をながれるデータ流量を一定に保つ方法が提案されている [6] が、このような場合には付加的情報が任意の時点に変更されることを考慮しなければならない。

4 今後の課題

コンダクタおよびストリーム中で隣接するパフォーマと通信をおこなう null パフォーマのテンプレートを作成し、パフォーマ構築キットのベースとする。

また本稿においては、ストリームの起点および終点に位置するソース/シンクパフォーマの実現については未着手となっているが、Keio-MMP 上ではウィンドウシステムパフォーマ [7] や音声入出力パフォーマ [8] などのソース/シンクパフォーマの実装をおこなっており、今後はフィルタパフォーマと統合されたインタフェイスによる構築の可能性を含めて検討していきたい。

謝辞

本研究をおこなうにあたり御協力いただいた開放型基盤ソフトウェア研究開発評価事業「分散マルチメディア統合環境 Keio-MMP」プロジェクトの皆様に感謝いたします。

参考文献

- [1] 西尾: “Real-Time Mach 3.0 上の連続メディア処理のための協調サーバ群の設計と実装,” 情報研報, Vol.94, No.32, 94-OS-63, pp. 57-64 (1994)
- [2] 斎藤, 萩野: “分散マルチメディア統合環境 Keio-MMP における JPEG パフォーマの設計,” 第 49 回情処全大, pp. 2-425-426
- [3] H. Tokuda, T. Nakajima and P. Rao: “Real-Time Mach: Towards a Predictable Real-Time System,” USENIX Mach Workshop, pp.73-82 (1990).
- [4] 平林, 他: “実時間メディアサーバの設計,” 第 5 回コンピュータシステムシンポジウム論文集, pp. 25-32 (1993)
- [5] 中島, 手塚: “連続メディア処理のための動的 QOS 管理方式,” 第 49 回情処全大, pp. 3-315-316 (1994)
- [6] 藤井, 斎藤: “任意ビットレート下における動画像の動的 QOS 制御方法の提案,” 第 48 回情処全大, (6C-8), 1994
- [7] 多田, 斎藤: “Keio-MMP における X ウィンドウシステムの実時間拡張方式,” 第 49 回情処全大, pp. 2-403-404 (1994)
- [8] 平林, 萩野: “Keio-MMP におけるサウンドパフォーマ,” 第 49 回情処全大, pp. 3-319-320 (1994)