

6R-3 Keio-MMP におけるプロトコルアーキテクチャ†

南部 明* 盛合 敏* 尾上 裕子* 木原 誠司* 光澤 敦* 徳田 英幸**

*NTT 情報通信研究所 **慶應義塾大学環境情報学部

1 はじめに

ビデオやオーディオストリームなどの時間的な制約を持った連続メディアストリームを扱い、分散マルチメディアシステムを構築するための通信環境が求められている。

著者らは分散マルチメディアシステムを実現する Keio-MMP プロジェクト [1] の統合プラットフォームにおいて利用されることを想定し、連続メディアストリームを扱うことができるプロトコルアーキテクチャについて研究している [2]。本稿では、このプロトコルアーキテクチャについて述べ、これらの Real-Time Mach[3] 上での実装について報告する。

2 連続メディアストリーム向きプロトコル

連続メディアストリームを扱う上で最も重要なことは、時間的に変化する情報を時間的な構造を保ったまま、可能な限り正確に伝達し、再現することにある。しかし、システムの資源は有限であるため、その再現品質は利用できる資源とのトレード・オフとなる。

連続メディア処理に適したネットワークシステムを実現するためには、物理ネットワークの高速広帯域化に加え、通信プロトコルにおいて時間制約および QOS の保証機能を備えることが必要である。時間制約保証とは、処理時間を一定の枠内に収めなければならないと言ったタイミングの正当性保証に加え、映像や音声など連続メディアオブジェクト間のタイミングの同期保証のことを言う。また、QOS 保証とは、アプリケーションの要求する品質をネットワークで保証可能な品質にマッピングし、CPU やメモリなどの資源と協調制御することである。

時間制約を保証するためには、プロトコルおよびレイヤ構成がレイヤ間の機能の重複を避けたオーバヘッドの少ないこと、遅延時間とそのゆらぎの上限が押さえられることが必要である。

また、QOS を保証するためには、プロトコルの各レイヤが QOS パラメータを持ち、レイヤ間協調による QOS 制御が可能であり、アプリケーションの品質をネットワークの品質にマッピングし、CPU やメモリなどの資源の予約が必要となる。資源の予約についてはコネクション型プロトコルのコネクションごとといったセッションを単位とする方法が一般的である。さらに、きめ細かい QOS

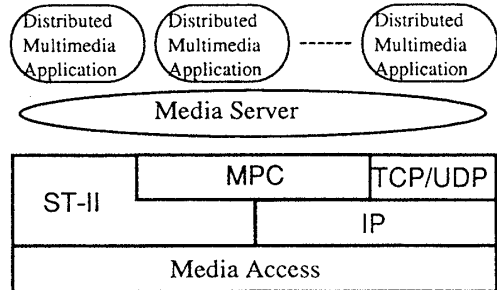


図 1: プロトコルの構成

制御を行う場合は、予約に依らず、連続メディア情報の中味に依存したフィルタリング処理などを、ネットワーク上の各種資源の状態に応じて動的に制御するメディアスケールリングがある。

予約をベースとした方式は、連続メディアストリームには一切手を加えないため、処理が単純となり、対話型アプリケーションなどのメディア処理に適している。一方、動的な制御方式は連続メディアストリームの中味に関与するため、品質制御ビットの挿入などデータの生成処理が複雑化するが、負荷の変動に強く、インターネットなどの異機種システムが混在するネットワーク環境へも適用できる。そこで、我々は連続メディアストリーム向きのプロトコルのアーキテクチャとして、予約をベースとしたプロトコルと動的な QOS 制御を可能とするプロトコルの併存モデルを採用している。図 1 に連続メディアストリーム向きのプロトコルの構成を示す。

トランスポートおよびネットワーク層に相当するプロトコルは、予約をベースとしたプロトコルとして ST-II[4] を採用している。更に動的な QOS 制御を可能とするマルチキャストプロトコルとして MPC[5] を新たに設計した。MPC にはエンドツーエンドの QOS 制御を行うためのエンドツーエンドでの FlowSpec と呼ぶ QOS パラメータのレベル設定とネゴシエーション機能、実際のデータフローのモニタリング機能がある。

連続メディアストリーム用のプロトコルアーキテクチャにおいて、静的な QOS 制御のみ行う場合、ST-II を上位から直接利用することができるようにする。また、動的な QOS 制御を行う場合、MPC と IP および IP Multicast を利用する。ここで IP を利用するのは、中継ルート上にルータなどのネットワーク機器に既存の物を入れることができ、既存インターネットを使ったマルチメディアシステムの構築が可能となるためである。

更に、ST-II の上位に MPC を利用することも可能である。この場合、QOS 制御の粒度を細かくすることが

† "A Protocol Architecture in the Keio-MMP Platform"
 Akira Nambu*, Satoshi Moriai*, Yuko Onoe*, Seiji Kihara*, Atsushi Mitsuzawa*, Hideyuki Tokuda**
 *NTT Information and Communication Systems Laboratories, 1-2356, Take, Yokosuka-shi, Kanagawa, 238-03 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」プロジェクトの研究協力として行なわれた。

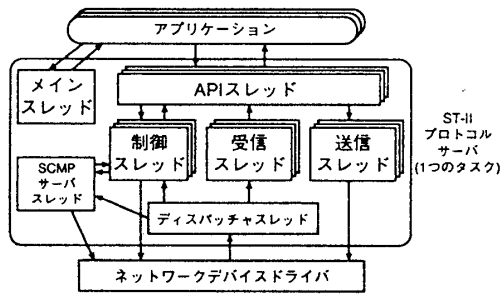


図 2: ST-II プロトコルサーバの構成

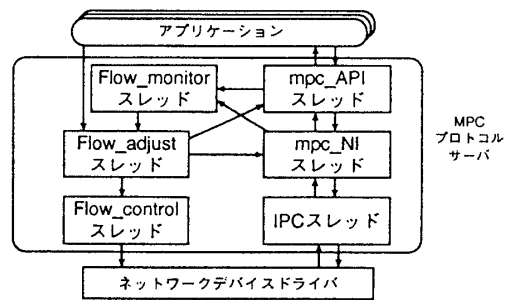


図 3: MPC プロトコルサーバの構成

き、マルチキャスト通信時の様に多種のノードが同時に受信する際の個々のノードの受信処理能力の違いを隠蔽し、様々なレベルのQOSを要求するターゲットをマルチキャスト通信のグループに許容することが可能になる。

3 プロトコルの実装

ST-IIとMPCを以下の理由によりReal-Time Machのユーザレベルサーバにより実装した。

- プロトコルの実験を目的としており、実装の容易性を優先したこと
- ST-IIの場合、アプリケーションのバグにより、予定していたより速いレートでデータの送出が行われると、予約より多くの資源を使用することになり、問題が起きること
- ST-IIの制御プロトコルであるST Control Message Protocol(SCMP)が複雑で状態を持つ必要があること
- データ送受信のレート制御、CPUの予約を行う必要があること

3.1 ST-IIプロトコルサーバの構成

ST-IIプロトコルサーバは、図2に示すように、7つのスレッド群によって構成される。

高速ネットワークにおいては、プロトコル処理のオーバーヘッドが全体の遅延時間に対して大きな割合を占める。そこで、ST-IIプロトコルサーバではプロトコル処理による遅延を小さくするため、プロトコル階層の間の複写を減らす工夫を行っている。サーバ内のスレッド間ではポインタ渡しによるデータの共有を、サーバとアプリケーションの間では仮想記憶ページを交換するシステムコールを用いることにより、パケットの複写を避け、プロトコル処理の高速化を図る。また、パケットの受け渡しには、リングバッファをもちいる。デッドラインを過ぎた、内容に意味がなくなったデータを次々に捨て、そのバッファ領域を再利用することにより、使用する記憶域の大きさが一定になるため、記憶域をコネクション設定時に予約することができる。

3.2 MPCプロトコルサーバの構成

MPCプロトコルサーバは、図3に示すように、6つのスレッド群によって構成される。

ST-IIの様な従来のプロトコルでは、マルチキャスト通

信を行うと、グループメンバ全員が一意のQOSとなる。一方、MPCではグループメンバやネットワークルートの能力に応じた相対的QOSが提供でき、ネットワークやCPUの資源状態に応じた動的なQOS制御を実現できる。

MPCにおけるQOS制御は連続メディアストリームのフロー制御により実現する。フロー制御はネットワーク層のパケットを対象にフィルタリングにより行う。これによりきめ細かく動的なQOS制御が可能となる。フロー制御の対象となるパケットにはQOSフィールドを付けている。このQOSフィールドは多次元的に表現することも可能であり、複数要因のQOS制御が可能となる。

4 おわりに

連続メディアストリームの転送が可能なプロトコルアーキテクチャについて示した。このプロトコルアーキテクチャは予約をベースとしたストリーム型ネットワークプロトコルであるST-IIと動的QOS制御を可能とするMPCを中心として構成される。

ST-IIとMPCはRT-Mach上のユーザレベルサーバで実装したが、今後、Real-Time Machの標準的なネットワークインタフェースを提供するNPS[6]にプロトコル処理部をポーティングし、他モジュールとの統合を図っていく予定である。

参考文献

- [1] 徳田, 萩野, 斎藤: 分散マルチメディア統合環境 Keio-MMP プロジェクトにおける連続メディア処理のためのソフトウェアアーキテクチャ, 第49回全国大会論文集, 7R-01 (1994).
- [2] 南部, 盛合, 尾上, 木原, 光澤: 連続メディアストリーム向きプロトコル, 信学技報, CS94-83 (1994).
- [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] Topolcic, C.: ST-II, in Proceedings of First International Workshop on Network and Operating Support for Digital Audio and Video, Springer-Verlag(1990).
- [5] 尾上: マルチキャストサーバにおける動的QOS制御, 情報研報(DPS), DPS66-1 (1994).
- [6] Nakajima, T. and Tokuda, H.: A User-Level Network System providing Predictable Communications, in 10th Conference Proceedings of Japan Society for Software Science and Technology (1993)