

Title	連続メディアデータ処理とそのアプリケーションの製作を支援するツールキットに関する研究
Author(s)	大平, 浩貴
Citation	
Issue Date	1997-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1048
Rights	
Description	Supervisor:中島 達夫, 情報科学研究科, 修士

連続メディアデータ処理とそのアプリケーション の製作を支援するツールキットに関する研究

大平浩貴

北陸先端科学技術大学院大学 情報科学研究科

1997年2月14日

キーワード: continuous media data , tool kit , stream manager .

背景

現在、ビデオプレーヤやビデオ会議システムといったマルチメディアアプリケーションが一般に浸透しつつある。これらのソフトウェアが扱っているビデオデータやオーディオデータは連続メディアアプリケーションの一種であり、時間に依存していて、大容量になりやすいという特徴がある。

連続メディアデータの処理は以下に示す事柄を考慮していなければならない。

- メディアデータの入出力時刻を管理しなければならない。
- 複数のメディアデータ間で同期をとることが可能でなければならない。

これらはテキストデータやイメージデータなどの一般的なデータを処理するプログラムではほとんどの場合考慮しなくてもよい事柄である。しかしながら、連続メディアデータは時間に依存しているという特徴を持っているために、連続メディアデータを処理する際にはこのような処理の時間管理を行う必要がある。

この要求を満たすようなプログラムを製作することは非常に難しく、また作業量も大きくなってしまふ。そのため、連続メディアデータを処理するアプリケーション(以下連続メディアアプリケーション)を効率的に製作するには製作を支援するシステムが必要である。また、連続メディアデータといえるデータは多岐にわたるため、この製作支援システムは拡張性に富んだシステム構成でなければならない。

更に連続メディアデータは比較的大容量になりやすいという特徴を持っている。このため連続メディアアプリケーションは大容量のデータを長時間に渡って処理するアプリケー

ションとなり得、CPU 資源を大きく消費する場合がある。連続メディアアプリケーションではデータ処理の時間管理が必要であるため、CPU 資源の過剰消費を抑えるために単純に処理の優先度などを操作すると時間管理に失敗することになりかねない。つまり、連続メディアアプリケーションには自主的に CPU 資源の過剰消費を抑制する機能が必要である。

本研究の他に連続メディアアプリケーションの制作を支援するシステムとして ACME, CMPlayer, VuSystem, Medusa, CINEMA 等があるが、いずれも処理の時間管理のサポートや拡張性、CPU 資源の管理、そして支援システムがユーザに提供しているアブストラクションは十分であるとは言えない。

目的

本研究は連続メディアアプリケーションの構築法を考察し、連続メディアアプリケーションの制作を支援するツールキットを作成、さらにそれを利用して評価を行なうものである。

連続メディアデータは時間依存特性を持っており、それを保証するような機能が必要である。連続メディアアプリケーションはこのような機能が実装可能な構成でなければならない。さらにアプリケーションプログラムの負担をこれまで以上に軽減する構成を提案する。またアプリケーションが使用している CPU 資源量を監視し、過剰に使用することを抑制するような機構も必要である。

以上の事柄を実装可能な構成を提案し、さらにその構成で連続メディアアプリケーションを構築できるような支援システムを実装して評価を行う。

概要

本研究と同様に連続メディアアプリケーションの製作を支援するシステムとして VuSystem がある。

VuSystem ではアプリケーションを「メディアデータを処理する In-band」「In-band を利用する Out-of-band」の二つのコンポーネントに分割して実装することを提案している。

In-band は Module と呼ばれるさらに細かい機能単位にわかれている。この Module を組み合わせることで様々な処理を実現することができる。さらにあらかじめ決められている Module のテンプレートに従って新たな Module を制作してシステムに追加することで、プログラム全体を変更することなく新機能が追加できる。

VuSystem ではデータ処理の部分を In-band という枠組に押し込み、ユーザが作成する Out-of-band の部分と明示的に分離することでデータ処理の抽象化を行なっている。しかしながらこの構成では接続された一連の Module をデータが流れるという「データの流れ」の抽象化ができていない。このため複数のメディアデータを流してそれらの間で同期

をとるといった「データの流を統合した制御」の処理は全て Out-of-band でアプリケーションプログラマが行なわなければならない。

本研究ではこの構成を拡張して In-band と Out-of-band の間に「In-band を抽象化する StreamManager」という部分を追加する。

この StreamManager では以下の処理をサポートする。

- Module だけでなく、それを利用するための様々なオブジェクトを提供してメディアデータ処理構造の作成を支援する。
- データの流れを Stream という単位にまとめ、Stream 間同期のようなメディアデータの流を統合した制御を可能にする。
- CPU 資源の過剰消費を監視して、必要に応じて CPU 資源の消費を抑制するといった高度な制御を可能にする。またこの制御ポリシーも変更可能にする。

本研究では CPU 資源監視にリアルタイム OS が提供している CPU 資源予約機構を利用し、リアルタイム OS からアプリケーションに CPU 資源の過剰使用の通知が受けられるようにする。

結論

本研究で提案するアプリケーション構成を利用することで、連続メディアデータの処理構造の構築が容易になっただけでなく、動的な機能切替えも可能になった。またメディアデータ処理機能の追加や変更も容易に行うことができる。

本研究では、メディアデータ処理を抽象化する StreamManager という部分を追加した。この部分では、メディアデータの流を Stream という単位に押し込めて抽象化している。このような構成にすることで以下のような利点が生まれた。

- メディアデータの流 (Stream) に対して処理制御命令を送ればあとは命令の配送オブジェクトが自動的に適切な Module を選択して呼び出すため、ユーザ (アプリケーションプログラマ) は Module がどの命令を欲しているかなどの詳細を知らなくても良い。
- Stream 制御命令やその配送オブジェクトは追加可能で、命令や配送のポリシーを自由に変更できる。
- Stream の制御命令を Stream 内から呼び出すことでフィードバック制御が簡単に実装できる。

本システムの構成は連続メディアデータの処理を行なうアプリケーション一般に適用でき、また製作したツールキットも部分的な変更だけで様々なシステムの製作に利用できるようになった。