

X ウィンドウシステムの動画像拡張方式

山本 英治 碓崎 賢一
九州工業大学 情報工学部

本論文では、X ウィンドウシステムに MPEG データの再生機能を導入した、動画像拡張方式を示す。提案方式では、動画像データを圧縮された状態でクライアントからサーバに転送可能であり、制約が厳しい現在の LAN の帯域でも十分な速度で動画像を再生可能にしている。また、サーバ側に実時間再生管理機構を持たせることにより、トラヒック変化の影響を最小限に抑えた安定した動画再生を実現している。これらの機能は、本システムの提供する動画像操作用ライブラリで容易に利用可能である。さらに、本方式の応用として IP マルチキャスト通信への提案方式の適用について提案している。

A Movie Extension for X Window System

Eiji YAMAMOTO and Ken'ichi KAKIZAKI

Faculty of Computer Science and Systems Engineering
Kyushu Institute of Technology

This paper describes a movie extension for X Window System, which supports MPEG movie playback. This extension enables to transfer compressed movies from X clients to an X server, so this extension realizes to play movies with sufficient speed through restrictive LAN bandwidth. Moreover, this X server extension keeps smooth and stable movie playback by suppressing effects of network traffic variation. Developers can handle this extension very easily by using A Movie Library supported by this system. Finally, this paper introduces to apply this extension to IP multicast communication.

1 はじめに

コンピュータのユーザインタフェースとしてウィンドウ環境が発達、普及し、これまでの文字、図形、画像に加え動画像や音声なども含めたマルチメディアも本格的に扱われようとしている。特に最近では動画像の再生に対する要求が大きくなっている。一方、ネットワークの普及も著しく、マルチメディアを分散環境で利用したいという要求が大きくなっている。しかしながら、動画像データは他のデータと比較して非常に大きいため、ネットワーク上で扱おうとした時、その通信速度の制約が問題となる。また、複数の動画像の同期を取りながら再生を行う場合には、ネットワークのトラフィック変化の影響を受けないように、同期管理機能が必要となる。X ウィンドウシステム [Que90] などの従来のウィンドウシステムにはこれらの機能が欠如しているために、動画像などの時間的情報を持つデータをネットワーク上で扱うための機能の改善が望まれている。

本論文では、X ウィンドウシステムに高品質な動画像再生を可能とする機能を持たせた、X サーバの拡張方式 [山本 94] を提案する。本提案方式では、X サーバに MPEG-1 [Gal91] 形式の動画像データの復号機能を付加した。これにより、クライアント・サーバ間の動画像データの転送を圧縮された状態で行え、少ない通信量で動画像を再生できる。また、X サーバに動画像再生の同期機能を導入することにより、ネットワークのトラフィック変化の影響を受けない動画像再生を実現している。さらには、本提案方式の拡張として、IP マルチキャスト通信への応用について提案している。

2 動画像表示に対する要求分析

現在、分散環境としてイーサネットによる LAN で接続されたワークステーション上で、X ウィンドウシステムを利用する形態が広く普及している。分散環境でマルチメディアを扱うためには、動画像等の再生を十分高速に行えることが必要である。一般に、マルチメディアで利用される動画像データは MPEG 等圧縮された形式で保存されている。しかしながら、現在の X ウィンドウシステムには動画像の再生機能がないために、動画像を再生する場合 [Pat93, Lam91] には、クライアントで再生する動画像の個々のフレームを静止画像と見なし、1 枚ずつビットマップに変

換して継続的にサーバに転送しなければならない。このため、データの転送量は非常に大きく、動画像としての再生速度が非常に低くなるという問題がある。LAN の通信速度は向上しつつあるが、解像度や色数等、画像の品質が上がれば、再び同じ問題が発生し、本質的問題は未解決のまま残る。したがって、ネットワークの制約を緩和するために、データ圧縮技術を利用した転送機構が必要である。

分散環境では、ネットワークのトラフィック変動のため、クライアント・サーバ間で常に安定した転送速度を維持することは困難である。したがって、ネットワークのトラフィック変動に影響されない安定した動画像の再生を維持するための対策が必要である。さらに、ネットワークのトラフィックが非常に混んでいて、動画像の再生に十分なデータ転送速度が得られない場合にも、動画像の利用目的に応じて必要な情報を最大限得られる機構が必要である。

複数の動画像を同時に再生する場合、それらを無関係に再生するだけではなく、それらの間の同期をとった再生を行いたいという要求もある。分散環境ではデータや制御の情報交換に遅延があるために、その影響を受けないように X サーバが同期機能を持つべきである。

3 実現方式

3.1 基本方式

提案方式の基本的なデータと処理の流れを図 1 に示す。本提案方式では、新たに導入する X のクライアントライブラリによって MPEG 形式で圧縮された状態のフレームデータをサーバに転送する。ネットワークを介して動画像を再生する場合、動画像データをクライアントからサーバへ継続的に転送する必要がある。本システムでは、開発者が複数の動画像を扱うアプリケーションを容易に記述できるライブラリを用意している。このライブラリを用いた場合、開発者はデータの継続的転送を専門に行う関数をイベントループ内に記述するのみで良く、複数動画像のフレームデータ転送量の調節等の複雑な操作は開発者から隠蔽される。

サーバ側は、大きく 2 つの部分から構成される。動画像再生用のクライアントからのリクエストを処理する部分と、動画像が表示されているウィンドウを定期的に更新する部分である。クライアントから

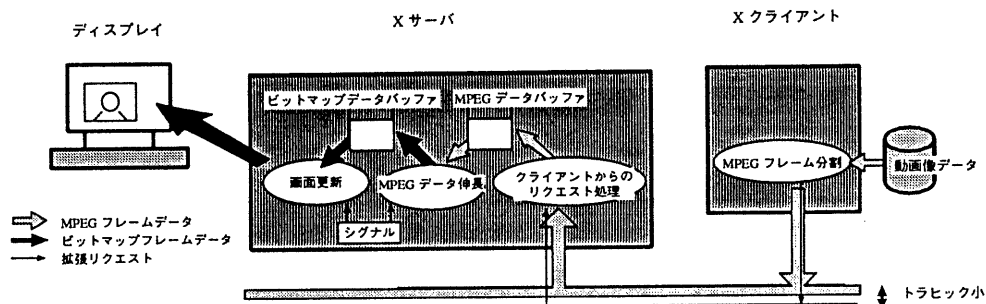


図 1: 本システムの構成

のリクエストを処理する部分においては、クライアント側から送られてくる MPEG フレームデータを蓄積する処理が大きな役割である。動画が表示されているウィンドウを定期的に更新する部分は、Xサーバの他の処理に関係なく MPEG フレームデータを伸長し、動画を表示しているウィンドウを更新することが大きな役割である。

3.2 画面の継続的更新

本提案方式では、Xサーバは蓄積したフレームデータを元に動画の再生を行うため、サーバはリクエストやイベントに関係なく定期的にウィンドウの更新処理を行う必要がある。定期的画面更新を実現する方法として、画面更新用の子プロセスを生成する手法も考えられる。しかしながら、Xサーバは非常に大きなプログラムなのでプロセス生成コストが大きい。そこで、我々は UNIX のシグナルによる定期的な割り込みにより、画面更新時刻を知らせる X リクエストパケットを X サーバ自身で生成し、サーバはそのリクエストの通知を受けて、画面更新処理を行う手法により実現している。更新処理ルーチンは、起動されると複数の動画の表示スケジュールを調べ、更新を行うかどうかを判断する。そして、更新の必要な動画についてのみ画面更新を行う。これにより複数動画の定期的画面更新が実現できる。我々は、割り込みの発生によるプロセス切替処理に伴うシステムの負荷を考慮し、30 ms の間隔でシグナルを発生させるようにしている。この間隔は、MPEG-1 において規定されている画面更新間隔と同じであり、これにより特に制約が生じることはない。

この方式では、サーバが何らかの高負荷な処理を行い、画面の更新間隔を維持できなかった場合でも、

更新時間を知らせるリクエストは同じ間隔で発生し続け、リクエストバッファに溜ることになる。しかしながら、更新時間を知らせるリクエストは、それ自身が画面の更新処理を起動するのではない。このリクエストにより画面更新処理を管理する部分が起動され、スケジュールを調べて実際の更新を行うかどうか判断する。したがって、更新時間を知らせるリクエストが多くやってきても画面更新時間でなければ無視されるので、画面の更新間隔には悪影響を及ぼさない。

3.3 同期制御

クライアントとサーバ間では、ネットワークを経由することによる遅延が発生する。そこで、本システムでは、クライアントからサーバへのリクエスト発信は、Tactus [Dan92] における実現方式と同様にそのリクエストの指定した実行予定時刻よりもネットワークの遅延時間分先行して行い、遅延による影響を抑えている。

複数のウィンドウの動画をあるスケジュールにしたがって順に再生したり、同時に複数の動画を再生したいという要求がある。しかしながら分散環境では、ネットワーク遅延によりクライアント・サーバ間の時間の同時性が保証できない。したがって、クライアント側で動画の再生スケジュールを指定してもサーバ側でそのスケジュール通りに再生できないという問題がある。そこで、本システムでは動画の再生、停止などのリクエストで指定する時刻は、サーバ側の時刻で指定するようにしている。このようにすると、リクエストはサーバ側の時計に基づき正確に処理されていくため、複数の動画間の同期の問題を解決することができる。また、全てのクライ

アントがサーバの時計という同一の時計を基準に再生スケジュールを立てるため、クライアントとサーバ間だけではなく、サーバを共有する複数のクライアント間での同期を簡単に行うことができるという利点がある。

本システムでは、クライアント側はサーバ側の時計に基づいて時間指定を行うために、動画像用ウィンドウの初期化時にサーバの時刻を問い合わせ、クライアントとサーバの時間差を記録しておく。そして、クライアントはこの時間差を考慮してサーバの時間を判断するようにしている。

3.3.1 2つの表示モード

ネットワークのトラフィックが非常に混んでいてデータの転送が遅れたり、サーバやクライアントの負荷が増加して処理速度が低下した場合、指定した速度で動画像を再生できなくなる。本システムでは、動画像の持つ属性として、

情報優先再生 指定の再生予定時間より遅れても全フレームを再生する。

スケジュール優先再生 フレームを一部省略して再生時間を指定時間に合わせる。

の2通りの対処法を指定できるようにしている。このようにして、ネットワークの十分な速度が得られない場合でも各動画像の特性に合った必要な情報を最大限得られるようにしている。

3.4 データ転送シーケンス

MPEG形式がデータの高圧縮率を実現している特徴的技術として、双方向予測符合化という方式がある。MPEG形式では、フレームデータの種類としてI、P、Bの3種類がある。I、Pフレームを参照フレームと呼ぶ。Iフレームはそれ自身のみでフレームを復元可能である。これに対し、Pフレームは直前のIフレーム、Bフレームは両隣の参照フレームがその復元に必要である。そして、MPEG形式ではI、B、..., P、B、..., I(P)、...の順にフレームデータが保存されている。すなわち、Bフレームはその後の参照フレームが揃わないと復元できないという制約がある。

そこで本システムでは、Bフレームに続くP、IフレームをBフレームに先行させ、I、P、Bフレーム

の順に並べ変えて送る方式を採用している。このような順でフレームデータを転送すれば、参照フレームは先に転送されることになり、Bフレームは転送されると直ちに再生可能になる。したがって先送りするデータ量が少なくて済み、操作の即時性が向上する。また、サーバ内に復号化待ちのBフレームを保持するための多数のデータバッファを確保する必要がなくなる。

3.5 MPEGデータの転送

本システムでは、Xサーバ側におけるMPEGフレームデータを蓄積するバッファをMPEGデータバッファと呼び、伸長したフレームデータを蓄積するバッファをビットマップデータバッファと呼ぶ。動画像の継続的な再生のためには、MPEGデータバッファ内に一定量のフレームデータを保持しておく必要がある。したがって、MPEGデータバッファ中のフレームデータが少ない時は単位時間あたりのフレームデータ数よりも多くのフレームデータを送り、多い時は転送を中断するなど、クライアント側からのフレームデータ転送量を変更する必要がある。また、スケジュール優先再生の場合、データ転送の遅延等によりフレームデータの転送が画面更新スケジュールに間に合わなくなった場合には、クライアントは現在転送しようとしているフレームを破棄して、スケジュールに間に合ういくつか先のフレームを転送するように変更する必要がある。

本システムでは、スケジュール優先再生の場合、クライアント自身の保持している情報からネットワークの遅延や現在再生されているフレーム番号、MPEGデータバッファの空き容量を判断する。情報優先再生の場合には、これらの状態はスケジュールからは判断できないので、サーバからMPEGデータバッファの状態を知らせるイベント通知を受け付けこれらの情報を得る。

データバッファ内のデータ量の多少の判断には、データバッファのサイズ、データバッファに蓄えられているフレーム数、フレームデータの大きさの平均などを使う。これらの要素から現在のバッファ中のフレームデータでどのくらい再生可能か、現在のバッファの空きであとどのくらいのフレームデータが格納できるか、などを判断する。そして、データ転送速度を変更してもネットワーク遅延によりデータ転送に要する時間の間にバッファが空もしくは一

杯になってしまうと予想されるフレーム数に達した場合には、クライアントから送られてくる単位時間あたりのフレームデータ数を変更する。データ転送量は、不足の場合には2倍に、超過の場合には0に変更し、バッファ内のデータがバッファサイズの半分になった時点で元に戻す。

情報優先再生の場合には、データバッファの状態はクライアント自身の持つ情報では判断できないので、サーバから MPEG データバッファの状態を知らせるイベント通知を受けつけこれらの情報を得る。バッファに関するイベントは、

- バッファ内データ量が不足である場合
- バッファ内データ量が超過する場合
- バッファ内データ量が半分である場合

の3種類が発信される。

クライアントからサーバへのデータ転送リクエストは、プログラム中のイベントループに組み込まれ、定期的に起動されている。リクエストは起動されると、バッファの状態やスケジュール等の情報から転送すべきフレームデータのフレーム番号、フレーム数を決定し転送を行う。

3.6 データバッファの管理

3.6.1 データバッファ用メモリの確保

本システムにおいて、複数のクライアントは任意の時にサーバ上に任意の大きさのデータバッファの領域確保・解放を行う。このため、メモリ空間上の空き領域は不連続に散在する。また、動画データは非常に大きいのでデータバッファを連続したメモリ領域に複数確保するのは困難である。そこで、フレームデータを格納するデータバッファは、MPEG データバッファ、ビットマップデータバッファのいずれも、ある決まったサイズのページを単位として領域を確保し、データの格納を行う。このようにすると、ある動画用のバッファメモリはメモリページを必要数確保すれば良い。したがって、バッファサイズ分の連続なメモリ空間は必要はなく、効率良くメモリ確保が可能である。

MPEG データバッファへのフレームデータの格納はページ単位で行われる。1 ページの大きさは、MPEG で規定されている画面サイズ 352 × 288 でフルカラーのビットマップデータを格納できる 300

KB としている。これよりも小さい画面サイズのビットマップフレームデータや MPEG フレームデータは、1 ページに複数格納される。そして、図 2 のような一つのメモリページに対応した構造体により巡回リスト構造を形成する。データバッファをこのような構造にすると、バッファ容量の変更がリストのつなぎ変えにより容易に行えるという利点がある。

本システムでは、クライアントアプリケーション開発者の負担を軽減するために、動画ウィンドウの初期化時に特に指定がない場合、サーバのデータバッファのサイズは自動的に設定されるようにしている。この場合、初期化時に確保する MPEG データバッファのバッファサイズは確保可能な最小量であるメモリページ 1 ページ分としている。

複数動画を再生する場合、全ての動画に十分な MPEG データバッファ用のメモリを確保できない場合が考えられる。そこで、サーバ上の複数の動画に優先度を付け、より優先度の高い動画の MPEG データバッファのメモリ確保を優先的に行う。動画のデータバッファ確保における優先度は次のような順にしている。

1. スケジュール優先再生モードで再生中の動画
2. 情報優先再生モードで再生中の動画
3. スケジュール優先再生モードで停止中の動画
4. 情報優先再生モードで停止中の動画

ここで、優先度の低い動画の MPEG データバッファがメモリを占有している状態で優先度の高い動画を再生しようとする場合、優先度の低い動画の MPEG データバッファは優先度の高い動画の MPEG データバッファが確保される際に容量を縮小される。

また、このようにサーバのメモリが占有されている状態で、新たに別の動画のバッファを確保する場合、必要な量のメモリはすぐには確保できないと考えられる。そこで本システムでは、まずは空いているメモリ分を確保して転送処理を始め、残りは空き次第確保するようにしている。これにより、必要な量のメモリが空くまで待つ必要がないので、初期化の時間が短縮可能である。

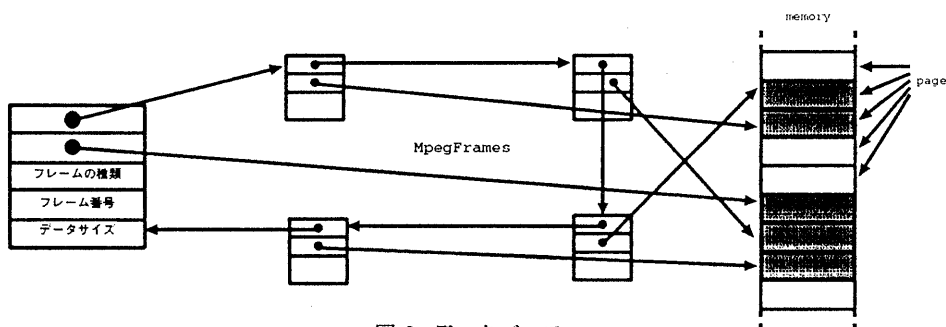


図 2: データバッファ

3.6.2 バッファ容量の動的変更

動画像の安定した再生のためには、サーバ側の MPEG データバッファがネットワークトラフィックの変化に十分に対応できるだけの大きさを持つことが必要である。しかしながら、これらの要素は動画像の再生中動的に変化し、動画像ウィンドウ初期化時には必ずしも判断できないという問題がある。そこで本システムでは、その時のトラフィックの状態に応じ、サーバのメモリの空き容量の許す範囲内で、動的に MPEG データバッファの容量を増加させるようにしている。バッファ容量の変更は、サーバ自身が動画像再生の状態を判断して行う。もし、トラフィック混雑が原因で再生スケジュールに遅れ、再生できなかったフレームが単位時間あたり一定数ある場合は、バッファ容量が不足であると判断して、そのフレーム数の分データバッファを増加すべきと判断する。そして、クライアントからのフレームデータ転送リクエストを処理する際に必要であればバッファ容量の変更を行う。

4 クライアント用ライブラリ関数

4.1 ライブラリ関数

本システムの拡張サーバが複数動画像再生を維持するために持つデータとしては、

- MPEG データバッファ
- ビットマップデータバッファ
- バッファ、ウィンドウを対応づける情報

がある。そして、これらを管理するための構造体を中心にして管理用データを保持している。

一方、クライアントは、サーバと同様に動画像に関する情報を管理している。したがって、クライアントは動画像に関する属性を自ら保持しているので、情報優先再生の場合のサーバ側のデータバッファの状態等を除きサーバクライアント間での通信なしに動画像の再生の状態を知ることができる。このようにして操作の即時性を上げている。

本システムでは、クライアントで動画像を操作するための拡張ライブラリ関数群を提供している。主なライブラリ関数を表 1 に示す。これらの関数群は、Xlib [Nye90] と同レベルでの実現となっている。

4.2 複数動画像間の同期

本システムの提供するライブラリは、動画像の再生スケジュールの指定方法として、ある開始時間から始まる絶対的な時刻で指定する絶対時間指定方式と、前の動画像再生の開始や終了時刻からの相対時間で指定する相対時間指定方式の、2 種類のスケジュール指定法を用意している。

同一の動画像ウィンドウに対して、再生スケジュールを再生時間が重なる形で指定したり、`XMEChangeMovieWindow()` で、変更先として動画像の再生が行われているウィンドウを指定した場合、スケジュールの重なる部分は、後に実行したリクエストの動画像が再生される。なお、前の動画像の再生停止は自動的に行われる。

表 1: 動画像操作のためのライブラリ関数

関数名	機能
XMESetupMovieWindow(dpy>window,gc,moviefile)	指定のウィンドウとデータバッファを対応づけ、クライアントからのデータ転送を開始し、再生開始可能な状態にする。
XMETerminateMovieWindow(dpy,moviewindow)	指定のウィンドウで動画像の再生を行うのをとりやめる。
XMEPlayMovie(dpy,moviewindow,starttime,framerate)	再生開始。
XMEStopMovie(dpy,moviewindow)	再生中止。
XMESeekFrame(display,moviewindow,frame)	指定のフレームに移動。
XMEShowOrderedFrame(dpy,moviewindow,frame)	指定のフレームを表示。
XMEPauseMovie(dpy,moviewindow)	再生を一時停止。
XMERevPlay(display,moviewindow)	逆再生。
XMESendFrame()	再生が行われているすべての動画像のバッファに対し、適切なレートでデータを転送する。
XMEShowNextFrame(dpy)	動画像画面を更新する。
XMEChangeMovieWindow(dpy,oldmoviewindow,newmoviewindow)	表示ウィンドウの変更。
XMEChangeMovieWindowAttributes(dpy,moviewindow,movieinfo)	動画像ウィンドウの属性の変更。
XMEQueryMovieWindowAttributes(dpy,moviewindow,movieinfo)	動画像ウィンドウの属性の照会。
MEShowTime(display,moviewindow,frame)	指定のフレームの再生時間。

5 提案方式の拡張

5.1 マルチキャスト通信における動画像通信への応用

インターネット上において近年盛んになっている利用形態に、IP マルチキャスト通信 [Wai88, Dec89] がある。IP マルチキャスト通信では、特定複数との通信が可能であり、その応用例として、映像や音声による講演会等の中継など、マルチメディアデータの通信が積極的に行われている。

現在、X ウィンドウ上においても IP マルチキャスト通信の様々なアプリケーションが利用されている。しかし、これらのアプリケーションをそのアプリケーションが存在する計算機とは別の計算機上の X サーバ上で使う場合 (図 3 上)、表示するサーバの数だけクライアントとの通信が行われ、トラヒックが大きくなるという問題がある。各マシンでクライアントを動かす方法も考えられるが、ネットワーク利用形態としては自由度が低い。

もし、全てのサーバが伸長機能を持てばクライアントとサーバの計算機が異なる場合でも同一のデータが何度もネットワークを通ることはなくなりトラヒックの無駄がなくなる。そこで、本提案方式を拡張し、IP マルチキャスト通信により動画像データを各サーバが直接受け取り伸長を行う方式を提案する。

マルチキャスト用の拡張方式では、図 3 下のよう
に、IP マルチキャストアプリケーションの指示を受けたサーバが直接 IP マルチキャストパケットを受け取り、データの再生を行う。このため、トラヒッ

クの無駄がなく、ネットワークを有効に利用したマルチメディアデータ通信が実現できる。

6 まとめ

本論文では、MPEG データの伸長機能と同期機能を X サーバに持たせることにより、X ウィンドウシステムの動画像再生機能を向上させる方式を示した。この方式はネットワークの通信速度の制約を受けにくく、効果的な方式であると考えられる。

本論文で述べた手法は動画像に関するものであるが、音声等のその他のマルチメディアデータにも応用可能である。また、提案方式は動画像圧縮方式として MPEG-1 を採用しているが、本方式は他のデータ圧縮方式に対しても拡張可能である。

現在、本論文で提案した拡張方式を導入した X ウィンドウシステムを開発中である。このシステムの評価によって、提案方式の効果を明らかにしたいと考えている。また、本システムでは明確な定義がなされていなかったデータバッファ内の適切なデータ量、ネットワーク遅延時間等動画像の安定した再生のために必要なパラメータは、実際のシステムを評価して決定したいと考えている。

謝辞

議論に参加していただき有益な御助言をいただいた、前畑淳也、松並勝氏をはじめとする碓崎研究室の諸氏に感謝します。

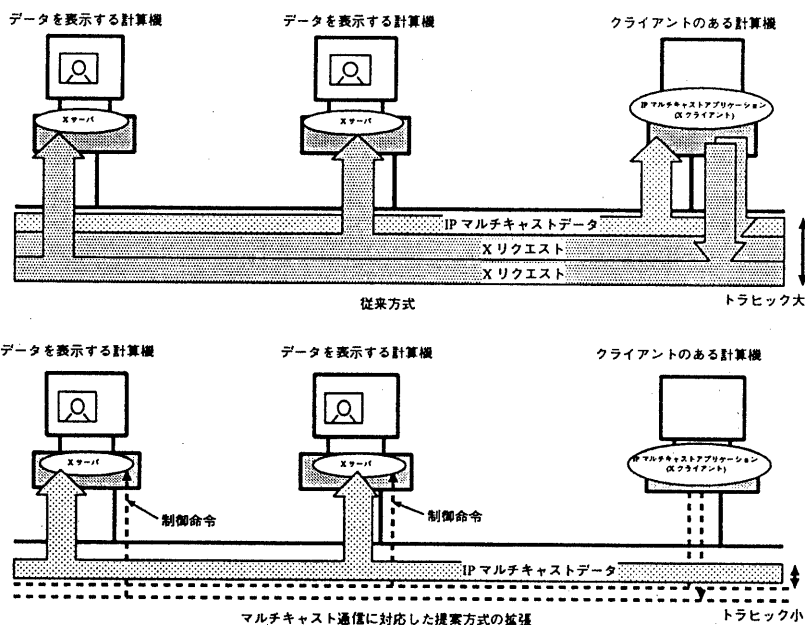


図 3: マルチキャスト通信への拡張

参考文献

- [Dan92] Dannenberg, R. B., Neuendorffer, T., Newcomer, J. M. and Rubine, D.: "Tactus: Toolkit-Level Support for Synchronized Interactive Multimedia", Proceedings of the Third International Workshop on Network and Operating System Support for Digital Audio and Video, pp. 302-313 (1992).
- [Dee89] Deering, S.: "RFC 1112: Host Extension for IP Multicasting", Technical report, Network Working Group (1989).
- [Gal91] Gall, D. L.: "MPEG: A Video Compression Standard for Multimedia Applications", *Communications of the ACM*, Vol. 34, No. 4, pp. 47-58 (1991).
- [Lam91] Lamparter, B. and Effelsberg, W.: "X-MOVIE: Transmission and Presentation of Digital Movies under X", Proceedings of the Second International Workshop on Network and Operating System Support for Digital Audio and Video, pp. 328-339 (1991).
- [Nye90] Nye, A.: "Xlib Programming Manual", O'Reilly & Associates, Inc. (1990).
- [Pat93] Patel, K., Smith, B. C. and Rowe, L. A.: "Performance of a Software MPEG Video Decoder", Technical report, University of California Berkeley (1993).
- [Que90] Quercia, V. and O'Reilly, T.: "X Window System User's Guide", O'Reilly & Associates, Inc. (1990).
- [Wai88] Waitzman, D., Partridge, C. and Deering, S.: "RFC 1075: Distance Vector Multicast Routing Protocol", Technical report, Network Working Group (1988).
- [山本 94] 山本英治, 磯崎賢一: "X ウィンドウシステムのマルチメディア拡張", 1994 年情報学シンポジウム, pp. 229-238 (1994).