

## 移動計算機情報発信環境におけるキャッシュの 更新方式について

田頭 茂明 稲田 文武 最所 圭三 福田 晃  
奈良先端科学技術大学院大学 情報科学研究科

〒 630-0101 奈良県生駒市高山町 8916-5

{shigea-t, fumita-i, sai, fukuda}@is.aist-nara.ac.jp

あらまし

本稿では、移動計算機とアクセスポイントまでの通信帯域の問題点を述べ、その帯域を有効に利用できるキャッシュの更新方式を提案している。通信帯域が狭い場合、移動計算機から発信できる情報量が制限され、情報の発信や更新作業をすべて処理することはできない。本研究では、要求されるリソースの性質を積極的に利用することにより、与えられた通信帯域でできる限りユーザの要求を満足できる通信方式を検討している。

本稿で提案している方式では、非蓄積型リソースの発信を優先するために、蓄積型リソースに対して通信量を制限して発信する。通信量の制限は、リソースの発信のための通信路（コネクション）の数を制限することで実現している。

キーワード 移動計算機, 情報発信, 蓄積型リソース, コネクション数, 優先度

## A Cache Update Method for Mobile Information Announcement Environment

Shigeaki Tagashira Fumitake Inada Keizo Saisho Akira Fukuda

Graduate School of Information Science, Nara Institute of Science and Technology  
8916-5 Takayama, Ikoma, Nara, 630-0101, Japan

Abstract

In this paper, the bandwidth problem between a mobile computer and an access point is pointed out, and the cache update method, which can make the best of the limited bandwidth, is proposed. If there is not sufficient bandwidth, the mobile computer cannot announce all resources, and cannot update all caches on fixed computers. To satisfy client's requests with given bandwidth as much as possible, properties of resources are used actively.

The proposed method limits traffic of storage type resources in order to announce non-storage type resources prior to storage type resources. Limitation on traffic is realized by limiting the number of connections for announcing storage type resources.

key words mobile computers, information announcement, storage type resources, a number of connections, priority

## 1 はじめに

近年、あらゆる場所に移動できる移動計算機が普及してきている。また、移動計算機を取り巻く環境も変化し、移動先でネットワークと接続できる環境も整備されつつある。移動計算機は、携帯性に優れていることから、情報の発生現場に移動し情報を取得したり、移動しながら情報を編集することが可能である。加えて、移動計算機は個人が占有して使用することが多いので、ネットワークとは独立した個人情報をも多く持つことが多い。

またインターネットにおいては、WWW(World Wide Web)などを用いることによって、容易に個人情報を発信できる。このことから移動計算機上の情報を、いつでもどこでも発信できれば、移動先で取得、編集した最新の情報を直接提供できるようになり、非常に有効である。本研究では、移動計算機からの情報発信環境の構築を目指しており、プロトタイプを実装している [1]。その応用として、移動先で取得した動画を発信する生中継放送や、音声を発信する携帯電話などを考えている。

移動計算機は様々な場所へ移動し、様々なインターフェースでネットワークと接続することが可能である。しかし、インターフェースの違いはネットワークの通信帯域、通信特性などに影響を与える。通信帯域が狭い場合、移動計算機から発信できる情報量が制限される。このような状況では、移動計算機からの情報の発信や更新作業をすべて処理することはできない。この場合、すべてのリソースを平等に発信しようとする、全てのリソースが中途半端に発信され、ユーザの全ての要求を満足できなくなる。そこで本研究では、要求されるリソースの性質を積極的に利用することにより、与えられた通信帯域で、できる限りユーザの要求を満足できる通信方式を検討する。

リソースの性質から、それらを蓄積することができる蓄積型リソースと、リアルタイムリソースなどの蓄積することに意味のない非蓄積型リソースに分類する。非蓄積型リソースは、クライアントからの要求時に移動計算機上で生成され、実時間で発信しなければならないので、遅延をできる限り小さくしなければならない。一方、蓄積型リソースは、クライアントからの要求が発生した時には、あらかじめ生成されているので、生成から要求までの時間的

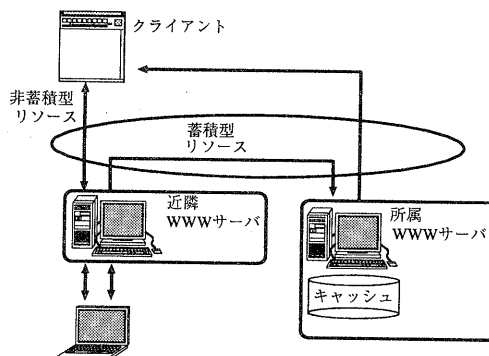


図 1: 発信システムの概要

余裕が存在する。このことから発信に多少の遅延があっても構わない。このように許容できる遅延の大きさはリソースによって異なる。これらを利用して、以下のようにリソースを発信することを考える。

- 非蓄積型リソースは蓄積型リソースより優先して発信する。
- 蓄積型リソースの発信でも緊急を要するものがあるので、それらは優先して発信する。
- 蓄積型リソースは固定計算機上にキャッシュができるので積極的に利用する。

本稿で提案する方式は、蓄積型リソースに対して、通信量を制限した発信を行う。通信量の制限は、移動計算機におけるリソースの発信のための通信路（コネクション）の数を制御することで実現する。このような機構を用いることにより、限られた通信帯域で蓄積型リソースと非蓄積型リソースの双方をできる限り快適に発信できる環境を構築する。

## 2 情報発信システムの概要と問題

### 2.1 情報発信システムの概要

我々が開発している移動計算機のための情報発信システムの概要を図1に示す。このシステムでは、蓄積することで再利用可能な蓄積型リソースと、リアルタイムリソースなどの再利用できない非蓄積型リソースを異なる方法で発信している。蓄積型リソースの発信は所属WWWサーバのキャッシュを介

して行い、非蓄積型リソースの発信は近隣 WWW サーバを通してクライアントと通信する。蓄積型リソースにおいてキャッシュを用いるのは、移動計算機の分断時になんらかの形で情報発信できるようにするためである。また、このキャッシュを用いることにより接続時においても移動計算機との通信量を削減できる。本システムにおける蓄積型リソースの発信のための移動計算機との通信は、移動計算機上のオリジナルのリソースが更新された時に、その更新を伝搬するときのみ発生する。

所属 WWW サーバを通る経路では通信に冗長性を含み遅延が増加する。非蓄積型リソースでは、実時間性を持つものを含むことが多く通信の遅延を小さくしなければならない。通信の遅延を小さくするために移動計算機にできる限り近い位置に近隣 WWW サーバを導入し、経路の冗長性を削減する。また、同じ要求が異なるクライアントから同時に入った場合、近隣 WWW サーバでそれらの要求を集約でき、移動計算機との通信量を削減が期待できる。

蓄積型リソースのキャッシュの更新方式として、以下のような2つが考えられる。

**WWW サーバ主導方式** クライアントが移動計算機上のリソースを要求した時に、そのリソースの更新がされているかの確認を行い、更新されていれば所属 WWW サーバ上のキャッシュを更新する。

**移動計算機主導方式** 移動計算機上のリソースが変更された時に、移動計算機が所属 WWW サーバ上のキャッシュを更新する。

これら2つの方式は機能的に等価であるが、移動計算機と所属 WWW サーバ間の通信量と、クライアントにおけるリソースを取得するまでのレスポンス時間が異なる。WWW サーバ主導方式では要求されないリソースを更新することはない。しかし要求が到着してから、キャッシュの更新を行うためレスポンス時間は長くなる。逆に移動計算機主導方式では要求されないリソースも更新するので、通信量が多くなるが、クライアントからの要求の前にキャッシュは更新されているので、レスポンス時間は短い。WWW サーバ主導方式においても、先読み機構を持たせることで応答時間を改善できる。

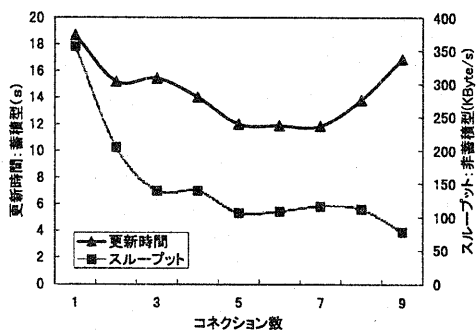


図2: コネクション数とキャッシュの更新時間およびスループットの関係

## 2.2 通信帯域が制限されたネットワークにおける情報発信の問題点

2.1節で述べたどちらのキャッシュ更新方式においても、複数のキャッシュを更新する機会が存在する。この場合、キャッシュの更新を同時に行うことによって、全体の更新時間を短くすることができる。しかし逆に同時に更新する数(コネクション数)を多くすると個々の更新のための通信帯域が狭くなるので、個々の更新時間は長くなる。これが、実時間性を持つ非蓄積型リソースの場合、使用できる通信帯域が狭くなることにより、実時間性を保証できなくなる。

これを確かめるために、開発したシステムにおける、すべてのキャッシュが更新される時間と非蓄積型リソースのスループットを測定した。移動計算機と WWW サーバ間を 10Mbps の Ethernet で接続して実験を行った。移動計算機から非蓄積型リソースとして、ネットワークを飽和させるだけのデータを発信した状態で、同時に 1M バイトの蓄積型リソース 10 個をコネクション数を変化させてキャッシュの更新を行った。そのとき、非蓄積型リソースに関しては、クライアントが受け取ったデータ量を測定し、蓄積型リソースに関しては、全てのキャッシュが更新される時間を測定した。その結果を図2に示す。コネクション数がキャッシュの更新時間と非蓄積型リソースのスループットに大きな影響を与えることがわかる。例えば、非蓄積型リソースが毎秒 150K バイトのスループットを要求する場合、本

実験環境ではキャッシュの最適なコネクション数は3か4になることがわかる。これ以上のコネクション数では、非蓄積型リソースの発信に影響する。また、これ以下では蓄積型リソースの更新時間が長くなる。

非蓄積型リソースの発信を妨げずに、限られた通信帯域を最大限に利用するために最適なコネクション数を決定する機構が必要である。

### 3 蓄積型リソースにおけるキャッシュの更新ポリシー

蓄積型リソースにおけるキャッシュの更新が非蓄積型リソースの発信に影響を与える。このため我々は以下の項目に注目してキャッシュを更新することにより、非蓄積型リソースを優先的に発信することを試みる。

**非蓄積型リソースの発信** 非蓄積型リソースは時間的制約が存在するリアルタイムリソースなども含まれているので、蓄積型リソースより優先して発信する。

**蓄積型リソースの優先度** リソースの重要度は一定ではないので、すべての蓄積型リソースを平等に扱うのではなく、重要度の高いリソースをキャッシュを優先的に更新する。

**移動計算機の移動** 移動計算機の位置は固定していない。様々な通信帯域の場所への移動や、通信媒体の変更を伴う。このため移動計算機の通信帯域の変化を考慮して発信する。

このような条件を満たす機構を実現する場合、以下のような方式が考えられる。

- ネットワーク層での実現
  - ATM(Asynchronous Transfer Mode)[2]などの帯域予約を用いる方式
  - 帯域保証できないネットワークにおいてパケットレベルでスケジューリングする方式
- アプリケーション層において制御する方式

ATMなどの通信帯域を保証できるネットワークにおいては、Admission Control[4][5]を用いて、使用できる通信帯域を管理し、状況に応じて通信量を制限することができる。これからQoS(Quality of Service)を保証した非蓄積型リソースの発信が可能である。

また、Ethernetなどの帯域保証できないネットワークにおいても、QoSを保証する通信方式が提案されている[3]。これは、あるセグメントの全体の要求量を管理し、ネットワーク層でのパケットスケジューリング機構を用いることにより、要求された通信帯域での通信を可能にしている。

最後はアプリケーション層での通信制御方式である。非蓄積型リソースのQoSを完全に保証できるわけではないが、以下のような利点を持つ。

- アプリケーション層で実装することから、多くのネットワークで使用できる。
- オペレーティングシステムを変更しないので移植性に優れている。

移動計算機は様々な場所に移動することが考えられる。これは様々なネットワークに接続する可能性を持つ。ネットワークを選ばないことは、移動計算機においては非常に重要な項目である。

以上の理由により、本研究ではアプリケーション層における制御方式を採用する。図2の結果で示されているように、コネクションの個数を制御することで非蓄積型リソースに割り当てる通信帯域を制御できる。非蓄積型リソースを発信している場合は、その発信を優先するために、蓄積型リソースをキャッシュの更新するためのコネクション数を制限する。これは、あるコネクション数を超えると、さらに更新要求があっても、その要求を待たせることにより実現できる。このとき、優先度の高いリソースに対してコネクションを割り当てることにより、更新ポリシーの2番目の目標(蓄積型リソースの優先度)を達成できる。優先度の低いリソースは優先度の高いリソースに比べて、時間的に遅い割り当てとなり、ある時間における通信量を抑制できる。非蓄積型リソースを発信していない場合においては、蓄積型リソースのキャッシュの更新のスループットが最大になるようにコネクション数を制御する。

コネクション制御アルゴリズムを示す。以下に示すように2つの条件を設定する。また、(1)の条件が成立しない時は、(2)の条件は無視する。非蓄積型リソースを発信しない場合においては、(2)の条件のみとなる。

(1) 非蓄積型リソースを優先して発信するための条件:

非蓄積型リソースを優先して発信するために、非蓄積型リソースにおけるスループットを求め、その合計があらかじめ要求されているスループットを満たすような条件を設定する。この条件を満たさない場合は、コネクション数を減らし、余裕がある場合はコネクション数を増やす。

(2) 蓄積型リソースを効率良く発信するための条件:

キャッシュの更新におけるコネクションのスループットの合計を求め、この合計が最大になるための条件を設定する。

#### 4 キャッシュ更新機構

3節で示したキャッシュの更新ポリシーに基づいた機構を提供するために本システムでは蓄積型リソースにおけるキャッシュの更新機構を、図3で示す1つのキューと5つのモジュールで構成する。図では太実線はDataの流れを点線は制御のためのメッセージの流れを示している。細実線は最適なコネクション数が変化した時の処理の流れを示す。

**更新キュー:** キャッシュ更新の要求を保持する。またキャッシュ更新中に中断されたものを保持する。

**リソース優先度決定機構:** 蓄積型リソースの優先度を決定する。

**キャッシュ更新受付機構:** キャッシュ更新の要求を更新キューに入れる。また更新キューには、リソース優先度決定機構で決定される優先度の順に格納する。

**コネクション数管理機構:** 通信帯域および現在の非蓄積型リソースの通信状況などから、最適なコネクション数を決定する。

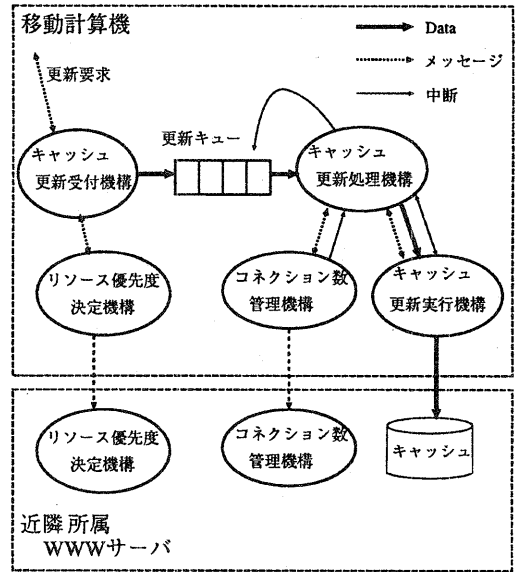


図3: キャッシュ更新機構

**キャッシュ更新処理機構:** 更新キューに要求が存在し、かつコネクションを増やすことができる場合、要求を取りだしキャッシュ更新実行機構にキャッシュ更新を依頼する。

**キャッシュ更新実行機構:** キャッシュの更新を実行する。この時、WWWサーバとは4.5節で示すキャッシュ更新プロトコルを用いて行う。

キャッシュの更新中に最適なコネクション数が変化することがある。この場合、コネクション数決定機構が動作し、キャッシュ更新処理機構にそのことを知らせるメッセージを転送する。コネクション数が増える場合と減る場合によって、動作が異なる。

**コネクション数が増える場合** コネクション数を増やすことができるだけなので、キャッシュ更新処理機構は特別な処理はしない。

**コネクション数が減る場合** キャッシュ更新処理機構は、現在のコネクション数が最適なコネクション数より多くなっている場合、最適コネクション数にするために、現在更新中のコネクションを中断するようにキャッシュ更新実行機構に依頼する。キャッシュ更新実行機構

は依頼された更新を中断する。キャッシュ更新処理機構は中断した更新を、更新キューに優先度順に格納する。

#### 4.1 リソースの発信に関する優先度

すべてのリソースの優先度について図4に示す。

非蓄積型リソースは蓄積型リソースに比べて優先される。これはリアルタイムリソースなどが含まれるからである。さらに蓄積型リソースは細分化され、優先度を設定される。この優先度により、キャッシュの更新順序が変わる。この優先度により、クライアントが最新のリソースを取得できる可能性を変化させる。クライアントに対して、できる限り最新のリソースを提供できるように優先度を決めなければならない。そのため、 $A/U$ を導入する。ここで  $A/U$  とは、

$$A/U = \frac{\text{アクセス頻度}}{\text{更新頻度}}$$

である。アクセス頻度と更新頻度との関係を示している。 $A/U$ の値が大きいリソースは、更新あたりのアクセス数が多いので更新が遅れると、その分クライアントが最新のリソースを取得できない可能性が上がる。逆に  $A/U$ の値が小さいリソースは更新あたりのアクセス数が少ないので、更新が遅れてもクライアントが最新のリソースを取得できない可能性はそれほど増加しない。従って、 $A/U$ の大きいリソースに対しては、できるだけ早く更新するように優先度を高く設定する。

このアクセス頻度と更新頻度の情報は所属WWWサーバが持つ。移動計算機からの要求があれば、この情報を移動計算機に渡す。この情報を用いてリ

ソース優先度決定機構は各リソースの優先度を決定する。

#### 4.2 キャッシュ更新受付機構

WWWサーバ（WWWサーバ主導方式）や移動計算機自身（移動計算機主導方式）からキャッシュ更新要求を受け付ける。キャッシュ更新を要求されたリソースをキャッシュ優先度決定機構から決定される優先度に従って更新キューに格納する。

#### 4.3 コネクション数決定機構

コネクション数決定機構においては、現在の通信の状況からコネクション数の管理、決定を行う。移動計算機のコネクション数決定機構がWWWサーバのコネクション数決定機構と協調し、3節で示した条件に基づいて最適なコネクション数を決定する。

また、最適なコネクション数が変動した場合は、コネクション数変動をキャッシュ更新処理機構に通知する。

#### 4.4 キャッシュ更新処理機構

更新キューに格納されている更新要求を処理する。現在のコネクション数が最適コネクション数より少ない場合に、更新キューの中にある先頭の要求を取り出し、その要求の処理をキャッシュ更新実行機構に依頼し、現在のコネクション数を1つ増やす。また、キャッシュ更新実行機構から処理メッセージを受け取ると、現在のコネクション数を1つ減らす。

コネクション数決定機構から最適コネクション数の減少が通知され、その値が現在のコネクション数より小さい場合は、現在更新中のリソースの中から、優先度の低い順に従って、更新中断の要求をキャッシュ更新実行機構に通知する。そして、その中断した要求を更新キューに優先度順に格納する。

#### 4.5 キャッシュ更新実行機構

キャッシュ更新処理機構で決定されたリソースの割り当てに対し、更新を開始するメッセージを受取り、更新プロトコルを用いキャッシュの更新動作を行う。また一つキャッシュの更新が終了すれば、終了のメッセージを返答する。

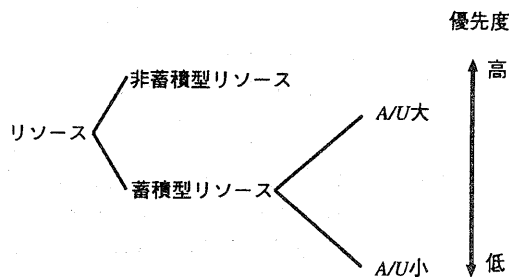


図4: リソースの優先度

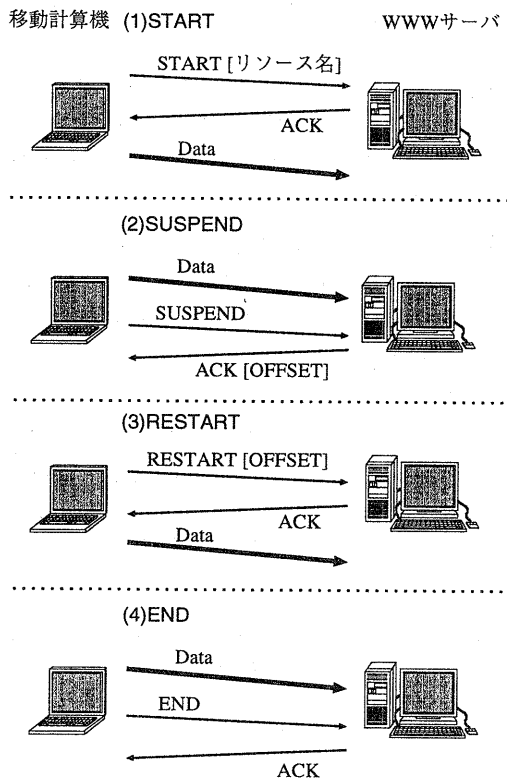


図 5: キャッシュ更新プロトコル

これらのキャッシュの更新に実現するために、移動計算機と WWW サーバ間でキャッシュ更新プロトコルを用いる。この更新プロトコルは我々が定義したものでキャッシュ更新の開始、終了、中断、再開を提供する。

次に更新プロトコルについて説明する(図5)。図で Data は HTTP を用いたデータ通信を表し、他は本プロトコルにより生じる制御の流れである。更新プロトコルは START, SUSPEND, RESTART, END の 4 つから構成される。

- (1) START: 更新動作の開始を通知する。移動計算機は WWW サーバに更新開始のメッセージを送信する。このメッセージには更新開始するキャッシュのリソース名を添付する。
- (2) SUSPEND: 現在作業中であるキャッシュの更新を途中で中断するプロトコルである。最

適コネクション数が変化し、その結果コネクション数を減らすことが決定した時に、このメッセージを送信する。メッセージを受け取った WWW サーバは、現在まで送信された履歴 (OFFSET) を移動計算機に通知し、このコネクションを切断する。

- (3) RESTART: SUSPEND プロトコルにおいて中断したキャッシュの更新を再開する。この時、移動計算機は SUSPEND プロトコル時に生成した履歴から、中断した地点からキャッシュの更新を再開することになる。
- (4) END: 更新動作が完了したこと通知するプロトコルである。

## 5 おわりに

移動計算機の通信帯域の問題点を述べ、その帯域を有効に利用できるキャッシュの更新方式を提案した。移動計算機から発信できる情報量が制限され、情報の発信や更新作業をすべて処理することはできない。この方式は、通信帯域が狭い場合において、要求されるリソースの性質を積極的に利用することにより、与えられた通信帯域でユーザの要求をできる限り満足させるための通信方式である。

また提案方式では、非蓄積型リソースの発信を優先するために、蓄積型リソースに対して通信量を制限して発信する。通信量の制御は、リソースの発信のための通信路 (コネクション) の数を制御することで実現する。

現在、本稿で提案している蓄積型リソースのキャッシュの更新方式を実装中である。現在開発中のシステム [1] の移動計算機上の情報発信部と、所属・近隣 WWW サーバ上の情報発信部にキャッシュ更新機構を追加する。

今後の課題としては、以下のものがある。

- 本稿で提案しているキャッシュ更新方式の性能を評価する。具体的には種々の通信帯域下での非蓄積型リソースにおけるスループットとキャッシュの更新時間を測定する。また、実際に非蓄積型リソースとして音声を送信し、従来方式と提案方式を使用したときのクライアントに到達した音声との比較も行う。

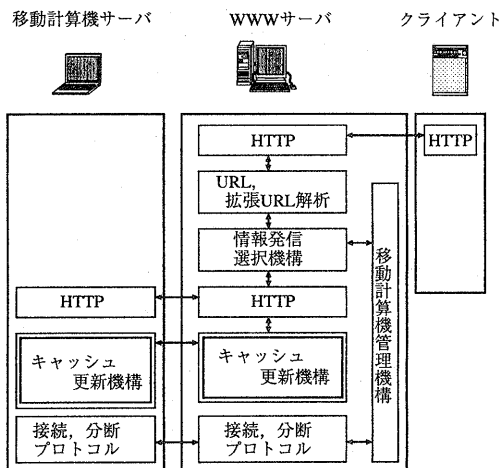


図 6: システムの構成

- キャッシュのスケジューリング更新方式に通信路（コネクション）を制御するだけでなく、QoSを制御する機構も取り入れる。本稿においては、リソースに対し優先度をつけることで非蓄積型リソースに割り当てる通信帯域を制御した。しかし、これはリソース間に対して行ったものである。リソースによっては品質をもとに階層化することができるものがある [6]。そこで、一つのリソースに対しても、細かく優先度を設定することを考える。階層化されたデータにおいて、ユーザの要求するQoSをもとに優先度を設定する。これをスケジューリングすることで、ユーザに対して、さらに快適なサービスを提供する。このような機構の実現を目指す。
- 提案した機構を用いたアプリケーションを構築する。

## 参考文献

[1] S. Tagashira, K. Nagatomo, K. Saisho, A. Fukuda, "An Information Announcement System based on WWW for Mobile Computers," IEICE Trans. Fundamentals (採録決定).

[2] 富永, 石川, "標準 ATM 教科書," 株式会社アスキー, 1995 年 3 月.

[3] 中野, 岩崎, 中原, 竹内, "Ethernet 上で QoS を保証する通信方式の設計と実装." 情報処理学会コンピュータシステムシンポジウム, pp.35-42, 1997 年 11 月.

[4] A. Pitsillides, P. Ioannou, D. Tipper, "Integrated Control of Connection Admission, Flow Rate, and Bandwidth for ATM based Networks," Proc. INFOCOM'96, pp.785-794, 1996.

[5] S. Jamin, S. J. Shenker, P. B. Danzig, "Comparison of Measurement-based Admission Control Algorithms for Controlled-Load Service," Proc. INFOCOM'97, pp.973-980, 1997.

[6] 稲田, 田頭, 最所, 福田, "移動計算機の情報発信環境におけるデータ構造を考慮したキャッシュシステム," 情報処理学会モバイルコンピューティング研究会 (発表予定).