

デマンド型ニュース配送方式における ディレクトリサーバへのアクセス負荷低減手法

菅野 浩徳、曾根 秀昭、根元 義章
東北大学大学院情報科学研究科

概要

同じデータを予め複数のサーバに転送蓄積する形態のコンテンツ配送システムでは、利用者の需要を直接配送制御に反映できないため無駄な配送が生じる。ネットニュースもその一つであり、この問題を解決するため、我々はデマンド型ニュース配送方式を提案している。本方式では、ニュースグループと記事の管理をディレクトリサーバで行なうが、大規模な環境に適用するためにはアクセス負荷の低減が課題となる。本稿では、このディレクトリサーバへのアクセス負荷低減について、複数のディレクトリサーバに分散する手法と、問い合わせ抑制時間 (TTL) 制御による手法について論じ、後者については実験によってその有効性を示す。

Reduction of the Access Load to the Directory Server in an On-Demand NetNews Distribution System

Hironori KANNO, Hideaki SONE, Yoshiaki NEMOTO
Graduate School of Information Sciences, TOHOKU University.

Abstract

In the contents distribution system that distribute the same data to plural servers such as the NetNews system, network traffic contains unnecessary data, because it is not able to reflect the demand of users on the delivery control. We proposed an On-Demand NetNews distribution system in which the reduction of the access load to the directory server in large-scale environment is a subject to apply, because the directory server does the control of newsgroups and articles. In this paper, we discuss on the following two methods to reduce the access load to the directory server. The first method employs directory server for each newsgroup. Another is the method that controls an inquiry by restrain time (TTL). The efficacy of the 2nd method is shown by the result of experiment.

1 はじめに

インターネットで利用されているコンテンツ配送サービスには、FTP、WWW、VOD サーバ等におけるミラーリングや、メールマガジン、ネットニュースなどのように、同じ内容のデータを複数のサーバに予め転送蓄積しておき、そこから近接のユーザに供する方式がみられる。この方式では、利用者の需要を直接コンテンツの配送制御に反映できないため、配送されたコンテンツの利用率が低く、結果として無駄な配送が生じている。

この問題を解決するためには、需要に応じたデータの配送が有効であると考えられる。アプリケーションによってプロトコルや実装方法に違いがある事から、我々はまずネットニュースに着目し、文献^{[1][2]}において、ユーザからのリクエストに応じて目

的とする記事を配送するデマンド型ニュース配送方式を提案し、シミュレーション及び実装による実験を通じてその有効性を示した。当方式では、ニュースグループと記事の管理をディレクトリサーバで行なうが、大規模な環境に適用し応答性を確保するためには、ディレクトリサーバへのアクセス負荷の低減が課題となっていた。

本稿では、このデマンド型ニュース配送方式におけるディレクトリサーバへのアクセス負荷低減について、ニュースグループと記事の管理を複数のディレクトリサーバで分散しアクセス負荷を低減する手法と、ディレクトリサーバへの問い合わせ抑制時間 (TTL) 制御によってアクセス回数を抑える事でアクセス負荷を低減する手法について論じ、後者については実験によってその評価を行う。

2 デマンド型ニュース配送方式

提案するデマンド型ニュース配送方式のモデルを図1に示す^[2]。

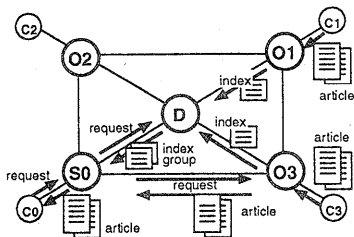


図 1: デマンド型ニュース配送モデル

図1中の{O1,O2,O3}は一次サーバを、S0は二次サーバを、Dはディレクトリサーバを、{C0,C1,C2,C3}はニュースリーダが動作するクライアントを示す。各サーバは、広域かつ広帯域なネットワーク上に配置される。図1のネットワーク接続は論理的なものであり、物理的な構成を示すものではない。ディレクトリサーバではニュースグループや記事の管理を行う。投稿記事は一次サーバに蓄積され、記事インデックスをディレクトリサーバに送る。記事インデックスとは、投稿記事のヘッダ^[9]から利用者が記事の選択を行うために必要となるヘッダ情報を切り出し、管理番号をX-Seq:ヘッダとして付与したものである(図2)。

```
X-Seq: 0000000001
Newsgroups: fj.os.bsd.freebsd
Message-ID: <03sp.siw@foo.com>
Date: 17 Feb 2000 21:54:38 +0900
From: bar <bar@foo.com>
Subject: Information
Lines: 41
X-Url: news://news.foo.com/82
```

図 2: 記事インデックスの例

記事を参照する際は、クライアントの要求に応じて、二次サーバがディレクトリサーバから記事インデックスを取り寄せ、クライアントに返す。そして記事を二次サーバが一次サーバから取り寄せる^[2]。

3 ディレクトリサーバの分散手法

3.1 ニュースグループの名前空間と構造

ネットニュースでは、記事が情報の単位であり、話題や用途によって記事を集約するためのニュースグループという枠組がある。このニュースグループは、あるカテゴリ名をトップレベルとする木構造の名前空間を持ち、このニュースグループを単位として記事の配送制御や管理が行われている。ニュースグループの名前空間構造を図3に示す。上が木構造図、下が模式図である。

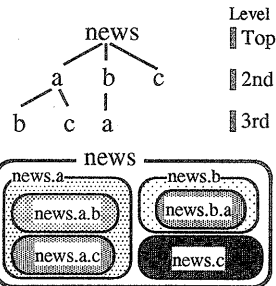


図 3: ニュースグループの木構造と模式図

ニュースグループ名は、[a-z0-9+.-]から構成され、各階層はドット'.'で区切られる。各階層は14文字までである^[4]。ニュースグループ名はファイル構造にも反映され、各'.'を'/'で置き換えたパスとなる。例えば、news.a.bはnews/a/bとなる。この模式図からわかるように、各ニュースグループはそれぞれに閉じた名前空間で記事を管理する。従って、提案するデマンド型ニュース配送方式において、すべてのニュースグループを単一のディレクトリサーバに集約しなくとも良く、どのニュースグループがどのサーバで管理されるかというロケーション情報があれば、異なる複数のサーバで分散管理が可能となる。

3.2 ディレクトリサーバファイル

提案方式では、購読するニュースグループは、グループファイルによって管理する。グループファイルは、ニュースグループ名、記事番号(最大番号、最小番号)、フラグといったフィールドから成る。これは従来のCNewsやINNなどのactiveファイルと同等である(図4)。

newsgroup	high	low	flag
news.rec	0000000020	0000000001	m
news.rec.radio	0000000010	0000000001	y
news.rec.tv	0000000023	0000000002	y
news.rec.tv.talk	0000000111	0000000001	y

図 4: グループファイルの例

先述の通り、ニュースグループと記事の分散管理を実現するには、サーバのロケーション情報が必要である。そこで、ニュースグループ名と管理するディレクトリサーバ名からなるディレクトリサーバファイルを新たに用いる(図5)。

newsgroup	server
news.rec	d1.bar.net
news.rec.radio	d1.bar.net
news.tv	d2.bar.net
news.tv.talk	d2.bar.net:d3.foo.net

図 5: ディレクトリサーバファイルの例

これにより、目的とするニュースグループ名から

そのグループと記事を管理するディレクトリサーバを特定できる。ディレクトリサーバ名は特殊記号'.'を挟んで複数記述できるものとする。先頭はプライマリサーバで、オリジナルなデータを持つサーバとする。それ以降のサーバをミラーサーバとする。これによってディレクトリサーバのミラー運用を可能とし、ロバスト性の向上が図れる。

このように、当手法によればディレクトリサーバへのアクセスを複数のサーバに分散し、一台当たりの負荷を低減させる事ができる(図6)。

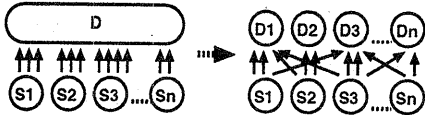


図 6: ディレクトリサーバの分散とアクセスの分散

4 問い合わせ抑制手法

4.1 記事購読時のシステム動作

提案するデマンド型ニュース配送システムでは、クライアントと一次、二次、ディレクトリサーバ間の通信プロトコルに、RFC977(Network News Transfer Protocol)^[6]に準拠したコマンドとメッセージを使用する。一次サーバに対する記事要求については、新たに XARTICLE コマンドを追加している。二次サーバでは、ディレクトリサーバや一次サーバから受け取った記事インデックスや記事などのオブジェクトを、キャッシュ機構により一定期間保存し、クライアントからの同様のリクエストについて再利用する。

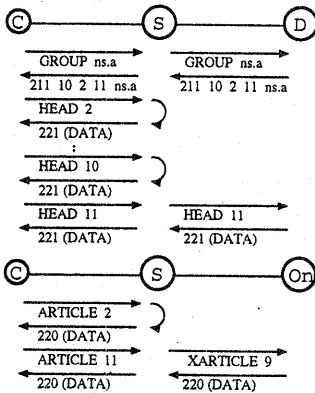


図 7: 記事の購読フロー

記事を購読する場合のシステム動作を図7により説明する。ここで二次サーバSは、ニュースグループ ns.a について、記事番号2から10までの記事インデックスと記事を既にキャッシュに保持してい

るものとする。

記事を購読する場合、まず GROUP コマンドによってニュースグループを選択し、レスポンスから最小記事番号 A_{min} 、最大記事番号 A_{max} 、記事数 A_{num} を得る。次に、HEAD コマンドを A_{min} から A_{max} まで順に実行し、 A_{num} 個の記事インデックスを得る。この時Sは、要求された記事インデックスがキャッシュ内にあればDにアクセスする事無くキャッシュのデータを返し、キャッシュに無ければDにアクセスしてデータを得る。図7の場合、 $(A_{min}, A_{max}, A_{num}) = (2, 11, 10)$ であり、記事番号2から10までの記事インデックスは、Dにアクセスする事無くキャッシュのデータを返し、記事番号11についてはDにアクセスしデータを得る。

記事の配送は ARTICLE コマンドによる。選択された記事がSのキャッシュ内にあればキャッシュのデータを返し、キャッシュに無ければ、XARTICLE コマンドによってOnにアクセスしてデータを得る。

このように、提案するデマンド型ニュース配送システムにおいて、記事の参照に用いられる HEAD と ARTICLE コマンドについては、キャッシュ機能によって不要なアクセスが抑制されている。一方、GROUP コマンドはクライアントからの要求の都度アクセスが行われているが、新着記事が無い場合その問い合わせは無駄である。よって、この不要な問い合わせ回数を減らせればサーバ負荷の軽減につながると考えた。

4.2 不要な問い合わせの抑制

あるニュースグループに7月3日から9日までの7日間に投稿された記事の投稿頻度(図8(a))と、7月10日から16日までの7日間の投稿頻度(図8(b))を示す。横軸は日時に縦軸は1分毎の投稿記事数である。

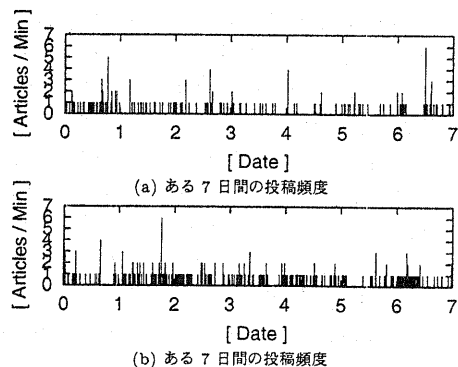


図 8: 記事の投稿頻度

このグラフから、記事の投稿時間間隔はランダムであり、日や時間帯による周期性は見られない事がわかる。これは、ネットニュースの利用者が地球に広く分散しており、記事の購読や投稿時間が特定の時間帯に限定される事が少ないためと思われる。一方、ある時点で投稿が集中して行われる傾向が見られる。これは、ある投稿やイベントなどを引金として、記事やフォロー記事の投稿が連続的に行われたものと考えられる。

このような記事の到着時間間隔の変化を推測して、ある記事の投稿から次の記事の投稿までの間は問い合わせを抑制する、といった動的な制御ができれば、ディレクトリサーバに対する問い合わせを効率良く低減する事ができる。本研究では、この抑制時間をTTL(Time-To-Live)と呼び、TTL値は可変で最適な値を選ぶ。TTLによる問い合わせ抑制制御について図9により説明する。

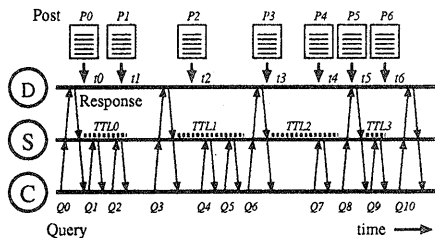


図9: TTLによる問い合わせ抑制

図中のDはディレクトリサーバ、Sは二次サーバ、Cはクライアントを示す。Cから問い合わせQ0に対して、Dではレスポンスと共に、TTL0を返す。SではレスポンスをこのTTL0の間保持して、以降のQ1, Q2による問い合わせはDに送らず自分自身で返答する。TTL0は、記事P1が投稿されたt1時間を過ぎた頃に無効となるので、問い合わせQ3はDに送られ、この時点での新たなTTLであるTTL1を得て、次のQ4, Q5の問い合わせを抑制する。以下Q10まで同様に処理を行う。このように本方式では、ディレクトリサーバが提供するTTL時間により二次サーバがディレクトリサーバへ問い合わせを抑制する事で、アクセス負荷の低減を実現する。

このTTLによる問い合わせ抑制制御では、TTLが長い程、問い合わせ回数が減り、ディレクトリサーバへのアクセス負荷は低減されるが、あまり長過ぎると、新たな記事が投稿されていても問い合わせがDに送られないため、結果としてアクセスが遅延するという、トレードオフがある。従ってTTLは、この相反する要求を最大限満たす値が望ましい。

4.3 履歴情報の利用

記事の投稿は、ネットワーク上に分散された不特定多数のクライアントから任意の時点で行われるため、予め記事の発生時刻を知る事はできない。そのため、次の投稿の発生時刻は、既に投稿された記事の投稿頻度や投稿時間間隔のデータから推測する事が必要となる。提案するデマンド型ニュース配送方式では、すべての投稿記事をディレクトリサーバで管理するため、投稿時間やその間隔を履歴情報として保持する事ができ、この履歴情報を用いる事で、次の投稿を確率的に推測する事が可能となる。

4.4 TTL算出アルゴリズム

先述のとおり、ネットニュースの投稿時間間隔には、時刻や曜日などによる周期的な変動が見られずランダムであり、任意の時点で連続的に記事が投稿される傾向が見られる。このような特性から、投稿時間の推測にはある一定以上の古さを持ったデータは考慮しなくとも良い事が予想できる。そこで、過去のデータのうち一定個数のみを採用し、それ以前のデータは考慮しない移動平均モデルを基本にTTL算出アルゴリズムを検討する。

移動平均モデルは、 P_t を予測値とし、

$$P_t = (P_{t-1} + P_{t-2} + \dots + P_{t-n})/n \quad (1)$$

とするモデルである^[2]。ここで P_{t-1} は一番最近に投稿された記事における投稿間隔時間で、 P_{t-n} はその n 個前に投稿された記事における投稿間隔時間である。得られた予測値 P_t は、最新の記事が投稿された時刻 $t-1$ から P_t 後に次の記事が投稿される事を意味する。

次にこの予測値をTTLに反映させる事を考える。仮に予測値=TTLとして、その予測値が大きく外れた場合、その誤差が大きい程、新たな記事にアクセスできない時間が長くなるため、サービスレベルの低下を招く。特に移動平均モデルでは、本質的に大きな値(ノイズ)の影響を受けやすい。従ってTTLにはフェイルセーフとして、上限についての閾値が必要と考える。そこで本アルゴリズムでは、移動平均モデルによる予測値が閾値以下の場合にはその値をTTLとし、閾値を超えた場合には閾値をTTLとする事とした。閾値を k とすれば、TTLの算出式は式(1)の P_t を使って、

$$TTL = \begin{cases} P_t & (P_t \leq k) \\ k & (P_t > k) \end{cases} \quad (2)$$

となる。

5 評価

5.1 評価指標

TTLによる問い合わせ抑制制御の性能を示す指標として、問い合わせ抑制率 Q_{sup} 、問い合わせ有効率 Q_{suc} を定義する。この Q_{sup} 、 Q_{suc} について、図 10 により説明する。

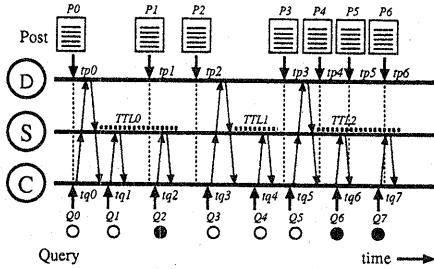


図 10: 問い合わせ抑制と問い合わせ有効

Q_1, Q_2, Q_4, Q_6, Q_7 は、 TTL_0, TTL_1, TTL_2 によって、それぞれディレクトリサーバ D への問い合わせを抑制されている。よって、全問い合わせ件数を N 、問い合わせを抑制された件数を s としたとき、問い合わせ抑制率 Q_{sup} を $Q_{sup} = s/N$ とする。例えば図 10 において、 $Q_{sup} = 5/8 = 0.63$ である。

Q_1 は、 TTL_0 によって D への問い合わせが抑制されている。しかし、 TTL_0 の開始時刻から Q_1 の問い合わせ時刻までに、D には新たな記事が投稿されていない。この場合、TTL を用いた問い合わせ抑制は成功であったとする。一方 Q_2 は、その問い合わせ時刻 tq_2 より以前の tp_1 に新たな記事 P_1 が投稿されているが、 TTL_0 によってその問い合わせが抑制され、結果として新たな記事の存在を知ることができない。この場合、TTL による問い合わせ抑制は失敗であったとする。よって、全問い合わせ件数を N 、問い合わせ抑制が失敗であった件数を f としたとき、問い合わせ有効率 Q_{suc} を $Q_{suc} = 1 - (f/N)$ とする。例えば図 10 において、 $Q_{suc} = 1 - (3/8) = 0.63$ である。この Q_{sup} と Q_{suc} の双方を最大とする TTL が最良である。

5.2 実験

TTL による問い合わせ抑制制御の効果について、実験によって評価する。投稿記事データには、図 8 (b) のデータを用いる。詳細は表 1 のとおりである。

閾値は、クライアントが新規記事の到着を知る事ができない最長時間である。これは現行のニュースシステムにおいて、nntpssend 等のバッチ型配送によるディレイ時間と同等と考える事ができる。バッチ

型配送の実行間隔は、INN 添付のドキュメントには例として 10 分毎、文献 [4] には 15 分毎と示されており、実運用においてもいずれかの値で設定されている事が多いものと推測し、本実験では閾値 $k = 600$ 秒 (10 分) とした。

クライアントからの問い合わせデータは、問い合わせ時間間隔の違いによる影響を検証するため、任意の間隔で一様に発生させる方法を取った。問い合わせ時間間隔が TTL を超える場合、TTL による抑制効果は得られないので、TTL による問い合わせ抑制効果について調査するためには、問い合わせ時間間隔は TTL の最大値以下すなわち閾値以下である事が必要である。そこで本実験では、問い合わせ時間間隔 Q を $Q = \{1, 10, 30, 60, 180\}$ 秒の各値とした。実験は性能比較のために、TTL の算出に閾値を考慮しない場合 (式 (1)) とした場合 (式 (2)) の 2 通りについて行った。

5.3 実験結果と考察

7 日間 ($d = 1.7$) の投稿記事データに対して、問い合わせ時間間隔 Q 毎に n を 1 から 20 まで変化させ、それぞれの Q_{sup} 、 Q_{suc} を求めた。得られた Q_{sup} 、 Q_{suc} について n の変化に伴う定量的な比較を行うために、その評価指標を $Q_e = Q_{sup} \cdot Q_{suc}$ とし、 n と Q_e との関係を図 11、図 12 に示す。なお、この図における Q_e は 7 日間の平均値を用いた。

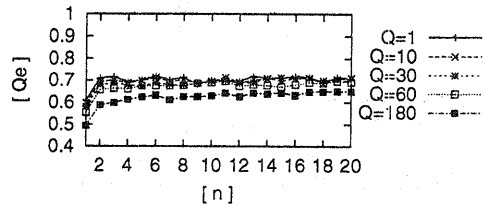


図 11: n の変化と Q_e (閾値なし)

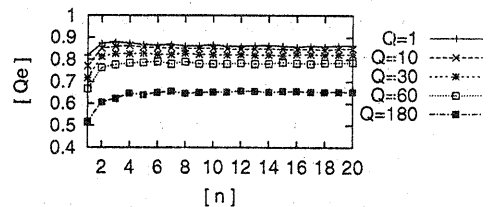


図 12: n の変化と Q_e ($k = 600$)

閾値が無い場合と比較して閾値がある場合のほうが、総じて Q_e が良い事が分かる。特に Q が小さい場合においてこの傾向が著しい。 n については、 $n = 3, 4$ 付近でピークが得られており、一定以上の古さをもったデータは考慮しなくとも良いという仮定が裏付けられた。

表 1: 記事投稿データ

Date	1	2	3	4	5	6	7
投稿記事数	35	74	60	50	52	32	65
平均投稿時間間隔	2576.7	1164.1	1453.4	1732.5	1598.3	2655.9	1392.6
最小、最大	0, 20796	0, 7789	0, 12288	0, 17648	2, 8502	4, 25480	2, 10285
標準偏差	4098.6	1513.5	2195.3	2811.8	1926.3	5025.5	2166.8

$n = 3$ について、日毎のデータを横軸に Q_{sup} 、縦軸に Q_{suc} としてプロットしたものを図 13,14 に示す。

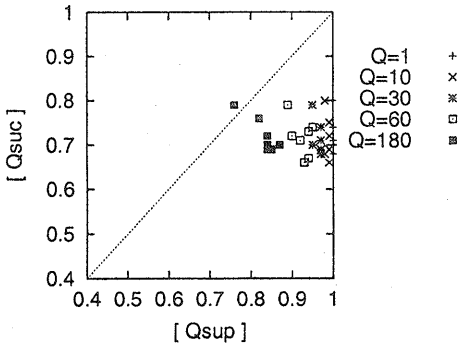


図 13: Q_{sup} と Q_{suc} ($n = 3$, 閾値無し)

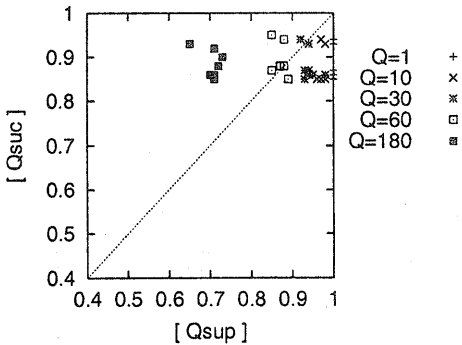


図 14: Q_{sup} と Q_{suc} ($n = 3, k = 600$)

図 14 では、 Q_{suc} が 0.8 以上を示している。 Q_{sup} がほぼ 0.7 以上である事から、1,2 割程度の問い合わせ失敗を容認する事で、7 割以上の問い合わせを抑制可能である。問い合わせ失敗の場合でも閾値は 10 分であり、それほど長い時間では無いため、サービスレベルの著しい低下にはつながらないと思える。一方、図 13 では図 14 と比べて Q_{sup} が良い半面 Q_{suc} が悪く、この場合 3,4 割程度の問い合わせ失敗が発生する事になりサービスレベルが低下している。この事から、閾値がサービスレベルの維持に有効である事がわかる。さらに、一般的に Q_{sup} は Q が小さい程高くなっている事から、問い合わせ間隔が短く、その頻度が高い程、より高い問い合わせ

抑制効果が得られる事が分かる。

6 おわりに

本稿では、提案するデマンド型ニュース配送システムについて、ディレクトリサーバへのアクセス負荷を低減するために、ニュースグループと記事の管理を複数のディレクトリサーバに分散する手法と、問い合わせ抑制時間 (TTL) 制御によってアクセス回数を抑える手法について論じた。TTL による問い合わせ抑制制御については、実験によりその評価を行い、わずかな問い合わせ失敗を容認する事で、高率で問い合わせを抑制可能である事、閾値がサービスレベルの維持に有効である事、問い合わせ間隔が短く頻度が高い程、高い問い合わせ抑制効果が得られる事、などが分かった。

本実験においては閾値を固定値としたが、記事の到着頻度やクライアントの問い合わせ頻度などから、最適な閾値を得る事で、さらに性能が向上するものと考えられる。

また、二次サーバをネットワーク構成に準じて階層化する事や、マルチキャストによるディレクトリサーバ側からのデータ配送によれば、より一層の改善が期待できる。今後、このような課題についても取り組んでゆきたいと考えている。

参考文献

- [1] 菅野浩徳, 曾根秀昭, 根元義章: ネットニュースにおけるデマンド型配送方式の提案, 電子情報通信学会 2000 年総合大会講演論文集, B-7-111, pp.204 (2000)
- [2] 菅野浩徳, 曾根秀昭, 根元義章: デマンド型ニュース配送方式の実装と評価, 情報処理学会研究会報告, 2000-DSM-17, Vol.2000, No.47, pp.19-24 (2000)
- [3] 上谷一: 広域情報ネットワークにおけるサービス管理に関する研究, 東北大学大学院情報科学研究科修士学位論文 (1999).
- [4] 紀太章監訳, 田中幸訳, H.Spencer, D.Lawrence: Usenet ネットニュース管理, オライリー・ジャパン (1999).
- [5] M.Horton, R.Adams: Standard for Interchange of USENET Messages, *RFC1036* (1987)
- [6] Brian Kantor, Phil Lapsley: Network News Transfer Protocol, *RFC977* (1986)