

ノード及びサービスの移動性を実現するモビリティ技術

高杉 耕一 片山 穰 久保田 稔

NTT 未来ねっと研究所

〒180-8585 東京都武蔵野市緑町3-9-11

E-Mail: {takasugi, katayama}@ma.onlab.ntt.co.jp, kubota.minoru@lab.ntt.co.jp

本論文ではノードの移動及びサービスの移動を制御管理することにより、両方の移動性を備えたサービス継続機構に関して述べる。従来から、サーバ/クライアントノードの移動に対しサービスを継続する機構を提案してきたが、サービスはノードに固定されており、アプリケーション間に構築されるセッションの管理がされていないため、サービスがノード間を移動する機能を構築することができなかった。また、ノードの移動に関しても、ノードの位置管理を行う必要があるが従来は固定された位置検索サーバの利用を想定していた。そこで、セッション/ノードの管理方式として、ネットワークに位置管理機能を持たない場合にも各ノードにおいて位置管理が可能な位置管理方式および、移動ノードの上でのサービスの移動を管理するセッション管理方式と、その評価について述べる。

Technology for Node and Service Migration

Koichi TAKASUGI, Minoru KATAYAMA, Minoru KUBOTA

NTT Network Innovation Laboratories

3-9-11 Midori-Cho, Musahino-shi, Tokyo 180-8585, Japan

This paper proposes the service continuance mechanism for the migration of the node and the service. Up to now we proposed that mechanism for the migration of the server/client node. However, the service on this mechanism was fixed to the node. Then it is not able to support the service migration between nodes. Then it is necessary to do the administration of the session constructed between the applications to support that migration. The other side concerning node migration it is important to do the position administration of the node to guarantee the service continuance. Then, we propose the session administration method to realize the migration of service and the position administration method that is no need for the look-up service in the network, and evaluate these methods.

1.はじめに

近年、端末の小型化、無線技術等の進展にともない情報家電、ロボット等、あらゆる物が移動ノード(PCやPDAに代表される通信装置を伴う端末)となり、ユービキタスコンピューティングの実現がまじかにせまってきている。このような移動ノードで構成されるネットワークにおいては、ノードの可搬性、移動性、実行環境の変化への適応性を持つ移動型コンピューティングを実現するシステムの構築が急務である。そこで、我々はノードの可搬性/移動性を有し、サービスの継続性を保証するMedlar[1]の研究を進めてきた。このようにMedlarにおいては、いくつかの異種通信媒体で構成されるネットワーク(図1)において、ストリームデータを受信中に、通信方式をLANからPHSのように切替えた場合においても継続的にサービスを受けることが可能である。

しかしながらユーザの立場から考えると携帯端末等のノードを持ち歩くことにより、ノードを移動させるだけでなく、異なるノードを適宜切替えて利用している。そこで、ノードが移動する環境に加え、

ユーザが享受または提供しているサービスがノード間を移動する環境という移動環境を統合して構築することが必要となる。これにより、完全にユーザに追従する究極的なモバイル環境が実現される。

ところが、従来のMedlarの枠組みにおいて、アプリケーション間でやり取りされる通信内容であるサービスは各移動ノードに固定されたままであり、ユーザに追従して、他の移動ノードにサービスを移動させることはできない。そこで、本論文では二つの

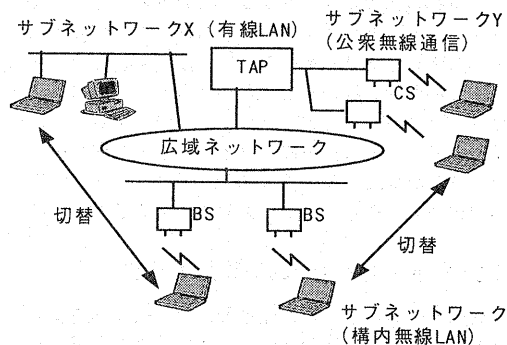


図1 想定ネットワーク

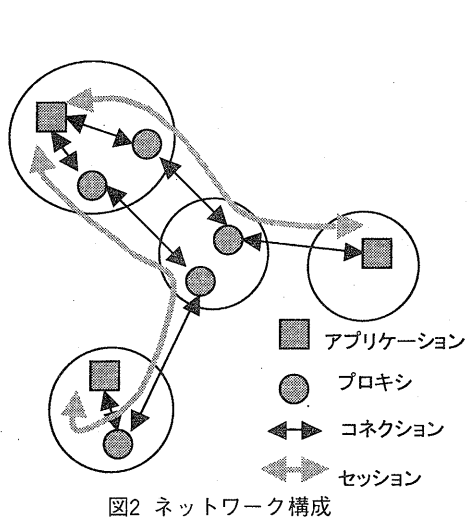


図2 ネットワーク構成

移動のモデルを構築するため、ノードの移動とサービスの移動をそれぞれ制御、管理する方式を提案する。また、評価の一部として、ノードの位置管理に関する計算機シミュレーションにより性能を測定し、要求条件に対する考察を行う。

2. サービス継続の仕組み

Medlarにおける、アプリケーション間にプロキシを挟んだセッションを構築することで、サービスの継続性を実現する仕組みを説明する。まず、セッションを構築した例を図2に示す。コネクションはアプリケーションとプロキシ、またはプロキシ間に設定される通信路である。また、セッションはアプリケーション間の通信路でいくつかのコネクションから構成される。プロキシは通信の蓄積転送とともに、セッション識別子(セッションID)の管理を行う。したがって、プロキシ間の通信方式が変化してリンク(リンク層でノードが通信する媒体)が切断された場合においても、リンク接続処理の後、セッションIDを用いて再接続することにより、従来の通信を再開し、コネクションを維持することが可能である。このとき、再接続処理はプロキシ間で行われるため、アプリケーションとプロキシが異なるノードに配置されている場合、アプリケーションの配置されているノードに移動性を持たせることができない。このように、プロキシを介してセッションを構築することにより、各ノードは自由にサブネットワーク間を渡り歩くことが可能となる[2]。また、その際、ユーザに通信デバイスやネットワークの変化を意識させない。

3. ノードとサービスの移動モデル

前節でのべたサービス継続機構の仕組みを踏まえた上で、ノードの移動とサービスの移動を実現するノードとサービスの移動モデルを図3に示す。ここで

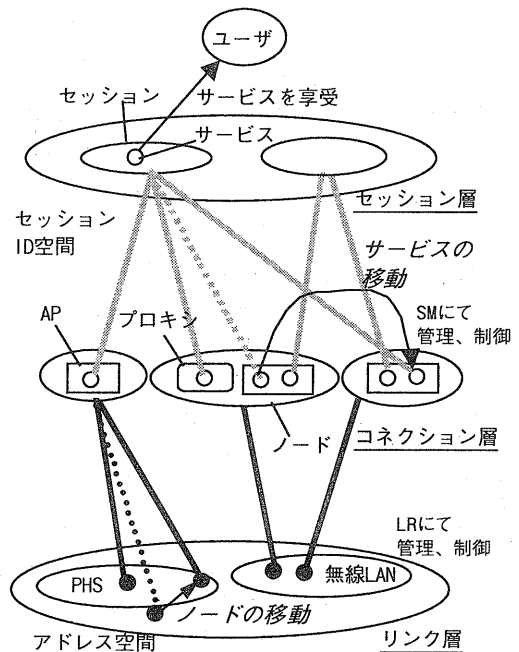


図3 ノードとサービスの移動モデル

扱うサービスとは各セッションを通してアプリケーションとプロキシの連携によって送受される通信内容とそれに付随する通信プロトコル(http等)及び表現方法(再生、ブラウジング等)を示す。また、サービスの移動は同一のセッションIDをもつセッションによって、他のアプリケーションまたはプロキシ上に構築されるサービスに対して元のサービスの状態のみを転送して同様の機能をはたすことで実現される。

そして、各ノードはリンク層のアドレス空間にマッピングされる。したがって、ノードの移動はノードとアドレスの対応関係を変更することによって、実現される。また、各ノードは複数の通信デバイスを持つことが可能であり、それぞれが固有のアドレスを保持している。

これらの本移動モデルを扱うため、ノードの移動は位置管理を行うLR(Location Register)と、サービスの移動はセッション管理を行うSM(Session Manager)を用いる。これにより、移動ノード間でサービスが移動した場合においても継続的にサービスの提供が可能となる。これにより、利用する移動ノードを変更しても、ユーザはサービスを提供しつづけたり、利用しつづけることが可能となる。

4. 要求条件

ここでは前節で示したLR及びSMに要求される条件を明確にする。

LRにおいては、Jini[4]やWIND projectのINS[5]のようなlook-upサービスが想定され、これらのサービスにおいては固定サーバを設置していく必要がある。しかし、Medlarにおいてはノードがネットワークから離脱した場合にも再接続先の位置情報を検索できる必要があるため、以下のような要求条件を設定した[3]。

(1)LR (位置管理)

- (a) ノードが移動した場合に固定の位置検索サーバを設置しなくても位置(アドレス)を知ることができること。
- (b) 移動ノードの能力に応じて位置管理の負荷を分担できること。
- (c) 位置管理に伴う通信量を抑制できること。
- (d) ネットワーク離脱時にも最低限の位置検索機能を有すること。

(2)SM (セッション管理)

- (e) 各セッションが識別できること。
- (f) アプリケーション、プロキシ間のコネクションのつながりを管理できること。
- (g) コネクションの変更、追加、切替等、セッションの構成を変更させられること。

5. 提案方式

5.1 ノード構成

本節ではノード構成の概略を説明する。図4にノードにおけるソフトウェア構成を示す。各移動ノードは、通信の蓄積転送を行いコネクション下のリンクの切断、変更を隠蔽するプロキシをもち、新しいリンクへの接続、リンクの切断、ルーティング管理機能を行うリンク切替機能を有している。また、エージェントにおいてリンク切替機能に対しリンク切替を命令するとともに、リンク切替後、プロキシにコネクションの再開処理依頼する。また、どのノードでどのようなアプリケーションとプロキシ機能が存在するかを記録したSR(Service Register)を有している。

また、各種管理機能を組み合わせノードの移動、サービスの移動をサポートするため、セッションの管理を行うSM(Session Manager)、ノードの位置の管理を行うLR(Location Register)がエージェントを介して接続されている。

エージェントにより、LR, SM, SRの各情報を仲介し、総合的な判断のもと、サービスの継続の制御が可能となる。例えば、サービスを継続したまま、現在サービスを提供しているサーバAPを終了したい場合に、SRから変わりのサーバAPを検索し、LRで切替先のサーバAPを動作させているノードの位置を特定し、SMでサーバAPをプロキシごと切替えるといった制御が可能となる。

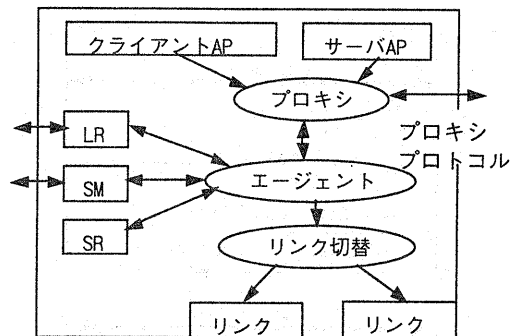


図4 ノードにおけるソフトウェア構成

5.2 各種管理機能

ここでは要求条件として提示した課題にする例題として、IPアドレスで位置管理されるようなネットワークに関してLR,SM管理方式の詳細を定義する。

5.2.1 LR (Location Register)

主に、ノードとIPアドレスのマッピングを行うが、位置管理の効率向上と、リソースの少ないノードにおいても実現可能とするためノードの移動量と位置情報の保持可能容量の管理も行う。また、管理するノードの範囲を限定するために、ノードのグルーピングを行う。詳細は5.3節で述べる。

管理すべき主な属性と機能は以下のとおりである。属性はグループに所属する任意の複数端末に対して作成されるテーブルとなる。

● 属性

- ▶ ノードの識別名
- ▶ グループ名
- ▶ IPアドレス(複数)とその状態(利用可否)
- ▶ 移動量(移動、固定、半固定)
- ▶ 各ノードのLRの位置情報保持可能容量
- ▶ 更新時刻

● 機能

- ▶ IPアドレスの変更(位置の移動)
- ▶ IPアドレスの状態変更(ノードの離脱、回線断、障害)
- ▶ エージェントに変更を通知
- ▶ グループの参加、離脱
- ▶ グループ内のノードのLRと位置情報の送受

5.2.2 SM (Session Manager)

サービスの移動を実現するためには、個々のセッションがどのアプリケーションとプロキシを用いて構成されているかを把握し、アプリケーションやプロキシの変化に対しては柔軟に対応する必要がある。SMは個々のセッションの管理を行い、プロキシの切

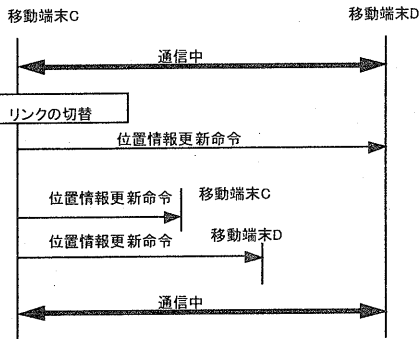


図5 端末の移動時の位置情報更新手順

替やノードの切替に対応するものである。

管理すべき属性と機能には以下の通りである。属性は各セッションに対して作成されるテーブルとなる。

- 属性
 - セッション ID
 - 利用ノード (ノード名、ポート番号)
 - プロキシ (ノード名、ポート番号)
 - 更新時刻
- 機能
 - プロキシ追加、削除、切替
 - 他ノードの SM とセッション

5.3 位置管理方式

前節で述べたLRの属性と機能を用いて位置管理を行う方式について述べる。

ここでは各ノードで管理する位置管理情報の数を限定するため、いくつかのノードをグループとしてまとめる。このグループ単位でLRの管理を行うことにより、位置管理を行う必要のないノードの位置情報を管理しなくてもよいため、更新するLRの数が限定され、LR更新効率が向上する。グループは複数のサブネットに渡り、サブネット内には複数のグループが共存することができるようにする。

(1) グループ加入時

グループに加入しようとするノードはIPアドレスを取得し、グループ検索要求をサブネットワーク内にブロードキャストするか、既知のノードに対して送信する。その後、グループ検索要求を受けとったノードからのグループ検索応答を取得し、加入するグループを選択する。その後、選択したグループに関するグループ検索応答を送信したノードに対し登録要求を送信し、登録要求を送信したノードからグループ内の位置情報を取得することでグループ加入を完了する。また、グループ登録要求を受けとったノードは、LRに記載の位置情報を送信するとともに加入したノードの位置情報をLRの位置情報に付け加え

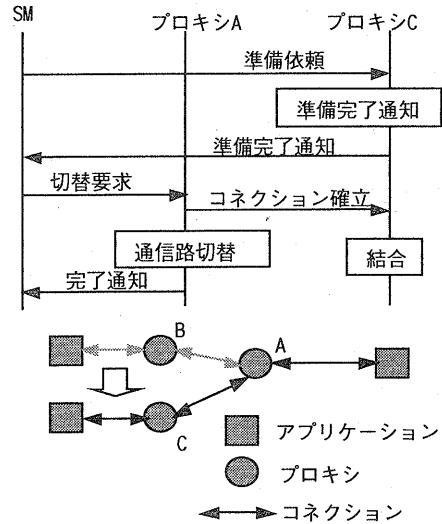


図6 アプリケーションの切替

ることグループの加入登録を終了する。

(2) ノード検索時

保持しているLRの位置情報を検索する。検索できなかった場合はLRに記述されている、任意のノードに検索要求をかける。検索結果が返ってきたら、LRに付け加える。

サブネット内の同グループに属しているノードを検索する場合はブロードキャストパケットを送り、その応答を元に、LRに位置情報を登録する。

(3) 通信デバイスを切替える時(サブネットワーク間の移動時)

切り替えに伴い保持しているLRを更新すると同時に通信相手がいれば位置情報更新命令をだす。また、移動後、LRに記載されている任意のノードに位置情報更新命令をだす。(図5)

(4) 位置更新命令の優先順位

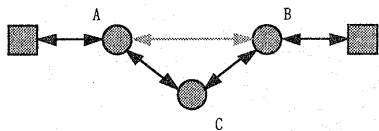
また、(3)において、以下のように位置更新の命令を出す順序に次の優先順位をつける。

- ノードの移動量が少ないノード
- ノードの位置情報を保持できる容量の高いノード

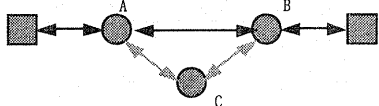
5.4 サービス移動の実現

ここではサービスの移動に関して説明する。サービスの移動は同一セッション ID を持つセッションによって、他のアプリケーションまたはプロキシ上に構築されるサービスに対し、もとのサービスの状態を転送することで実現される。転送される状態としてはセッションを構成しているアプリケーション、プロキシの接続状態、送受信しているデータのバイト数、フレーム数等がある。

プロキシの追加



プロキシの削除



プロキシの切替

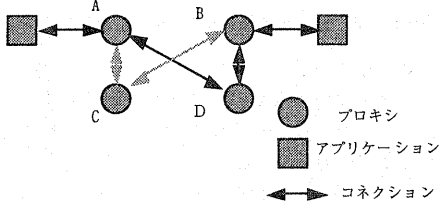


図7プロキシ構成の変更

例えば、携帯ノードの無線環境でストリームデータを受信、閲覧中に、大画面スクリーンを有する固定ノードにサービスを移動させることにより、連続的にサービスを受信したり、移動放送局となるサーバノードはクライアントアプリケーションに送信しているストリームデータを高速で接続している他のサーバノードに転送することにより、ネットワークから離脱したりすることが可能となる。

アプリケーションの切替を例にとり、本提案によるSMを用いた実現方法を述べる。アプリケーションの切替のシーケンスを図6に示す。まず、アプリケーションの切替要求を受けたSMは切替先のプロキシCに対し、切替準備を依頼するこの際、セッションIDやプロキシの状態の通知等を同時に行う。プロキシCはアプリケーションを起動し、通知されたセッションID保持し、通知された状態に遷移するなどの準備をしたのち、SMに対し準備完了通知を送信する。準備完了通知を受信したSMはプロキシAに対し、切替要求を送信し、プロキシAはプロキシCに対し、コネクションを確立する。その後、プロキシAはプロキシCとのコネクションをプロキシCとのコネクションへと切替え、完了通知をSMに送信する。プロキシCはプロキシAから確立されたコネクションをアプリケーションとの間に確立したコネクションと関連づけコネクションを結合することにより、通信路を確立する。完了通知を受信したSMはセッションの構成の変化をSMの管理テーブルを更新することにより管理し、他のSMに対しても必要に応じて変更を通知する。以上の動作により、アプリケ

表1: シミュレーション諸元

端末の移動間隔	10 min (指数分布):66.6%の端末 30min(指数分布):33.3%の端末
通信待機時間	20min(指数分布)
通信時間	50min(指数分布)
位置情報保持可能容量	すべての端末の位置:33.3%の端末 1/2の端末の位置:33.3%の端末 1/4の端末の位置:33.3%の端末

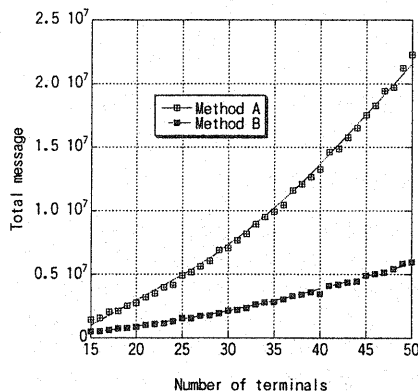


図8 位置情報更新命令の発行数

ーションの切替えが完了する。

また、図7に示すように、プロキシの追加、削除、切替といった動作も同様に行うことが可能である。したがって、提案したSMを用いて、セッションを構成しているノード構成を動的に変更することによりサービスの移動を実現することが可能である。

6. 評価

提案方式が4節の要求条件を満たしていることを確認する。

位置管理の方式に関しては計算機シミュレーションにより性能を測定し考察する。シミュレーション諸元を表1に示す。ここではグループを構成しているノードのうち、移動量に関しては1/3のノードを平均移動間30分、残り2/3を平均移動時間10分、としLRの容量に関しては、1/3のノードをすべてのノードの位置情報を保持可能であるとし、1/3のノードを1/2のノードの位置情報、残り1/3のノードを1/4のノードの位置情報を保持可能であるとした。すべてのLRの情報を正しく設定した状態から、1200分間に発行された位置情報更新命令の数、LR検索成功率を400試行分、測定した。ただし、最初の100分間は統計から除去している。また、位置情報更新命令は自らが保持しているすべての位置情報を送信し、受信側で位置情報のタイムスタンプにより、取舍選択し自らノードのLRを更新するものとする。

ここでは位置情報更新命令の優先順位の異なる方式A,Bの二つの方式に関して評価する。方式Aは自

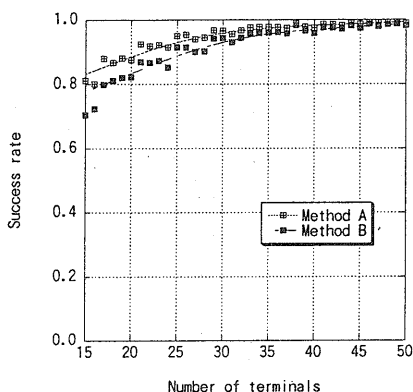


図9 位置検索成功率(自ノードLR)

らの位置情報を更新した際、位置更新命令(図5)をすべてのノードに送信した場合である。また、方式Bは移動間隔が30分であつすべてノードの位置情報の保持可能な容量をもつノードのみ、位置情報更新命令をすべてのノードに送信し、その他のノードは位置情報更新命令を前述のノードのみに送信するものである。つまり、方式Aは5.3節の位置管理方式(1)~(3)に相当し、方式Bは位置管理方式(1)~(4)に相当する。方式Aは各ノードの位置が更新される毎にすべてノードに位置更新命令を送信する方法のため、通信量は多くなるが位置管理の能力の一番高いものとして評価の比較のために用いる。

ノードに対する位置情報更新命令の発行数を図8に示す。また、自ノードのLRの情報またはその情報に基いて、他のノードに位置情報を問い合わせ、正しい相手のノードの位置を取得できた割合を図9に示し、自ノードのLRを検索して位置情報更新を行った割合を図10に示す。

図8,9の結果から方式Aに比べ、方式Bの方が、少ない通信コスト(60~80%減)であつ位置検索能力の劣化も少ないことがわかる。また、この効果はグループを構成しているノード数が大きいほど、顕著であり、位置検索成功率の差も小さくなることから、位置管理方式(4)を用いることにより、各ノードの位置管理の負荷を分担し、通信量の抑制がはかれていることから、要求条件(b)(c)を満たしている。また、検索成功率の劣化も大きくないことから要求条件(a)も同時に満たしている。

また、図10に示すように、自ノードLRを検索することにより25~45%のノードの位置を正しく捕らえている。したがって、ノードがネットワークから離脱している場合においても再接続の手がかりを得ることが可能である。したがって、要求条件(d)を満たしている。

以上の結果より、位置管理の分担を行うことにより通信、位置管理のコストを削減できることがわかった。つまり、提案方式は少数の固定位置管理サー

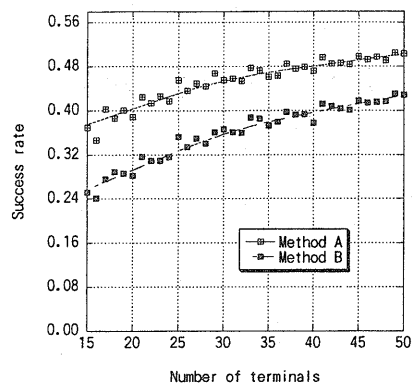


図10 位置検索成功率(他ノードLRを検索)

バを置く場合から、すべてのノードで位置管理を行う方法まで対応できるだけでなく、位置管理の分担状況に応じた、通信や位置管理コストで位置管理を行うことが可能である。

また、セッション管理においてはセッションIDを用いセッションを識別し、SMの属性にて、接続セッションのつながりを管理し、5.4節の手順を用いることで、セッションの構成を変更できるため、要求条件(e)~(g)を満足している。

7. まとめ

本論文では位置、サービス管理の枠組みを提案するとともに、位置管理方式に関して、その評価をした。また、AP間に構築されるセッションとノードをそれぞれ管理する位置管理とセッション管理を導入することで、それぞれのセッションにおいて、ノードの構成を動的に変化させられるように拡張を提案した。これにより、ノードの移動とともにサービスの移動も実現することが可能となる。

今後、プロトタイプの実装とともにセッション管理の方式について評価する。

[参考文献]

- [1] 片山, 高杉, 久保田, 小菊, "モバイル通信におけるサービス継続機構", 信学技報 SSE99-103, TM99-46, 1999.
- [2] 片山, 高杉, 久保田, サーバの移動性を考慮したモバイル環境, 信学会総合大会, B-7-171 2000.
- [3] 高杉, 片山, 久保田, サーバの移動性を考慮したモバイル環境における位置情報の管理, 信学会総合大会, B-7-171, 2000.
- [4] <http://www.sun.com/jini>
- [5] W. Adjie-Winoto, E. Schwartz, H. Balakrishnan, Jeremy Lilley, The design and implementation of an intentional naming system, Proc. 17th ACM SOSP, Kiawah Island, SC, Dec. 1999.