

村永 哲郎
MURANAGA Tetsuro

吉田 英樹
YOSHIDA Hideki

木村 哲郎
KIMURA Tetsuro

小型パソコン(PC)や携帯情報機器、PHSや携帯電話の普及により無線データ通信の基盤が整ってきたが、モバイルコンピューティング環境として使うには、通信速度が遅く不安定であり、小型化によりユーザーインタフェースが限定されるという問題がある。当社ではこれらの問題を解決するために、ソフトウェア技術 Move! (Mobile virtual environment) の一環として二つのミドルウェアの研究開発を行っている。FastMove! では既存の WWW (World Wide Web) ソフトウェアを使いながら、通信速度が遅いネットワークでの情報アクセスを高速化した。一方 PortableMove! では、自分が使うファイルをすべて携帯機器に入れて持ち歩き、外出先のデスクトップ機で使い慣れた利用環境を再現することで、ユーザーインタフェースの問題を解決した。

Mobile computing has become popularized with the widespread dissemination of small computers, portable devices, PHS and cellular phones. However, the mobile computing environment still has problems; namely, slow and intermittent communication links, and limitations on user interfaces imposed by size constraints.

To address these issues, we have developed the Move! (mobile virtual environment) middleware systems. FastMove! speeds up access to the World-Wide Web (WWW) via slow communication links, while using existing WWW software. PortableMove! enables users to obtain their personalized environment at a desktop host at any site, by serving all personal files stored in a portable device to the desktop.

1 まえがき

小型 PC や携帯情報機器が普及すると同時に、PHS や携帯電話を使った無線データ通信環境が整備され、“いつでも” “どこでも” オフィスにいるのと同じように作業ができるモバイルコンピューティングを実現する基盤が整いつつある。今後はビジネスマンのほか一般のユーザーにまで、モバイルコンピューティングのすそ野が広がることが予想される。

一方、モバイルコンピューティング環境では、オフィスの場合と比べて、いくつかの物理的な制約が課せられる。たとえば無線ネットワークは通信速度が遅く、しかも途中で通信がとぎれるなど不安定である。またオフィスでデスクトップ機を使うのと比べると、携帯機器では性能が限られ、ユーザーインタフェースの面からも使いにくい。このようにオフィスにいるときより、過酷な状況に置かれても、できる限り快適な利用環境を提供するソフトウェア技術の開発がキーポイントとなる。

ここでは、当社が開発しているモバイルコンピューティングのソフトウェア技術 Move! について述べる。当社は FastMove!, PortableMove! という二つのミドルウェアの研究開発を行っている。FastMove! は、既存の Web ソフトウェアをそのまま使いながら効率的かつ快適な WWW アクセスを可能とする。PortableMove! では、ふだん使うファイルをすべて携帯機器に入れて持ち歩き、煩雑な操作なしに外

出先のデスクトップ機で使い慣れた利用環境を再現する。

2 FastMove!: 快適なモバイル WWW アクセス

Web ブラウザ^(注1)の普及により、インターネットを介して世界中の最新情報へアクセスすることが容易になった。企業内の情報も Web 技術を使って取り出す方法が急速に浸透している。外出先から PHS や携帯電話を使って、対話的に WWW を利用する場合、通信回線の速度と品質、料金が大きな問題となる。

2.1 アーキテクチャ

FastMove! では、Web プロキシサーバ^(注2)によって独自の通信プロトコル(データ送受信のための手順や規則)と HTTP (HyperText Transfer Protocol: WWW アクセスの標準プロトコル)との相互変換を行い、既存の Web ソフトウェアをそのまま使った高速アクセスを実現する。このため低速回線の両側にそれぞれプロキシサーバを配する構成をとる(図1)。クライアント側プロキシは、クライアント(ブラウザ)に対しては Web サーバとして振る舞い、サーバ側プロキシは Web サーバに対してクライアントとして振る舞う。この二つのプロキシ間を独自の通信方式で結び、遅い回線での Web アクセス性能を向上させる。

(注1) WWW のサイトを閲覧、利用するためのソフトウェア。

(注2) 基の情報提供者の代わりに、情報提供をするサーバ。

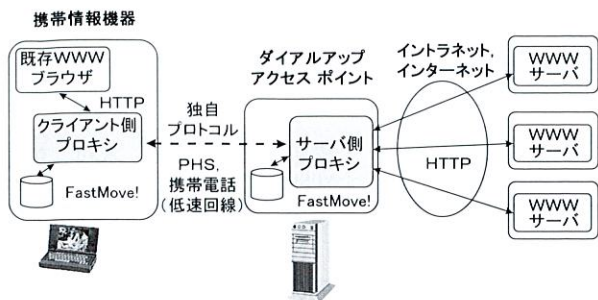


図1. FastMove!のアーキテクチャ クライアント側とサーバ側両方にプロキシを置き、プロキシ間を独自の通信方式で高速化する。
Dual-proxy architecture of FastMove!

2.2 高速化手法

- (1) HTTPの多重化 HTTPでは一つのWebデータを取り出すのに、一つのTCP (Transmission Control Protocol) 接続を確立する。複数のデータが含まれたページを取り出すのに、複数のTCP接続が必要となる(図2(a))。TCP接続の確立にはメッセージのやり取りが必要で、低速回線では時間がかかる。そのうえTCPは徐々に効率を上げて行くという特徴をもつため、接続直後は伝送性能がよくない。このため低速回線で複数のTCP接続を行うのは効率が悪い。FastMove!では、一つのTCP接続の上に、複数のHTTPを多重化して送信することにより、上記の問題を解決した(図2(b))。
- (2) 優先度付加 HTTP多重化を行う場合に、送信データに優先度を付加し、優先度の高さに応じて送信順序を変更する。サーバ側プロキシが先に送るべきであると判断したデータを優先的に送信することが可能となる。画像データは、大きさなどの情報が入ったヘッ

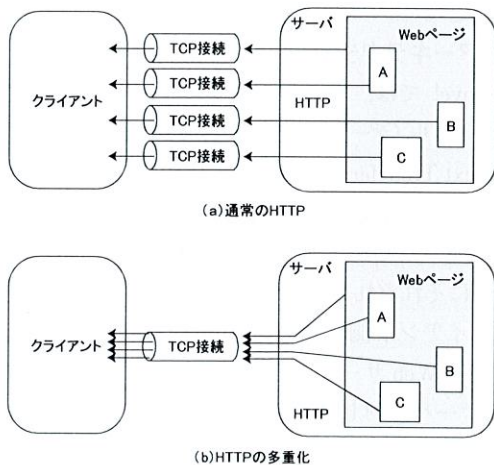


図2. HTTP多重化による高速化 一つのTCP接続上に複数のHTTPを多重化して送信することにより、通信効率を上げる。
Multiplexing HTTP streams over a TCP connection

ダ部分と実際の画像データ部分とからなる。ヘッダ部分および、解像度の低い大まかな部分の優先度を高くすることにより、ページ全体の大まかなイメージを早く利用者に提示できる。

- (3) プリフェッチ Webページには、他のページや画像データへのリンク情報が書かれており、リンク先のデータが次に要求される可能性が高い。そこで利用者がページを読んでいる間に、リンク先のデータを先読みして、クライアント側プロキシのキャッシュに格納する。このプリフェッチしたデータが、次に要求されれば即座にそれが提供される。

FastMove!では、上記の高速化手法に加えて、サーバ側プロキシで圧縮されていないデータを圧縮することにより、通信量を減らしている。現在の実装では、最大で40%程度の高速化を達成した。

3 PortableMove!: 使い慣れた利用環境の外出先での再現

3.1 個人利用環境の持ち運び

携帯情報機器は小型・軽量で持ち運びに優れているものの、画面やキーボードの大きさが制限され、文書作成など本格的な作業には向かない。そこで外出先のオフィスに利用できるデスクトップ型の計算機があるならば、そこに間借りしようという発想がPortableMove!の原点である。その際、携帯機器には“自分が使うものすべて”を入れておく。それは文書ファイルのようなデータだけでなく、GUI (Graphical User Interface) の設定 (たとえばメニューの内容、アイコンの配置情報など)、アプリケーションプログラムなどすべてを含む。いわば“個人利用環境”を持ち運ぶのである。

携帯機器には、ネットワーク対応のファイルサーバを載せる。自分のデスクトップ機のそばに携帯機器を置き、ネットワークを介してデスクトップ機から携帯機器内のデータにアクセスする(図3(a))。外出時は携帯機器を持って出

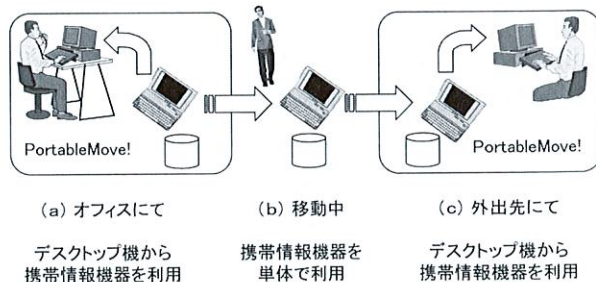


図3. PortableMove!: “個人利用環境”の持ち運び 文書ファイル、プログラム、GUI設定データなど、すべてを携帯情報機器に入れて持ち歩く。
Portable “user environment”

かけ、スケジュールを確認する、メールを読むなどの作業ができる(図3(b))。本格的に文書作成作業をするには外出先のデスクトップ機を借り、そこからネットワークを介して携帯機器内のファイルサーバにアクセスする(図3(c))。このときファイルサーバに置かれた個人用ファイルに基づき、自分がふだん使い慣れた利用環境を再現する。

PortableMove! の特長は、自分のオフィスの計算機と外出先とがネットワークでつながってなくても、ふだん使い慣れた環境を再現できること、そしてファイルの複製をもたないため、コピーしたり同期をとったりといった煩雑で時間のかかる手続きなしに、すぐに携帯機器をネットワークから切り離して持ち運べることにある。

3.2 PortableMove! の実現

PortableMove! の実現にあたって解決すべき課題は、外出先のデスクトップ機を使う際の設定をできるだけ簡便にすること、そして携帯機器内の情報を他人には見せないなど適切なアクセス制御を行うことである。

携帯機器はデスクトップ機に対して、NFS^(注3)(Network File System) サーバとして動作する。通常の NFS サーバと違うのは、接続するネットワークによりアクセス権の制御を変えることである。携帯機器内の利用者管理は、仮想的な利用者の識別子(identifier)に基づく。この仮想的な識別子を、外出先によって異なる実際の識別子にマップ(写像)することで、携帯機器内のファイルはつねに自分の所有物として使うことができる。この際、他の利用者には携帯機器内のファイルが使えないようなアクセス制御を行う(図4)。

携帯機器を外出先のネットワークにつなぐと、DHCP

(Dynamic Host Configuration Protocol) サーバから新しいアドレスが渡される。デスクトップ機には携帯機器を探す lookup 機能を載せる。デスクトップ機からログインすると、lookup 機能がこの利用者のファイルが載った携帯機器を探し、携帯機器のファイルシステムを自動的にマウントする。これによりデスクトップ機のファイルシステムの一部として、携帯機器のファイルが見える。そして携帯機器内にある個人用環境設定ファイルを読み込んで、利用者がいつも使っている環境をそのデスクトップ機上で再現できる。

4 あとがき

Move! に見るモバイル コンピューティングのためのソフトウェア技術をまとめると次のことが言える。

- (1) 適応性の追求 従来の分散システム技術に加え、遅く不安定なネットワーク、移動先により異なる利用環境といった過酷な状況に適応するソフトウェア技術が必要とされる。
- (2) クライアント/サーバモデルの拡張 ネットワークから切り離されても動けるよう、サーバからの自律性を確保するために、クライアントにもサーバ機能を載せるようなアーキテクチャが必須(す)である。
- (3) アプリケーション透過性の重視 アプリケーションプログラムに変更を加えなくても、モバイルコンピューティング環境でそのまま動くことが要求される。

以上をキーポイントとし、今後、FastMove! はいつその性能向上と機能強化を行う。PortableMove! は移動先のプリンタサービスなどを活用できるように拡張していく。

文献

- (1) 木村哲郎, 他. 移動先で同じユーザ環境を実現するための携帯ファイルサーバ, 第4回情報処理学会モバイルコンピューティング研究会, 2, 1998.

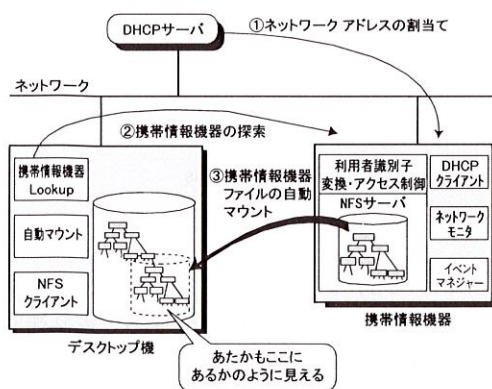


図4. PortableMove!の実現 携帯情報機器を外出先のデスクトップ機につなぐと、携帯機器内のファイルが自動的にデスクトップ機から使えるようになる。

Implementation of PortableMove!

(注3) NFSは、米国SunMicrosystems社の商標。



村永 哲郎 MURANAGA Tetsuro

研究開発センター 情報・通信システム研究所研究主務。モバイルコンピューティングなどの分散システムの研究開発に従事。情報処理学会, ACM, Usenix 会員。Communication & Information Systems Research Labs.



吉田 英樹 YOSHIDA Hideki

研究開発センター 情報・通信システム研究所。モバイルコンピューティングなどのネットワーク・基本ソフトウェアの研究開発に従事。情報処理学会, ACM, IEEE 会員。Communication & Information Systems Research Labs.



木村 哲郎 KIMURA Tetsuro, Ph.D.

研究開発センター 情報・通信システム研究所研究主務, 工博。モバイルコンピューティングなどの分散システムの研究開発に従事。情報処理学会, ACM, IEEE 会員。Communication & Information Systems Research Labs.