

## 推薦論文

## モバイル向けシンクライアントシステムの検討

高橋 竜男<sup>†</sup> 高橋 修<sup>††</sup> 水野 忠則<sup>†††</sup>

本論文では、モバイルコンピューティングの現状の課題として、携帯性と機能性を両立するクライアント端末が存在しないことをあげ、この解決策として、移動機を利用することを前提として設計した、新しいシンクライアント/サーバシステムを提案する。最初に、移動機を端末にするという観点、W-CDMA 網を介するという観点、モバイルコンピューティングにおける利便性という観点から、モバイルコンピューティングにおけるシンクライアントシステムに関する要件を抽出する。次に、既存のシンクライアントシステムに関して、これらの要件に対する充足度の評価を行い、その結果、要件をすべて満足しているものは存在しないことを示す。さらに、この問題を解決するために、既存方式を基礎に、GUI 仕様変換ゲートウェイの導入、移動機上での擬似 GUI の提供等のカスタマイズを行った、モバイル向けシンクライアントシステム方式を提案する。最後に、作成したプロトタイプの実装およびその評価結果について述べ、提案方式の有効性を実証する。

## A Study of a Thin Client System for Mobile Computing

TATSUO TAKAHASHI,<sup>†</sup> OSAMU TAKAHASHI<sup>††</sup>  
and TADANORI MIZUNO<sup>†††</sup>

This paper examines the current limitations placed on mobile computing by the client device; the mobility of the client device degrades performance and functionality. In order to solve this problem, we adopt the thin client/server approach. A new architecture is proposed that is designed around the use of cellular phones as thin client devices. In the system assumed here, thin clients (cellular phones) and the server communicate over a W-CDMA network. We describe the requirements of the thin client/server system from the viewpoint of the usability of mobile computing. Existing thin client systems are examined against these requirements, and the result is that no one of them satisfies all requirements. Our solution is a new thin client architecture for mobile computing. It includes a GUI conversion gateway and a GUI emulator on the cellular phone. Finally, we show the first implementation of our prototype system and demonstrate its efficiency from evaluation results.

## 1. はじめに

第3世代の移動通信網である W-CDMA の商用サービスが開始されたことにより、モバイルコンピューティングにおけるインターネットアクセス環境は飛躍的に向上している。

一方で、モバイルコンピューティング用の端末に関

しては、ノート PC、PDA、ブラウザ機能付き携帯電話（以下、本論文では単に移動機と称する）等、選択肢は増加しているものの、携帯性と利便性（機能、性能）の双方を両立しうる端末はまだ存在していない。たとえば、ノート PC は、機能面、性能面では優れるが、携帯性に関しては、他の端末に比較して、重量が重く、サイズが大きいという点、および利用者自身が移動しながら操作することが困難であるという点が問題となる。逆に PDA や移動機は、携帯性には優れるが、性能面では処理能力が低く、記憶容量等が小さい点が問題となり、機能面では利用可能なアプリケーション（本論文では、ワープロ、メーラ、表計算等、

<sup>†</sup> NTT ドコモネットワーク研究所

Network Laboratories, NTT DoCoMo, Inc.

<sup>††</sup> 公立はこだて未来大学システム情報科学部情報アーキテクチャ学科

Department of Media Architecture, School of Systems Information Science, Future University- Hakodate

<sup>†††</sup> 静岡大学情報学部情報科学科

Department of Computer Science, Faculty of Information, Shizuoka University

本論文の内容は 2002 年 7 月の DICOMO2002 にて報告され、MBL 研究会運営委員により情報処理学会論文誌への掲載が推薦された論文である。

一般的な応用ソフトウェアを意味する)の種類や機能が少ないという点が問題となる。

このような端末に関する問題を解決する手段として、シンクライアント/サーバコンピューティングモデル<sup>1),2)</sup>に基づくシステム(以下、本論文ではシンクライアントシステムと称する)を利用し、移動機からオフィス内のデスクトップ端末のアプリケーション、作業環境を遠隔操作するというアプローチが有望である。

シンクライアントシステムは、ストレージ機能やアプリケーションの実行機能等をサーバに集中し、その分端末(この場合の端末をシンクライアント端末と称する)に要求される機能、性能を軽減することによって、低コスト化を実現する技術である。このため、移動機のように簡易な端末をシンクライアント端末として利用することが比較的容易と考えられる。また、シンクライアント端末の場所や機種、および装置の違いにかかわらず、サーバの作業環境をシームレスに利用できるというメリットがある。

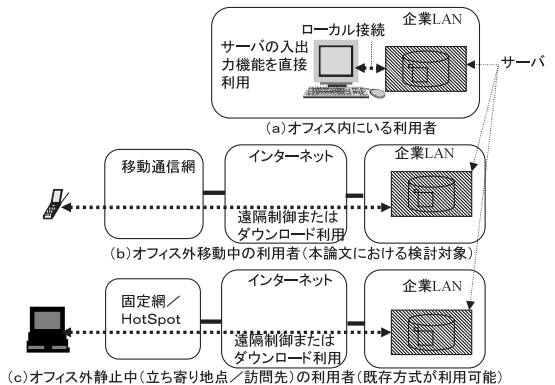
しかしながら、既存のシンクライアントシステムは、基本的に LAN 内において、PC と同等の HMI 機能 (Human Machine Interface: 端末の対利用者入出力機能) を有する端末で利用することを前提に設計されている。このため、モバイル環境に適用する際に生じる諸問題を明確にし、これらに対する解決策を検討することが重要である。

このような観点から、本論文では、最初にモバイル環境におけるシンクライアントシステムの要件を整理し、既存方式の問題点を明確にする。さらに、この問題を解決するモバイル向けシンクライアントシステムを提案し、試作したプロトタイプの評価により、提案方式の有効性を実証する。

## 2. モバイル向けシンクライアントシステムへの要件と課題

### 2.1 シンクライアントシステムの要件

本論文で前提とするシンクライアントシステムを利用したモバイルコンピューティング環境を図 1 に示す。オフィス内には企業 LAN に接続された PC が設置されており、利用者がオフィス内にいる場合には、この PC を直接操作する(図 1(a))。利用者がオフィスから外出している場合には、利用者はシンクライアント端末からネットワークを利用してオフィス内の PC(この場合オフィス内の PC はシンクライアントシステムにおけるサーバとして機能する)を遠隔操作することにより、オフィス内にいる場合と同一のアプリケー



(c) オフィス外静止中(立ち寄り地点/訪問先)の利用者(既存方式が利用可能)

利用者の状態(オフィス内、オフィス外移動中、オフィス外静止中)にかかわらず常に、同一のアプリケーション、作業環境を利用できる

図 1 シンクライアントシステムによるモバイルコンピューティングの概要

Fig. 1 Mobile computing environment using thin client system.

ションおよび作業環境を利用し続けることが可能になる。外出中の利用者は、自らの状態(移動中/静止中)において最も利用しやすい端末を利用する。

利用者が無線 LAN ホットスポットに立ち寄っている場合のように、比較的高機能な端末と高速低遅延なネットワークが利用可能な状態であれば、既存のシンクライアントシステムが利用可能である(図 1(c))。

本論文では、移動中の利用者に対しても、オフィス内に準じたモバイルコンピューティング環境を提供することを目標とし、移動機を端末としたシンクライアントシステムの構築に関して検討する(図 1(b))。

以下、本節では、移動機を端末とするという観点、移動網を介してサーバに接続するという観点、および利用者にとっての利便性の観点から、上記環境におけるシンクライアントシステムに対する機能面、および実用性に関する要件を抽出する。

#### (1) 機能面での要件

##### ● 汎用アプリケーションの利用

上記目標より、利用者は移動機をシンクライアント端末として利用する場合でも、オフィス内で PC を利用する場合と同一のアプリケーションを利用する。導入するシンクライアントシステムが、サーバ上のアプリケーションに関して、シンクライアントシステム専用のもの以外を利用できないという制約がある場合、利用者はシンクライアントシステム導入前にオフィス内の PC で利用していた汎用アプリケーションをリプレースする必要性を生じる。このことは、モバイル環境におけるシンクライアントシステム普及の大きな障害要因となる。

このため、提案するシンクライアントシステムでは、

PC 上の汎用アプリケーションがそのまま利用可能である必要がある。

- アプリケーション操作

利用者がシンクライアント端末として移動機を利用する場合と、オフィス内で PC を利用する場合とで、同じアプリケーションの操作が実行できる必要がある。このため、移動機 HMI により、主要なキーボード入力操作およびマウス操作に相当する操作（以下、本論文ではこれらを、それぞれキーボードエミュレーション、マウスエミュレーションと称する）が実行可能である必要がある。

- 端末非依存

利用者が状況に応じて端末を使い分けることを可能とするために、認証方式等において、利用可能な端末を限定しない方式を採用する必要がある。

- セキュリティ

シンクライアント端末は、インターネット経由で企業 LAN に接続する必要がある。このため、インターネットからサーバに対する不正操作を防止する強固な認証技術が必須となる。また、機密度の高い処理を実行する場合に対応するため、送受信データの暗号化が必要である。

(2) 実用性に関する要件

- 移動機 CPU における実用性

現在の移動機は、PC と比較すると、CPU クロック、利用可能メモリの点で大幅に劣る。このような制約の下でも、実用的な時間内でアプリケーションの操作が可能である必要がある。

- 移動機 HMI における実用性

アプリケーション操作の要件で述べたキーボード、マウスエミュレーションを実現することにより、デスクトップ端末で行う多くの操作を移動機の HMI から行うことが可能となる。しかしながら、モバイルコンピューティングにおいて利用頻度が高いと考えられる操作（本論文では、テキスト編集操作、プルダウンメニュー等による基本的なアプリケーションの制御、およびディレクトリ、ファイル制御のような基本的なサーバ OS の操作をあげる）に関しては、単に実行が可能であるというのみではなく、実用的な時間内に実行可能であることが必要となる。このためには、アプリケーションへの機能追加等により、移動機 HMI 向きの GUI (Graphical User Interface: 端末の HMI を利用して、グラフィカルな操作環境を提供するアプリケーションの入出力インタフェース) を提供する必要がある。

- 移動網を介した場合における実用性

本論文では、移動通信網として W-CDMA 網を利用することを前提とする。W-CDMA 網におけるパケットサービスは、帯域に比較して遅延が大きいことが知られている。このことは、TCP におけるスライディングウィンドウバッファ枯渇による実効スループットの低下と、ラウンドトリップタイム (RTT) 増大による、レスポンス低下という問題を生じうる。

実効スループット低下の問題に関しては、TCP プロファイルの調整によって解決可能である<sup>(3),(4)</sup>。しかしながら、RTT そのものを短縮することはできないため、上位プロトコルにおいて、シーケンシャルなパケット送受を頻繁に行う場合には、レスポンス低下等の問題が生じる<sup>(5),(6)</sup>。このため、移動網を介したシンクライアントシステムが実用的に利用可能であるためには、通信頻度を抑制し、RTT の影響を最小限にする必要がある。

## 2.2 既存方式の課題

### 2.2.1 既存方式の分類

既存シンクライアントシステムは、シンクライアント端末とサーバの機能分担の違いから以下の 2 方式に大別することができる。

- アプリケーション実行方式

本方式では、シンクライアント端末は、HMI 機能、アプリケーション実行機能を分担し、アプリケーションプログラムの保存管理機能、作業環境の提供をサーバが分担する。本方式によるシンクライアントシステムの例として NC<sup>(7)</sup> (Network Computer) があげられる。本方式では、Java で記述されたアプリケーションプログラムがサーバ上に格納されており、アプリケーション実行時にこれをシンクライアント端末にダウンロードする (図 2(a))。

- HMI 方式

本方式では、シンクライアント端末は、HMI 機能のみを分担し、アプリケーションプログラムの保存管理、作業環境の提供、アプリケーションの実行機能をサーバ側が分担する (図 2(b))。本方式によるシンクライアントシステムの例としては、MetaFrame<sup>(8)</sup>、WBT (Windows-based Terminal)<sup>(9)</sup>、VNC (Virtual Network Computing)<sup>(10),(11)</sup>、Rajicon<sup>(12)</sup> 等がある。

Java およびその他の Java を含む商標は米国 Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

MetaFrame は、Citrix Systems, Inc. の米国およびその他の国における登録商標です。

Windows2000, Windows-based Terminal は米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

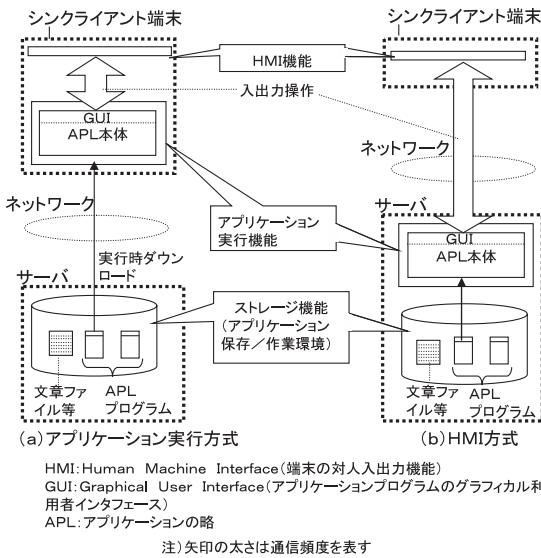


図 2 シンククライアントシステムの基本方式

Fig. 2 Basic architecture of existing thin client systems.

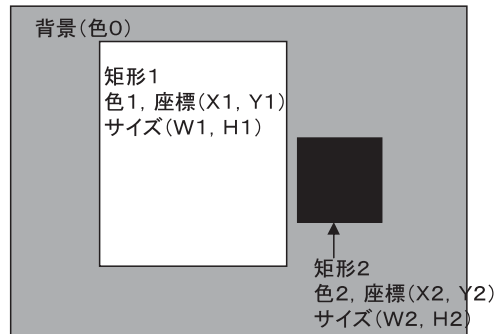
なお、次項において、上記のうち、本論文と関連の深い、VNC, Rajicon について概説する。

2.2.2 既存方式の例

(1) VNC

VNCは、主としてPCをシンククライアント端末として利用することを目的とした、HMI方式のシンククライアントシステムである。VNCでは、RFB( Remote Frame Buffer )と称する独自プロトコルにより、サーバのフレームバッファから画面情報を抽出し、これをシンククライアント端末側に転送することにより、シンククライアント端末の画面上でサーバの画面を再現する。また、シンククライアント端末上で行われたキーボード操作、ポインティングデバイス操作に関するイベント情報をサーバに転送し、サーバ上で該当操作を実行する。

また、RFBではRRE( Rise and Runlength Encoding ), CoRRE( Compact RRE ), Hextileと称する3種類の画面情報圧縮方式がサポートされている。RREはサーバの画面情報から、最も多く使われている色を抽出してこれを背景色とし、それ以外の色が使われている領域を色ごとに複数の矩形領域に分割する( 図 3 )。こうしてサーバの画面情報を、背景色( 色 )と、複数の矩形情報( 色, 座標, サイズ )で表すことが可能となり、モノトーンのデスクトップ上でいくつかのウィンドウを開いた場合等、画面構成が単純な場合において、特に高い圧縮率が期待できる。CoRREおよびHextileは、それぞれ、画面を1辺255ドット、16ドットの正方形に分割した後、個々の正方形ごと



背景色と、矩形1, 矩形2の色, 座標, サイズのみをシンククライアントに送れば、シンククライアント側で同じ画面を再現できる

図 3 RRE( Rise and Runlength Encoding )の概要  
 Fig. 3 Illustration of RRE (Rise and Runlength Encoding).

に、上記RRE圧縮を実行する方式である。画面分割により個々の矩形領域を表す情報量をそれぞれ、8bit, 4bitに短縮することが可能となるので、サーバ画面情報がより複雑な場合にも圧縮率を高くすることが可能となる。

さらに、RFBでは、サーバとシンククライアント間で双方のクリップボードの情報を交換する機能を有する。

(2) Rajicon

Rajiconは、本論文と同様に、移動機を端末として利用するシンククライアントシステムである。移動機側ではView ModeおよびInput Modeと称する2つの利用者インタフェースを有し、これらを適宜切り換えて、サーバの遠隔操作を実現する。View Modeでは、移動機画面上でのサーバ画面の再現を行う。また、移動機画面と、サーバ画面サイズの違いを考慮して、ズーム機能を有している。Input Modeでは、キーボードおよびマウスエミュレーション機能を提供している。

2.2.3 既存方式の評価

2.1節で述べた要件に関する、両方式の充足度を評価した結果を表1に示す。

表1より、アプリケーション実行方式は汎用アプリケーションの利用に関する要件、移動機のCPUにおける実用性の要件を満たすことができないことが分かる。

一方、HMI方式は、移動機HMIにおける実用性の要件、および移動網を介した場合の実用性の要件を満たすことができない。2.2.2項で述べたRajiconは、ズーム機能を提供することにより、移動機HMIの問題に関して一定の考慮が行われているが、キーボードおよびマウスエミュレーションのみによる操作環境は他のHMI方式シンククライアントシステムと基本的に

表 1 既存シンクライアントシステムのモバイルコンピューティング向け要件の充足度評価

Table 1 Evaluation of existing thin client systems from requirements for mobile computing.

要件分類		アプリケーション実行方式	HMI方式
機能面	汎用アプリケーションの利用	×:Javaで記述されている等、シンクライアント端末上で実行することを前提に作成された専用アプリケーションが必要であり、既存のPC用の汎用アプリケーションをそのまま使うことはできない	○:アプリケーション自体は、サーバ上で実行される形態のため、既存のPC用のアプリケーションを原則そのまま利用することが可能となる
	アプリケーション操作	△:移動機HMI上で、キーボード、マウスをエミュレートし、これをアプリケーションに対する操作に変換する機能が必要	△:移動機HMI上で、キーボード、マウスをエミュレートし、これをサーバ上のアプリケーションに対する操作に変換し、転送する機能が必要
	端末非依存	○:方式自体では、端末を限定する要素は無い	○:方式自体では、端末を限定する要素は無い
	セキュリティ	ー:別途対応	ー:別途対応
実用性	移動機CPUにおける実用性	×:シンクライアント端末上でアプリケーションを実行するため、PCと同程度の処理能力が必要となる	○:アプリケーション自体はサーバ上で実行され、シンクライアント端末は入出力のみを分担するため、負荷は非常に軽い
	移動機HMIにおける実用性	△:専用アプリケーションにおいて、移動機に対応したGUIを提供する必要がある	×:PC用に設計された汎用アプリケーションのGUIをそのまま利用する必要がある
	移動網を介した場合における実用性	○:アプリケーションプログラムのダウンロード時のみ遅延が影響するが、アプリケーション利用時は影響しない	×:アプリケーションに対する入出力操作毎に移動網の遅延が影響し、操作時間が増大する

は同等であり、2.1 節(2)で述べた、本論文における移動機 HMI における実用性の要件を満たしてはいないと考える。

上記より、2.1 節で述べた本論文の目標を達成するためには、新たな方式を検討する必要がある。

### 3. モバイル向けシンクライアントシステムの提案

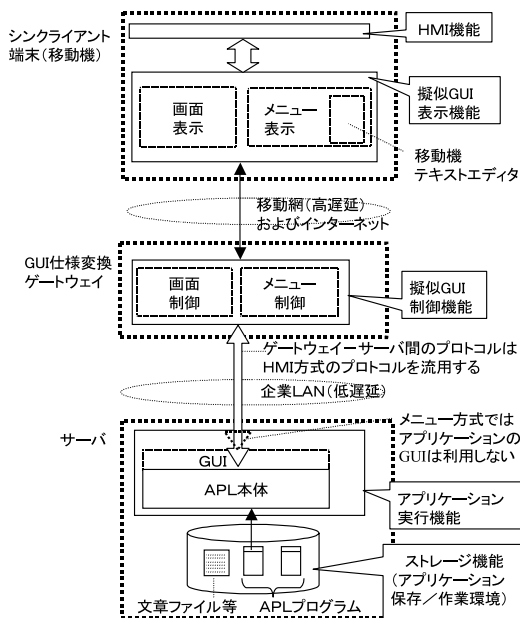
#### 3.1 提案方式

本論文で提案するモバイル向けシンクライアントシステムを図 4 に示す。提案方式は、図 2 に示した HMI 方式を拡張し、シンクライアント端末(移動機)側に、本来アプリケーションの一部である GUI 表示機能をエミュレートする機能を追加することを特徴とする。また、アプリケーションの一部を移動機上で実行するという意味において、アプリケーション実行方式と HMI 方式を融合する方式と見なすこともできる。移動機側が提供する GUI を擬似 GUI と称する。

アプリケーション自体の実行および保存管理、作業環境の提供は、HMI 方式と同様にサーバが分担する。これにより、汎用アプリケーションの利用が可能であること、移動機 CPU における実用性があることは、HMI 方式と同等の利点となる。

さらに、提案方式では、擬似 GUI を提供することにより、HMI 方式における、移動機 HMI における実用性がない、移動網を介した場合の実用性がないという問題を解決している。この理由を以下に述べる。

現在の移動機は、移動機画面上に簡易な GUI を構築するための利用者インタフェース(UI 部品)を備えている。提案方式ではこの UI 部品を利用して擬似 GUI を構築する。UI 部品は移動機 HMI における利



HMI: Human Machine Interface(端末の対人入出力機能)  
 GUI: Graphical User Interface(アプリケーションプログラムのグラフィカル利用者インタフェース)  
 APL: アプリケーションの略 (注) 矢印の太さは通信頻度を表す

図 4 提案するモバイル向けシンクライアントシステム  
 Fig. 4 Proposed thin client system for mobile computing.

用を前提に設計されているため、移動機 HMI における実用性に関する要件を満たすことができる。

また、2.1 節(2)で述べた移動機 HMI における実用性が求められる処理のうち、HMI 方式において移動網の遅延の影響を最も受けるのがテキスト編集処理である。特にサーバの仮名漢字変換機能(FEP)を利用した日本語の文章編集処理の場合、正しく漢字変換されたことを利用者自身が確認しなくてはならないため、上位プロトコルの改造のみで対応することはでき

ない。

提案方式では、擬似 GUI を構成する UI 部品から移動機のテキストエディタを呼び出すことにより、移動機の FEP を利用した文章編集が可能となるため、サーバ上の FEP を利用する必要がなくなり、通信頻度を削減できる。これにより高遅延な移動網を介した場合の実用性に関する問題を解決することが可能となる。本擬似 GUI 実現方式をメニュー方式と称する。

しかし、たとえば GUI 操作の中でもグラフィックオブジェクトの操作等、UI 部品でエミュレートすることが困難なアプリケーション操作や、アニメーション表示等に関しては、メニュー方式では対応することは困難である。このために、アプリケーション操作に関する要件を満たすことができなくなるという新たな問題が生じる。この問題を解決するため、移動機のグラフィック描画機能を利用して、移動機上にサーバの画面イメージを再現し、これを利用した操作環境を提供し、メニュー方式と併用する必要がある。これを画面方式と称する。両方式は、利用者が操作内容の種別に応じて、適宜切り換えることが可能である。

なお、擬似 GUI の操作をアプリケーションの操作に変換する機能が必要となるが、移動機の負荷を軽減するために、この機能は移動機とサーバの中間に設置された GUI 仕様変換ゲートウェイ（以下、単にゲートウェイ）が分担することとする。

また、ゲートウェイ-サーバ間のアプリケーション層通信プロトコルには、HMI 方式に属する既存の PC 用シンクライアントシステムのプロトコルをそのまま利用する。ゲートウェイがサーバに対して行う、コマンド/システムコール実行要求、情報転送、サーバ画面情報の取得、マウスイベントの実行要求は、上記既存プロトコルの機能を利用する。

擬似 GUI とセキュリティ実現方式の詳細については次節以降で述べる。

## 3.2 擬似 GUI の実現手法

### 3.2.1 メニュー方式

#### (1) 特徴

メニュー方式では、サーバ上のアプリケーションに対するプルダウンメニューによる制御や文章編集、サーバに対するディレクトリ制御、サーバのキーボードエミュレーション等をラベル、ボタン、テキストボックス等の UI 部品に対応させることによって、移動機画面上に、独自のアプリケーション操作環境（メニュー）を構築する。利用者は、ラベルやボタンの標題を参照して、所望の操作に対応する UI 部品を選択し、ボタン押下等、該当する UI 部品に規定されたイベントを

発生させ、選択された UI 部品の識別子をゲートウェイに送信する。

本方式により、移動機 HMI に適した擬似 GUI の提供、および入出力情報量の大幅な削減が可能となり、実用的な操作時間で利用することが可能となる。

#### (2) 基本制御方式

メニュー方式は、移動機のメニュー表示機能とゲートウェイのメニュー制御機能で実現される。メニュー制御機能部は、メニューを構成する UI 部品種別とその識別子、および当該 UI 部品が選択された場合に実行するスクリプトを管理している。スクリプトには、サーバまたはサーバ上のアプリケーションで実行する一連の操作（システムコール、コマンド実行等）、もしくはサブメニューの識別子が記述されている。これらの情報をメニュー情報と称する。メニュー情報はアプリケーション別に管理されている。

メニュー情報を利用したアプリケーションの制御方式を説明する（図 5 参照）。

①利用者がメニュー表示を要求すると、メニュー制御機能部は、実行中のアプリケーション名を取得するためのシステムコールを生成し、これをサーバに対して実行要求する。その結果、当該アプリケーションの名称を取得し、対応するメニュー情報を検索する（図 5(a)）

②メニュー制御機能部は①で取得したメニュー情報中の、UI 部品の種別、識別子を移動機側に送信し、対応スクリプトを内部で管理する（図 5(b)）。

③移動機のメニュー表示機能部は、取得した UI 部品情報に基づき移動機画面上にメニューを表示する（図 5(c)）

④利用者が任意のメニューを選択すると、該当する UI 部品の識別子がメニュー制御機能部に送信される。メニュー制御機能部は、取得した識別子に対応するスクリプトを実行する。スクリプト中にサーバ上のアプリケーションに対するコマンドがあった場合、これをサーバに実行要求する（図 5(d)）。

#### (3) メニュー情報の作成方法

新規のアプリケーションを利用するためには、利用者はメニュー情報を作成し、これをメニュー制御機能部に登録する必要がある。メニュー情報は UI 部品種別と対応するコマンド、メニューのネスト構造等を定義することによって、容易に作成可能であるため、2.1 節(1)で述べた汎用アプリケーションの利用に関する要件を損なうことはないと考えられる。

これに対して、サーバ内の特定ディレクトリのファイル一覧をメニュー表示する場合には、随時変化する



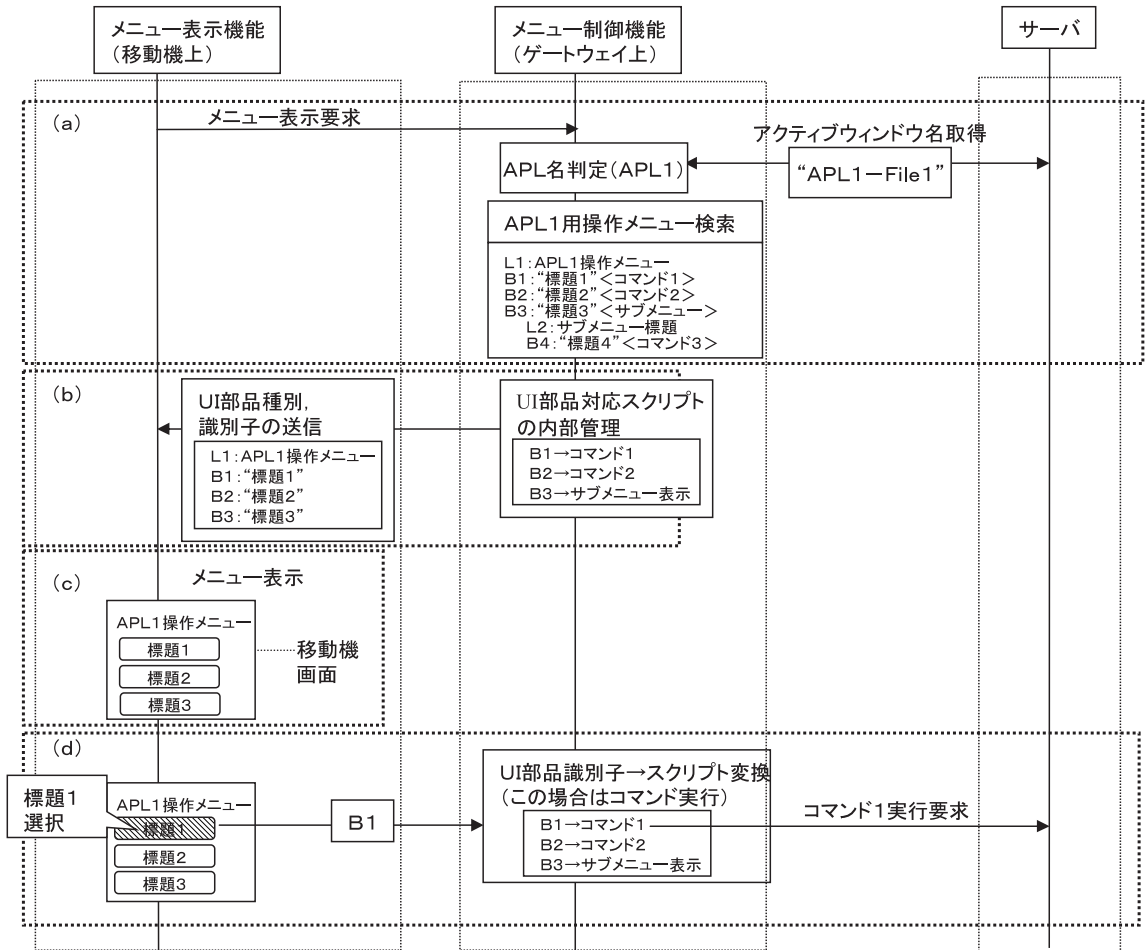


図5 メニューによる制御方式  
Fig. 5 Example of the menu mode sequence.

ファイル情報を正確に表示するために、メニュー情報を自動的に生成する必要がある。この方法を以下に示す。

①利用者が特定ディレクトリのファイル一覧の取得を要求すると、メニュー制御機能部は、当該ディレクトリのファイル一覧を取得するためのシステムコールを生成し、これをサーバに実行要求する。その結果、当該ディレクトリを構成するファイル、サブディレクトリの一覧を取得する。

②ファイル、サブディレクトリに対する操作は、起動、削除、サブディレクトリへの移動等に限定することができる。このため、メニュー制御機能部は①で取得したファイル名等から該当ファイル等に対して実行可能な操作を行うUI部品と、コマンド、システムコールを自動生成する。これを取得したすべてのファイル等に対して行うことにより、ファイル制御のメニュー情報を作成する。

メニュー情報作成後の動作は(2)の②以降と同様である。

(4) アプリケーションの文章データの抽出

メニュー情報のUI部品としてテキストボックスを利用することによって、移動機の日本語編集機能を利用することは可能である。アプリケーションの文章編集を行うためには、これに加えて、サーバ上のアプリケーションから編集対象個所の文章データを抽出し、移動機上で編集を行った後、これをサーバに書き戻す機能が必要となる。

このため、メニュー制御機能部は、サーバ上のアプリケーションから文章データを抽出する機能を有する。また、メニュー情報に文章データ抽出要求のためのオプションを追加する。

メニュー制御機能部は、上記オプション付きテキストボックスを検出した場合には、1段落分の文章データを選択し、複製するコマンドを生成し、これをサー

パに実行要求する．さらに，ゲートウェイサーバ間の情報転送機能を利用して複製した文章データをゲートウェイに読み込む．メニュー制御機能部は読み込んだ文章データを，テキストボックスの初期値として移動機に渡す．

移動機のテキストエディタを利用して編集された文章データは，利用者が書き戻しを要求することによってゲートウェイに転送される．メニュー制御機能部は情報転送機能を利用してこの文章データをサーバに返送し，さらに，上書きコマンドの実行要求を行うことにより，アプリケーションの元の文章データを編集後の文章データで上書きする．

上記より，移動網の高遅延性の影響を受ける機会を文章データ転送時のみに限定することができる．

### 3.2.2 画面方式

#### (1) 特徴

画面方式は，移動機が有するグラフィック描画機能を利用して，移動機画面上にサーバ画面を復元し，その上でエミュレートされたマウスポインタ(擬似ポインタ)による入力操作機能を提供する．本機能により，サーバ上のアプリケーションにおける，アニメーション表示や，グラフィックオブジェクトの操作，およびコマンドインタフェースがないアプリケーションへの対応が可能となり，提案方式はアプリケーション操作に関する要件を満たすことが可能となる．HMI方式と類似しているが，後述するように，移動機HMIを考慮した画像情報の加工による通信頻度の削減や，情報量の削減による画面表示高速化を行っている点が異なる．

#### (2) 基本制御方式

本方式は，移動機の画面表示機能とゲートウェイの画面制御機能で実現される．画面制御機能部では，画面情報生成/加工と，擬似ポインタの座標変換を行う．

画面情報生成処理において，画面制御機能部はサーバ画面の取得要求を行い，取得した画面情報に対して，ズーム処理，および移動機の画面の仕様(色数，ドット数)に合わせた変換を行い，移動機画面上における描画要素(矩形情報の集合)として，移動機に転送する．移動機の画面表示機能部では，当該情報に基づいた描画処理を行うことによって，サーバ画面を再現する．

擬似ポインタは画面表示機能部によって，復元されたサーバ画面上に表示され，移動機のダイヤルボタンを利用して制御することが可能である．マウスイベントが要求された時点で，擬似ポインタの座標とイベント種別がゲートウェイの画面制御機能部に送信される．画面制御機能部は，移動機座標をサーバ画面の座標に

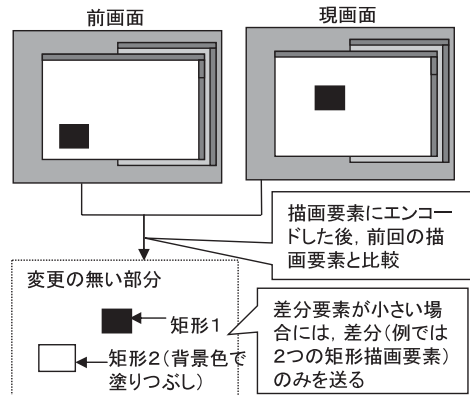


図6 画面方式におけるレスポンス向上のための情報量削減手法  
Fig.6 Data compression method for reduce response time of the picture mode.

変換した後，サーバ上の該当座標へのマウスイベントの実行要求を行う．

#### (3) 画面表示高速化手法

画面制御機能部は画面情報生成後，前回表示画面との差分情報転送を行うことによって，送受する描画要素情報量の削減とレスポンスの向上を行う(図6)．

①画面制御機能部は，前回取得した描画要素を保持する．

②現描画要素を取得したとき，画面制御機能部は，取得した描画要素と，前回の描画要素の差分が一定量以下である場合，両者の差分に関する描画要素を作成し，当該情報が差分情報である旨を識別するフラグを描画データに付与して，移動機の画面表示機能部に送付する．

③移動機の画面表示機能部では，受信した描画要素が差分情報フラグを有する場合には，現在表示している画面に，送付されてきた描画要素を上書きすることにより，サーバ画面を再現する．

#### (4) 移動機HMIの効率的利用方式

移動機とサーバの画面の形状，サイズの違いから，移動機上ではサーバ画面の一部分のみを表示し，表示領域を切り換える必要がある(図7(a))．また，サーバ上のアプリケーションが表示するウィンドウ自体も，スクロールバーによる表示位置切換え機能を有している．このため，利用者は，移動機上の表示領域のシフトと，アプリケーション自体のスクロール処理とで2段階のスクロール処理を行う必要がある．このことが利用者に混乱を招いたり，無用な画面更新処理を招いたりする可能性がある．

また，移動機の小さい画面を有効利用するためには，ウィンドウの背景等 unnecessary 部分を排除し，制御対象



表 2 認証方式の比較  
Table 2 Comparison of authentication algorithms.

評価項目	CHAP <sup>15)</sup>	ワンタイムパスワード <sup>16)</sup> (RFC2289)	ワンタイム パスワード <sup>17)</sup> (時間同期)	デジタル証明書 <sup>18)</sup>
利用者を認証する手段	パスワード	パスワード*	認証トークンが生成する乱数+パスワード	デジタル証明書+パスワード(パスワードはオプション)
安全性と利用者負担軽減の両立可否	×: 安全性を確保するためにはパスワードの複雑化、頻繁な変更が必要となる	×: 安全性を確保するためにはパスワードの複雑化、頻繁な変更が必要となる	○: 認証トークンと併用するため、パスワード管理に関する利用者の負担は軽減される	○: デジタル証明書と併用するため、パスワード管理に関する利用者の負担は軽減される
端末非依存(任意の端末から利用可能か否か)	○: 端末限定無し	○: 端末限定無し	○: 端末限定無し	×: クライアント証明書がインストールされている端末からのみ利用可能
移動機CPUにおける実用性	△: ハッシュ関数の演算が必要	△: ハッシュ関数の演算が必要	○: 端末による演算は不要	×: 公開鍵暗号の演算が必要

CHAP: Challenge Handshake Authentication Protocol

\* 認証時にサーバから種とシーケンス番号が送信されるため、クライアントのパスワードを推定することにより、ワンタイムパスワードを算出できる。

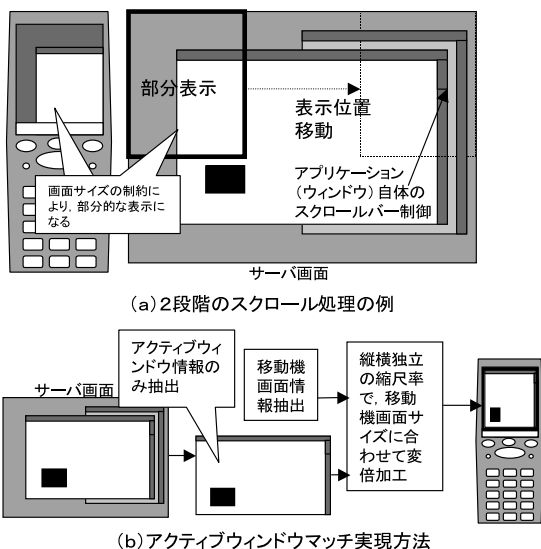


図 7 画面方式におけるアクティブウィンドウマッチの必要性と実現方法

Fig. 7 The needs and the method of the active window mach in the picture mode.

のウィンドウ(以下アクティブウィンドウと称する)のみを表示することが望ましい。

上記の 2 つの課題を同時に解決するために、アクティブウィンドウ全体が、移動機画面全面に表示されるように、表示位置および、拡大率を自動調節する方式が有効である。本論文では、この機能をアクティブウィンドウマッチと称する。この実現方法を以下に述べる(図 7(b))。

①画面制御機能部は、サーバ上のアクティブウィンドウの名称、座標、およびサイズを取得するシステムコールを生成し、これをサーバに実行要求して、その結果を取得する。次に、現在利用中の移動機( http リ

クエストヘッダに埋め込まれた識別子から判定)の画面サイズから、移動機画面全体にアクティブウィンドウが表示されるように、画面の拡大/縮小率、および切り出し座標を算出する。

②画面制御機能部はサーバに、画面情報の取得を要求し、取得した情報から上記部分の切り出し等の加工を行い、加工後の描画要素を移動機に送付する。このとき、移動機画面の座標とサーバ画面の座標のマッピング情報も加工後の画面を基に更新する。

③移動機の画面表示機能部は加工された描画要素に基づき描画処理を行う。

### 3.3 セキュリティ

#### (1) 盗聴防止

現在、国内キャリアが提供している移動機においては、インターネットアクセス時に利用可能なパケット暗号化プロトコルは SSL<sup>13)</sup>のみであるので、シンクライアント端末-ゲートウェイ間での盗聴防止が必要な場合には SSL を利用する。また、アプリケーションの修正が不要という点および、多くのサーバ用 OS で標準搭載されているという点において優れているため、ゲートウェイ-サーバ間での盗聴防止が必要な場合には IPsec のトランスポートモード ESP<sup>14)</sup> を利用する。

#### (2) 認証方式

インターネット経由のクライアント認証に利用可能な方式として、現在広く利用されているものを表 2 に示す。このうち、CHAP<sup>15)</sup>と RFC2289<sup>16)</sup>は利用者を認証する手段がパスワードのみであるのに対し、時間同期式のワンタイムパスワード<sup>17)</sup>、デジタル証明書<sup>18)</sup>は、それぞれ認証トークンが生成する乱数、デジタル証明書を認証手段とし、さらに認証トークン、デジタル証明書が第三者に利用されることを防ぐために、パスワードを併用する方式である。

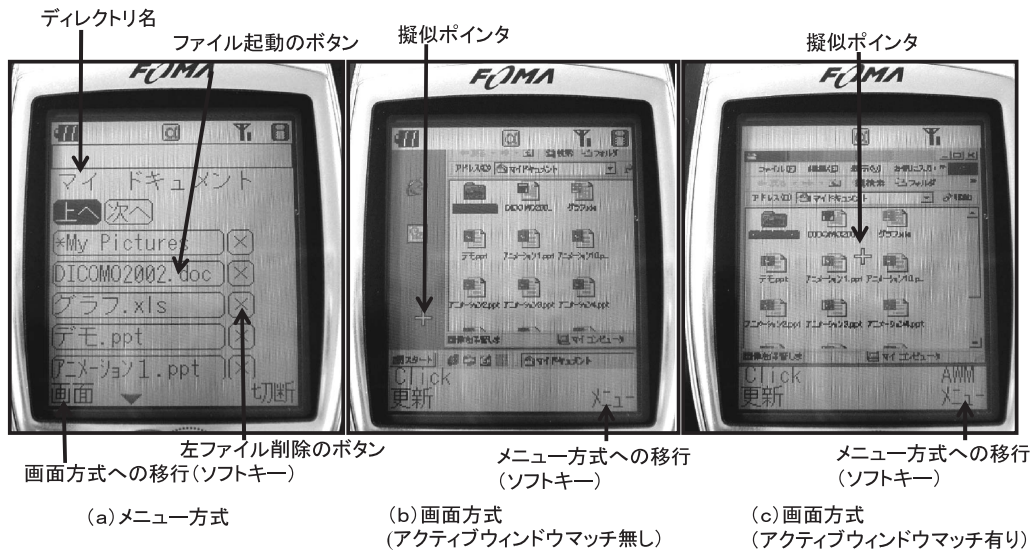


図 8 メニュー方式，画面方式擬似 GUI の比較

Fig. 8 Comparison of the menu mode and the picture mode GUI emulations.

認証する手段がパスワードのみの方式では，辞書攻撃等によるパスワードの推定や漏洩からサーバを防御するため，パスワードを複雑化したり，周期的に変更したりする必要があり，利用者のパスワード管理の負担が大きいことが問題となる．一方，デジタル証明書を利用する方式では，この問題はないが，デジタル証明書がインストールされている端末のみで利用可能なこと，端末側で処理量の大きい公開鍵暗号の演算を行わなければならないことから，2.1 節で述べた端末非依存，および移動機 CPU における実用性の要件に反するという問題を生じる．

このため，本論文では，これらの問題が生じない，時間同期式のワンタイムパスワードを採用する．

#### 4. 実 装

移動機として FOMA 端末を利用し，提案方式の評価を行うためのプロトタイプを作成した．移動機側の擬似 GUI 表示機能は i アプリ (DoJa1.0 および 2.1 仕様) を利用して実装し，ゲートウェイ側の擬似 GUI 制御機能は JSP (JavaServer Pages) を利用して実装した．メニュー情報は XML (eXtensible Markup Language) によって定義し，JAXB-API<sup>19)</sup> により Java オブジェクトに変換する．

また，ゲートウェイ-サーバ間のプロトコルとしては 2.2.2 項で述べた VNC を利用した．3.1 節および 3.2 節で述べたコマンド/システムコールの実行要求，

およびマウスイベントの実行要求は VNC の基本機能を利用し，さらに情報転送機能には，VNC のクリップボード転送機能を利用した．サーバの画面情報取得処理に関しては，VNC サーバで Hexstex 圧縮を行い，ゲートウェイは，移動機の画面仕様に合わせた形式にこれを変換する処理のみを分担することとした．

パケット暗号化は，SSL 対応の JSP を利用することにより実現した．利用者認証には，SecureID を利用した．利用者はゲートウェイにログインする際に，ユーザ ID とともに，認証トークンに表示されている乱数とパスワードを入力し，ゲートウェイに送信する．ゲートウェイは，SecureID の認証エージェント機能を有し，利用者のログインの可否を判定する．なお，利用者の利便性を考慮し，認証はゲートウェイに接続する際に 1 回のみ行い，個々のサーバにログインする際には，パスワードの入力を不要とした．このため，ゲートウェイが管理する利用者情報に当該利用者が利用できる VNC サーバ名とパスワードを登録し，ゲートウェイ認証の成功時に，自動的に VNC サーバ名とパスワードが設定される方式とした．

なお，i アプリサイズは，DoJa1.0 用で約 9.4 KB，DoJa2.1 用で，約 10.2 KB になった．

ディレクトリ内のファイル一覧表示に関する，メニュー方式とおよび画面方式 (アクティブウィンドウマッチなしおよびあり) 擬似 GUI 表示の比較を，それぞれ，図 8 (a)，(b)，(c) に示す．

FOMA，i アプリは株式会社 NTT ドコモの登録商標です．

SecureID は，RSA Security, Inc. の登録商標です．

5. 評価結果と考察

提案方式の効果を評価するために、日本語文章編集処理、アプリケーション終了、アプリケーションウィンドウスクロール、アニメーション表示に関して、それぞれ提案方式と従来方式による操作時間の比較を行った。また、擬似ポインタ操作に関しては、提案方式による操作時間を測定した。

評価実験システムの構成を図9に示す。評価のための操作は2人の被験者によって行った。図9における各装置の仕様および被験者の簡単な経歴を表3に示す。

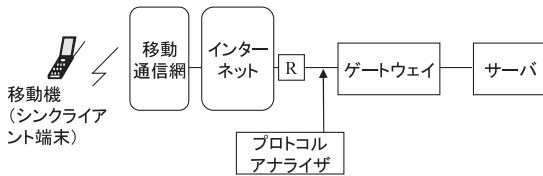


図9 評価実験システムの構成

Fig.9 Experiment environment for evaluation.

す。また、本評価では、プロトコルアナライザによりパケットの内部を観測するため、SSL および IPsec による暗号化は行っていない。

なお、従来方式の評価には、擬似ポインタ機能およびキーボードエミュレーションによる入力とレスポンス画面の取得のみをサポートする試験用シンクライアントシステムを作成し、これを利用した。実験系および被験者に関しては、提案方式評価時と同一である。評価結果を表4に示す。

5.1 メニュー方式の評価

(1) 日本語文章編集処理に関する評価

日本語文章編集作業の評価は、図10に示す文章データを利用し、漢字2文字“近年”を削除し、2文字“最近”を挿入するに要する時間を測定した。従来方式の評価においては、キーボードのエミュレーションにより、サーバ側の仮名漢字変換機能を遠隔操作して、上記編集作業を行った。編集に要したキーボード操作は、2文字削除要求 → キー入力 (saikinn) → 変換実行 → 変換候補表示要求 → 変換候補選択 → 確定

表3 評価用装置の仕様と設定

Table 3 Specifications and configurations of equipments for evaluation.

項目	仕様等
評価実験用装置の仕様	ゲートウェイ CPU: Intel® Xeron™ (2.40GHz), Cache: 512KB, Memory: 1GB OS: Microsoft® Windows 2000 Server
	サーバ CPU: Intel® Pentium III (800MHz), Cache: 256KB, Memory: 256MB OS: Microsoft® Windows 2000 Server 画面: 800×600ドット、256色
	移動機 P2101v
被験者経歴	A 30代男性、評価対象アプリケーション(PC上)利用経験約5年、携帯電話版シンクライアント評価作業経験約1年
	B 30代男性、評価対象アプリケーション(PC上)利用経験約5年、携帯電話版シンクライアント評価作業経験約3ヶ月

表4 評価結果

Table 4 Evaluation results.

評価項目	測定条件			評価結果			
	操作内容(概要)	アプリケーション種別	アプリケーションウィンドウサイズ(ドット)	被験者	操作に要した時間(秒)		提案方式による操作時間短縮率(%)
					従来方式(HMI方式相当)	提案方式	
1	日本語文章編集(2文字削除2文字挿入)	Microsoft® Word2002(図10)	最大	A	111.4	19.1	82.9
				B	136.4	27.8	79.6
2	アプリケーション終了		最大	A	26.8	8.6	67.9
				B	40.2	8.3	79.3
3	アプリケーションウィンドウスクロール		791×563	A	39.2	27.3	30.4
				B	44.3	31.3	29.2
4	アニメーション表示	Microsoft® Window日付と時刻の調整プロパティ(図12)	176×170	自動	5.4	2.0	63.0
5	擬似ポインタ操作	Microsoft® PowerPoint2002(図13)	最大	A	—	17.4	—
				B	—	22.3	—



この部分”近年”を削除し,”最近”に変更する

#### 文章データの詳細

- ・頁数:3, 全文字数:5610,
- ・図形オブジェクト:2, 表:2
- ・編集対象段落の文字数:345
- ・編集対象段落の次の段落の文字数:72
- ・編集対象文字数:2

図 10 評価対象文章データ (画面モード)

Fig. 10 Text data for evaluation (picture mode).

である。

提案方式の評価においては、メニュー制御により、該当段落のテキスト情報を移動機に転送し、移動機上で上記編集を行い、編集後のテキスト情報をサーバに書き戻すまでの時間を測定した。

このときの移動機の受信データ量と、操作内容、操作時間の関係を図 11 に示す。ただし受信データ量はプロトコルアナライザの測定結果から TCP のシーケンス番号差分を取得し、算出した。

表 4 の評価項目 1 より、被験者 A, B とも提案方式により操作時間を 80% 程度削減し、19~28 秒で上記編集操作を完了することが可能となった。この結果、一定の実用性を有するに至ったと考える。

#### (2) GUI 操作に関する評価

GUI 操作時間に関する評価は、図 10 で示した文章データを利用し、GUI 操作により該当ファイルを閉じる (アプリケーションを終了する) までの時間を測定した。

従来方式の評価では擬似ポインタを利用してサーバ上のアプリケーションの GUI を遠隔制御することにより、上記処理が完了するまでの時間を測定した。

行った GUI 操作は、アプリケーションのプルダウンメニューに対する“ファイル”→“終了”の選択 (クリック) と、この後表示される確認ダイアログで、“はい”を選択する (クリック) 操作である。最初の操作を行ってから、アプリケーションの終了を移動機上で確認するまでの時間を測定した。提案方式の評価では、移動機上のメニュー制御により、アプリケーションの終了を要求し、その終了を移動機上で確認するまでの時間を測定した。

表 4 の評価項目 2 より、提案方式により、被験者 A は約 70%、被験者 B は約 80% 操作時間を短縮し、8~9 秒程度で上記操作を完了することが可能となり、一定の実用性を有するに至ったと考える。

#### 5.2 画面方式の評価

##### (1) 移動機 HMI の効率的利用方式に関する評価

本評価においては、図 10 に示した文章データにおいて、アプリケーションウィンドウの表示位置を 1 画面スクロールするのに要する時間を測定した。

従来方式の評価では、画面左端を表示した状態で、表示領域をシフトし画面右端を表示後、擬似ポインタでスクロールバーをクリックすることにより、1 画面スクロールし、表示領域を画面左端に戻すまでの時間を測定した。提案方式の評価では、通常画面 (画面左端を表示) からアクティブウィンドウマッチ画面に切り換え、同様に擬似ポインタを利用して、スクロールバーをクリックすることにより 1 画面スクロールを行った後、通常画面 (画面左端を表示) に戻すまでの時間を測定した。

表 4 の評価項目 3 より、提案方式により、被験者 A, B とも操作に要する時間を約 30% 程度削減し、約 30 秒で上記操作を完了できるに至ったことが分かる。

##### (2) 画面表示高速化手法に関する評価

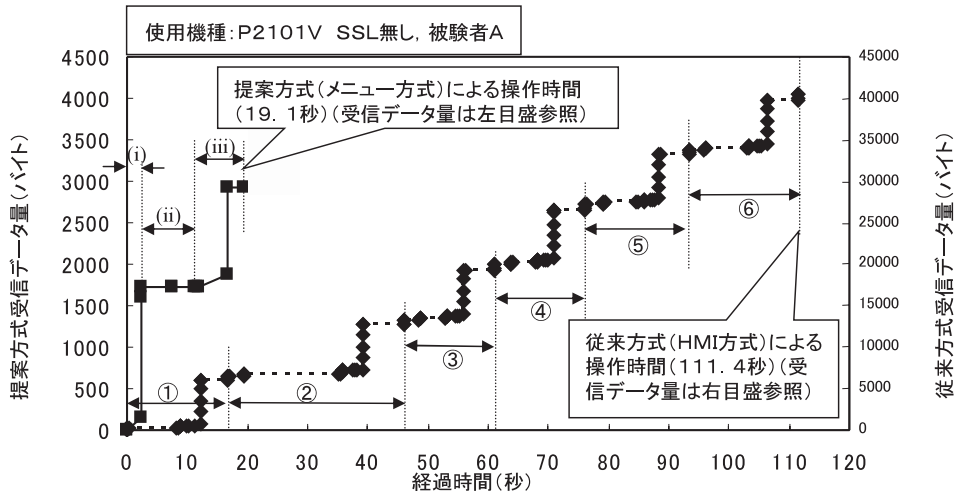
画面表示高速化手法に関する評価は、サーバ上で図 12 に示す時計アニメーションを表示し、画面モードで 1 画面更新に要する時間を測定し、これを比較した (1 分間測定し、平均した)。

表 4 の評価項目 4 より、提案手法による情報量削減により、1 画面の更新時間を約 63% 削減できていることが分かる。

##### (3) 画面方式における入力方式に関する評価

本評価では、矩形のオブジェクトを選択し、その後擬似ポインタを画面右上に移動し、矩形のオブジェクトをドロップする操作 (図 13) を完了するのに要する時間を測定した。表 4 の評価項目 5 にその結果を示す。被験者 A は約 17 秒、被験者 B は約 22 秒で上記操作を完了している。グラフィック操作に関しても、





操作内容: 漢字2文字削除後2文字挿入(近年→最近)

・提案方式における手順

(i) 文章データ1段落分サーバから移動機に転送

(ii) 移動機テキストエディタ上での文章編集“近年”→“最近”

(iii) 編集後文章データをサーバ上に書き戻し(同時に次段落の文章データをダウンロードしている)

・従来方式における手順

① BackSpaceキー2回入力“/b/b”を要求し、サーバ画面を確認→“近年”が正常に削除されていることを確認

② キー入力“saikinn”を要求し、サーバ画面を確認→“さいきん”と表示されていることを確認

③ Spaceキー入力“/”を要求し、サーバ画面を確認→“細菌”と誤変換されていることを確認

④ Spaceキー入力“/”を要求し、サーバ画面を確認→変換候補が表示されていることを確認

⑤ キー入力“2”を要求し、サーバ画面を確認→“最近”と正常変換されたことを確認

⑥ Enterキー入力“/r”を要求し、サーバ画面を確認→変換が確定したことを確認

ただし、“/b”はBackSpace、“/”はSpace、“/r”はEnterキー入力を要求するキーボードエミュレーション記号

図 11 日本語文章編集時間に関する提案方式と従来方式(アプリケーション実行方式)の比較

Fig. 11 Comparison of proposed method with existing method (application execution method) in the case of Japanese text editing time.

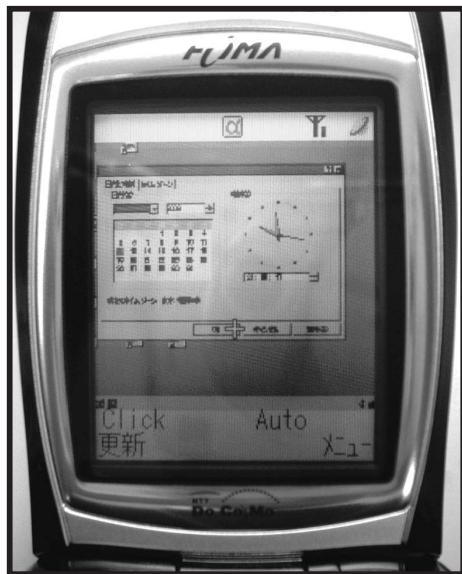
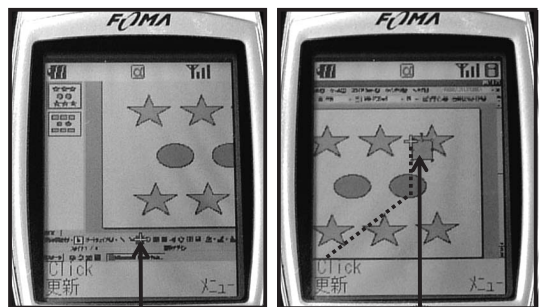


図 12 評価対象アニメーション  
Fig. 12 Animation for evaluation.



(a) 矩形オブジェクトを選択

(b) 擬似ポインタを移動し、矩形オブジェクトをドロップ(矩形オブジェクトのサイズはデフォルト)

グラフィックデータの詳細  
 ・スライド数: 2  
 ・図形オブジェクト数: 8/スライド

図 13 評価対象グラフィックデータ  
Fig. 13 Graphic data for evaluation.

複雑な操作でなければ、最低限の実用性を有している  
と考える。

## 6. おわりに

本論文では、移動機を端末としたシンクライアント  
システムの実現によって、移動機の携帯性とサーバの  
利便性を両立した、理想的なモバイルコンピューティ  
ング環境が提供できるとの考察に基づき、その構成法  
に関して論じた。

その結果、GUI仕様変換ゲートウェイを導入し、移  
動機上で擬似 GUI を提供することを特徴とする、モ  
バイル向けシンクライアントシステムを提案した。

さらに、提案方式のプロトタイプを作成し、その評  
価結果より、提案方式が一定の実用性を有すること  
を示した。

今後は、メニュー方式においてサポートする操作の  
拡充と、画面方式におけるいっそうの操作時間短縮方  
式の検討を行う予定である。

## 参 考 文 献

- 1) Sinclair, J. and Merkow, M.: *Thin Clients Clearly Explained*, Morgan Kaufmann, San Francisco (2000).
- 2) Kanter, J.: *Thin Clients/Server Computing*, Microsoft Press, Washington (1998).
- 3) Inamura, H., et al.: TCP over Second (2.5 G) and Third (3 G) Generation Wireless Networks, IETF, RFC3481 (Feb. 2003).
- 4) 石川太朗, 稲村 浩, 高橋 修: W-CDMA 向け TCP プロファイル, 情報処理学会研究会報告, MBL15-3 (Nov. 2000).
- 5) 関口克己, 鶴巻宏治, 北口雅哉, 高橋 修: モバイルインターネットアクセスにおけるリクエストパイプラインの有効性, 情報処理学会研究会報告, MBL15-1 (Nov. 2000).
- 6) 横田和久, 稲村 浩, 高橋 修: ワイヤレスネットワーク環境におけるメール受信の効率改善, DICOMO 2002 シンポジウム論文集, pp.21-24 (Jul. 2002).
- 7) Open Group: Network Computing — A New Business Paradigm, Open Group White Paper W801 (Apr. 1998).
- 8) Craft, M., et al.: Configuring Citrix Metaframe for Windows 2000 Terminal Services, *SYNGRESS*, Rockland (2000).
- 9) Mathers, T.W.: *Windows NT/2000 Thin Client Solutions*, MTP, Indianapolis (2000).
- 10) Richardson, T. and Wood, K.: The RFB Protocol, ORL, Cambridge (1998).
- 11) 中村文隆: VNC 詳細解説, CQ 出版社, 東京 (2000).
- 12) Su, N., Tsukamoto, M. and Nishio, S.: Rajicon: A system for Remote PC Access through a Cellular Phone, DICOMO 2001 シンポジウム論文集, pp.349-354 (Jun. 2001).
- 13) Frier, A., Karlton, P. and Kocher, P.: The SSL 3.0 Protocol, Netscape Communications Corp. (Nov. 1996).
- 14) Kent, S. and Atkinson, R.: IP Encapsulating Security Payload (ESP), IETF, RFC2406 (Nov. 1998).
- 15) Simpson, W.: PPP Challenge Handshake Authentication Protocol (CHAP), IETF, RFC 1994 (Aug. 1996).
- 16) Haller, N., et al.: A One-Time Password System, IETF, RFC2289 (Feb. 1998).
- 17) RSA ACE/Server and RSA SecureID Deployment Guide, RSA Security, Bedford (2001).
- 18) Housley, R., et al.: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, IETF, RFC3280 (Apr. 2002).
- 19) The Java APIs and Architectures for XML: A Technical White Paper, Sun Microsystems, Inc. (Jun. 2002).

(平成 15 年 8 月 11 日受付)

(平成 16 年 3 月 5 日採録)

## 推 薦 文

本研究は、モバイルコンピューティング環境において携帯端末から PC 上の APL を制御するためのモバイルシンクライアントを提案している。実機上に実装し評価を行った点、機能性と汎用性の両者を兼ね備えた方式であることから高く評価できる。しかしながら、評価が少なく、ターゲットアプリケーションが文書編集のみとなっている点から、その点をコメントした。( MBL 研究会運営委員 井手口 哲夫 )





高橋 竜男(正会員)

1964年生。1989年3月東海大学大学院工学研究科修士課程修了,同年4月NTT入社。交換システム研究所にてノードシステム用制御系アーキテクチャ,分散プラットフォームの研究開発に従事。1998年NTTドコモに異動。マルチメディア研究所にてモバイルインターネットの研究開発に従事。現在,同社ネットワーク研究所。電子情報通信学会会員。



高橋 修(正会員)

1951年生。1975年3月北海道大学大学院工学研究科修士課程修了,同年4月電電公社(現NTT)入社。情報通信研究所でコンピュータネットワークアーキテクチャの研究開発,およびOSIの標準化に従事。1999年NTTドコモに異動。マルチメディア研究所にてモバイルインターネットの研究開発に従事。2004年4月より公立はこだて未来大学システム情報科学部情報アーキテクチャ学科教授。工学博士。電子情報通信学会会員。



水野 忠則(フェロー)

1945年生。1968年名古屋工業大学経営工学科卒業。同年三菱電機(株)入社。1993年静岡大学工学部情報知識工学科教授,現在,同大学情報学部情報科学科教授。工学博士。情報ネットワーク,モバイルコンピューティング,放送コンピューティングに関する研究に従事。著書としては『プロトコル言語』(カットシステム),『コンピュータネットワーク概論』(ピアソン・エデュケーション)等がある。電子情報通信学会,IEEE,ACM各会員。