

Keio Media Space Board と Keio Media Space Navigator 上のプロトタイプサーバの実装と評価

望月 祐洋¹ 富田 修平² 川又 浩一² 石川 直太² 徳田 英幸^{1,2}

¹慶應義塾大学大学院 政策・メディア研究科

²慶應義塾大学環境情報学部

あらまし

現在、我々は人間の情報行動に基づく協調活動支援基盤である Keio Media Space Family(以下 KMSF)を開発中である。KMSF は、Keio Media Space Board(以下 KMSB) および Keio Media Space Navigator(以下 KMSN)を中心として、これらと協調動作するハードウェア、ソフトウェアなどのさまざまな構成要素の統合環境である。

本稿では、KMSB 上で動作するサーバプロトタイプであるボードサーバおよび KMSN 上で動作するクライアントプロトタイプであるナビゲートクライアントで構成されるボードシステムの実装と評価について述べる。

Keio Media Space Board and Keio Media Space Navigator: Prototype Server Implementation and Evaluation

Masahiro Mochizuki¹ Shuhei Tomita² Kouichi Kawamata³ Naota Ishikawa⁴ Hideyuki Tokuda⁵

¹moma@mag.keio.ac.jp, Keio University, 5322, Endo, Fujisawa-shi, Kanagawa, 252 Japan,

²shuhei@sfc.wide.ad.jp, Keio University, 5322, Endo, Fujisawa-shi, Kanagawa, 252 Japan,

³kawauso@sfc.wide.ad.jp, Keio University, 5322, Endo, Fujisawa-shi, Kanagawa, 252 Japan,

⁴naota@slab.sfc.keio.ac.jp, Keio University, 5322, Endo, Fujisawa-shi, Kanagawa, 252 Japan,

⁵hxt@sfc.keio.ac.jp, Keio University, 5322, Endo, Fujisawa-shi, Kanagawa, 252 Japan

Abstract.

At present, we have been developing an collaborative system architecture named Keio Media Space Family(KMSF) which supports various types of human collaborations. KMSF is an integrated environment which consists of Keio Media Space Board(KMSB), Keio Media Space Navigator(KMSN) and other cooperative hardwares and softwares.

In this paper, we describe the implementation and evaluation of a prototype system named "Board System" which consists of several servers and clients running on the KMSB and KMSN.

1 はじめに

計算機の小型化, 高性能化, 低価格化にともないその普及が進み, 人間のさまざまな活動の場での利用機会が増加している。

従来, 計算機は主として個人用途に利用されてきたが, 通信機器の発達, LAN 製品の普及, 公衆無線網の整備, インターネットの一般への浸透などにより, 場所や時間を問わないネットワークの利用が可能となった。広域ネットワークが身近になることでさまざまな情報やサービスへのアクセスが容易になり, 人間同士のコミュニケーションの道具としても活用されている [1, 2]。

計算機ネットワークの利点の一つは世界規模の情報共有が可能となる点にあるが, 既存のファイルシステムレベルの情報共有から, 他者とのインタラクティブな情報のやり取りの中で, より動的かつ柔軟な情報の創造, 共有が実現可能な環境が重要となる。

KMSF は, このような要求に応えることを目的とした環境の構築を目指している。

他者との情報の共有過程の円滑化および情報の作成, 編集, 交換といった情報行動の連続性を向上することで, 不要なオーバヘッドを軽減し, 人間の思考や行動に暗黙裡に課される制約を緩和する。

本稿では, このような環境の実現のため KMSB, KMSN のプロトタイプ実装であるボードサーバ, ナビゲートクライアントで構成されるボードシステムの実装と評価について述べる。

2 KMSF での知的協調作業

KMSF は, 複数人で共有可能な可搬型表示装置である KMSB および個人が所有する PDA である KMSN を中心に構成される。

KMSF 環境では, 個人が所有する KMSN を KMSB と連携させつつ日常生活のさまざまな場面での積極利用を目指す。

KMSB をグループや研究室でのホワイトボードの代替品として利用する。あるいはキャンパス掲示板を電子化したものとして位置づけ, 学生が KMSN 経由で必要な情報の取得, 発信を行う。また WS, Macintosh, PC の画面を大型プロジェクタに投影し, 各種プレゼンテーションツールを利用して講義を行っている状況を想定し, 学生の所有する KMSN とプロ

ジェクタ画面となる KMSB との間で動的な情報交換を可能にする。

これらさまざまな場面で生じる複数人での遠隔会議, 協働編集, 協働描画, スケジュール調整などの要求を, 複数メディア間の情報共有, 情報の互換性, 情報行動の連続性などの保証を通じて実現する。

KMSF のプロトタイプ実装であるボードシステムでは, KMSF 環境で実行されるさまざまなコミュニケーションの中でも KMSB-KMSN 間でのコミュニケーションに着目し, これを実現する機能を提供する。

3 KMSB-KMSN の通信機構

KMSN 上では, 個人が目的に応じてオペレーティングシステムを使い分ける可能性がある。

複数のオペレーティングシステムプラットフォームにおける任意の KMSB-KMSN 間通信の互換性を保証するために, クロスプラットフォーム通信ライブラリを実装している [3]。人間の知的協調作業実現の目的から要求される機能を定義した高レベルな通信機能の実現部が HXLib であり, 機種依存のない互換性重視の低レベル通信機能の実現部が CXLib である (図 1)。

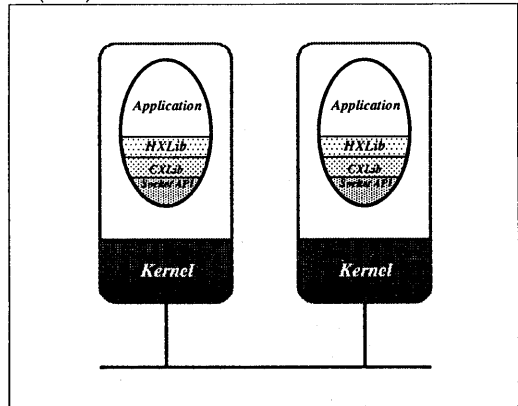


図 1: KMSB-KMSN 間の通信モデル

以下で HXLib, CXLib それぞれの実装について述べる。

3.1 HXLib の実装について

HXLib は, KMSF アプリケーション層からの要求と低レベルのクロスプラットフォーム通信ライブラ

りである CXLlib との接点となる高レベルのライブラリである。

KMSB-KMSN 間の情報交換には情報の post-it, fetch-it というメタファを用いており、これらのオペレーションのサポートが HXLlib の機能の中心である。

post-it, fetch-it のメタファは、文書データの交換に限らず静止画像やリアルタイム動画像などの交換にまで適用される。

現在の実装では、ボードサーバナビゲートクライアント間通信における文書データ交換に特化した機能に限定されている。

文書データのさまざまな加工処理を想定し、SGML(Standard Generalized Markup Language)によって文書の構造化を行っている [4]。

HXLlib を利用して構造化文書データを交換することで、すべてのデータではなくユーザが必要な部分データのための指定に基づく情報取得が可能である。

以下に、ボードサーバから特定データを fetch する際に必要となる情報の取得用の関数を示す。

- (1) 過去のデータが保存されているディレクトリ数
HXServerCmdGetNumofDirectories()
- (2) データが保存されているディレクトリ名
HXServerCmdGetNameofDirectories()
- (3) 指定ディレクトリに存在するファイル数
HXServerCmdGetNumofFiles()
- (4) 指定ディレクトリに存在するファイル名
HXServerCmdGetNameofFiles()
- (5) 指定ディレクトリ内の指定ファイルの内容
HXServerCmdGetFiles()

SGML 文書からタグ別情報を fetch する関数は以下の通りである。

```
HXGetTagContent(HXBoardName* board,
                char* dirname, char* filename,
                char* recvbuf, int recvlen);
```

ディレクトリ名/ファイル名/タグ名で指定される内容を返す。

```
HXGetDocumentType(HXBoardName* board,
                  char* dirname, char* filename,
                  char* recvbuf, int recvlen);
```

ディレクトリ名/ファイル名で指定されたファイルの文書型を調べて返す。

3.2 CXLlib の実装について

CXLlib は、クロスプラットフォーム (Macintosh, Windows, UNIX) 開発を目的とする低レベルの異機種間通信ライブラリである。通信プロトコルとして TCP/IP を採用している (現在のバージョンでは Macintosh と BSD UNIX をサポート)。表 1 が開発

表 1: 開発環境

使用ハードウェア	Macintosh Quadra 840AV Macintosh Duo280c
使用システム	Macintosh OS 7.1.1+MacTCP
コンパイラ	THINK C 6.0(Symantec) CodeWarrior CW4(metrowerks)

環境である。

CXLlib を利用することで、クライアントプログラム開発時のオーバヘッド軽減やソースコードのシンブル化を目指す。表 2 で CXLlib の基本関数を示す。

表 2: CXLlib 基本関数 (抜粋)

CXOpen(server, port)	サーバへの接続を試みる
CXRead(buffer, length)	サーバからデータを読む
CXWrite(buffer, length)	サーバへデータを書く
CXClose(void)	サーバとの接続を閉じる

基本関数の他に、クロスプラットフォーム用に CXLlib の初期化を行う関数として CXInit() 関数を用意している。この関数で、漢字コードや改行コードなど機種依存の情報を指定する。

4 ボードサーバの実装

KMSB 上で動作するサーバプロトタイプであるボードサーバについて述べる。

現在のバージョンでは、定型文書を中心とする構造化データの取り扱いが主であり、ボードサーバはナビゲートクライアントを通じて送信される post-it 要求に応じてデータを格納し、ユーザからの fetch-it 要求に応じてデータを返すというのが基本動作である。

4.1 ボードサーバの機能

ボードサーバの機能は以下の 3 つに分類できる。

- (1) 入出力管理
- (2) 表示管理 (座標, 表示, 消滅)
- (3) データ管理 (時間, 分類, 保存, 検索)

現在のバージョンの実装では①受信, 整理, 保存部, ②表示部の 2 機能に分割して実装を行っている (図 2)。

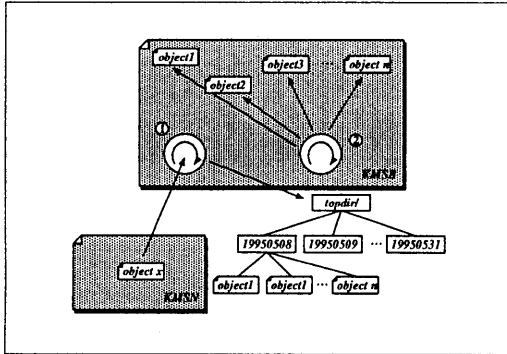


図 2: ボードサーバの構成

受信, 整理, 保存部は受信したデータの SGML タグ情報に基づき, 情報が発信された日付別にディレクトリを作成し, 到着順に番号付けを行い情報を整理すると同時に, 情報内容別にシンボリックリンクを張る. 表示部は受信したデータをディスプレイ上に表示する.

現在 BSD/OS2.0, SunOS4.1.4 上の X11R6 の環境で Tk ツールキットを用いて実装を行っている.

4.2 SGML による文書構造化

KMSF では, 種々の知的情報処理の実現を目的としてあらかじめデータの構造化を行う. 任意の KMSF-KMSN 間でやり取りされるデータの形式はさまざまだが, データを構造化することでデータ検索やフィルタリング処理の効率化を図る.

KMSF のプロトタイプであるボードシステムでは, テキスト処理に重点を置き SGML による文書構造化を実現している. 文書構造化に際し, ボードを介したコミュニケーションで送受する情報の種類をいくつか類型化し, 文書型を定義している.

表 1~3 は, 現在定義済みの文書型一覧である.

表 1: 簡単なコメントやメモ用	
FREEMEMO 形式	
<Poster>	post-it した人の名前
<Date>	post-it した日時
<Content>	内容
MEMO 形式	
<Subject>	主題
<Poster>	post-it した人の名前
<Date>	post-it した日時
<Importance>	重要度 0~255 までの整数
<Content>	内容
EVENTGLOBAL 形式	
<Subject>	主題
<Poster>	post-it した人の名前
<Date>	post-it した日時
<Place>	イベント開催場所
<Eventtime>	イベント開催日時
<Content>	内容

表 2: 掲示板用	
EVENTLECTURE 形式	
<Subject>	科目名
<Poster>	post-it した人の名前
<Date>	post-it した日時
<Teacher>	担当教員
<Eventtime>	時間
<Place>	教室
<Content>	内容
MEMOTO 形式	
<Subject>	主題
<Poster>	post-it した人
<Date>	post-it した場所
<To>	宛名
<Content>	内容

表 3: 研究室やプロジェクトなど小グループ活動用

表 3: 研究室やプロジェクトなど小グループ活動用	
FILERAW 形式	
<Subject>	主題
<Poster>	post-it した人の名前
<Date>	post-it した日時
<Keyword>	キーワード
<Filename>	ファイル名
<Content>	内容
FILELATEX 形式	
<Subject>	主題
<Poster>	post-it した人の名前
<Date>	post-it した日時
<Keyword>	キーワード
<Filename>	ファイル名
<Content>	内容

基本的には WWW における html のように利用者自身が記述するのではなく, クライアント側で記述の煩わしさを隠蔽する機構の提供を図る.

必要な文書型は利用者からの要求に依存する. 文書型をあらかじめ厳密に定義すべきか, 利用者の必要に応じて柔軟に定義可能な機構として提供すべきかなどを検討中である.

5 ナビゲートクライアントの実装

KMSN 上ではさまざまな知的エージェントが協調しつつユーザを支援するが, ここでは現在個別開発中のナビゲートクライアント上の諸機構を中心にプロトタイプ実装について述べる.

5.1 ナビゲートクライアント (Mac 版)

Macintosh を利用してボードからの SGML のタグ付け文書の fetch, 作成した文書の自動タグ付, ボードへの paste までをサポートする. 表示情報のレイアウト管理も行うため, Macintosh をボード用途に使用することも可能である (図 3).

5.2 情報スケーリング機構

KMSF では, 複合メディア空間から取得する情報を直接操作するために, ディスプレイという限られたスペースに多数の情報を同時に表示する必要がある (ディスプレイが大型になっても, 解像度と表示可

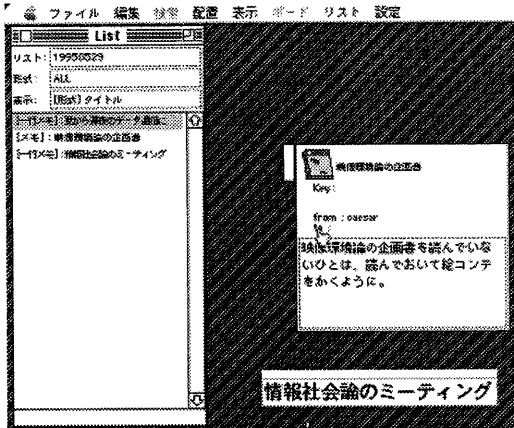


図 3: Macintosh 用ナビゲートクライアント

能な情報量には限界がある). この要求を実現する手法の一つが、情報スケーリング機構である [5].

情報スケーリングとは、入手した情報量と現在表示可能な情報量を比較しつつ、個人の興味に応じて情報表示の粒度を適宜変更する技術である。

プロファイルに記述した情報の重要度を利用してスコアを計算し、これに基づき情報表示の粒度を調節する。情報フィルタリングにおける単語抽出技術を利用し、文章中から単語だけを抜き出し表示することで限定範囲内に情報を納める。

文章は画像と異なり、過度に縮小すると判読不能になり情報価値が著しく低下する。文章中の単語のみを可読サイズに保ち、それ以外の部分をさらに縮小することで情報価値の維持しつつ全体を縮小する。

この縮小の段階をテキストレベルと呼ぶが、情報スケーリングではテキストレベルの動的変更に応じた情報の表示を行う (図 4)。

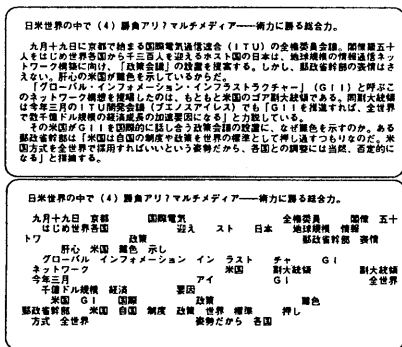


図 4: テキストレベル高低の変更例

5.3 移動オブジェクト制御機構

KMSF のプロトタイプであるボードシステムにおける情報の post-it と fetch-it の実体は SGML でタグ付けされたファイルのやり取りだったが、現在自律オブジェクトを情報交換/編集の単位とするシステムを実装中である (図 5)。コントローラ (あるいは種々の入力デバイス) を通じて自律オブジェクトにメッセージを送信することで移動制御を行い、post-it, fetch-it などのオペレーションを実現する。

例えば、特定のスクリーン上に新たにオブジェクトを post-it する場合、{screen, object_id, owner} の組で post-it 先スクリーンのオブジェクトマネージャに kms_obj_create(screen, object_id, owner, ...); メッセージを送信する。オブジェクトマネージャはオブジェクトを生成し、管理リストにオブジェクト ID を追加する。

複数のスクリーン上に同一オブジェクトを同時に post-it する場合、複数のオブジェクトマネージャ宛にメッセージをマルチキャストする。

ローカルなオブジェクトをリモートスクリーンに移動する場合、オブジェクトマネージャに対して

```
kms_obj_move(from, to, ...);
```

メッセージを発行する。この結果ローカルオブジェクトマネージャはオブジェクトに消去メッセージを送信し、移動先オブジェクトマネージャに管理情報を移転した後にローカルの管理情報を消去する。



図 5: オブジェクト制御機構

6 評価

ボードシステムを定性的な評価結果を、表 3 にまとめる。評価対象としてホワイトボード、電子掲示板、WWW KMSF を選択した。

表 3: 各メディアの機能別比較

評価項目	評価対象	ホワイトボード	電子掲示板	WWW	ボードシステム	KMSF
取り扱い情報形式		文字, 静止画	文字	文字, 音声, 静止画, 動画	文字, 静止画	文字, 音声, 静止画, 動画
インタラクティブティ		○	×	×	△	○
情報共有方式		△	○	△	○	○
情報保存方式		△	○	○	○	○
直接操作感		○	×	×	△	○
情報行動連続性		×	×	△	△	○
入力方式		ペン	キーボード	マウス, キーボード	マウス, キーボード	音声, ペン, 指, キーボード

扱える情報形式の多様さはどうなのか。複数人での同期/非同期のコミュニケーションに適するか否か、またその際情報をどのように共有、保存するのか。情報処理作業を進める過程で、性質の異なる作業間を行き来する場合でも継続性を保てるのか。あるいは情報やメディアにどのようにはたらきかけるのかといった点を考慮し、評価項目として7つを挙げた。

7 考察

情報行動連続性を保証するために情報保存方式、直接操作感は不可欠な要素である。また直接操作感を与えるためには、入力方式が重要である。複数人の協調作業を円滑に進めるためには、インタラクティブティや情報共有方式が十分考慮される必要がある。

このように見ると、個別機能の実現よりもむしろ全体の調和の取れた機能透過性が要求されていることが分かる。メディアの利用を通じて思考や行動に加えられる制約は、訓練や慣れによって徐々に緩められる。訓練による克服を目指すか、制約そのものの解消に努めるか結論を出すためには情報行動の過程をつぶさに観察する必要があるだろう。

8 まとめ

我々は、人間同士の知的協調作業支援基盤であるKMSFのプロトタイプの一部としてボードシステムの実装を行い、ボードシステムの構成要素であるボードサーバとナビゲーションクライアントの通信機構や実装済みの諸機能について述べた。

今後の課題として、自律オブジェクトを情報交換、編集などの人間の情報行動で扱う基本単位として位置づけシステム全体の再設計と実装を行う必要がある。

新たなシステム設計に基づき映像・音声情報の取り扱いも考慮し、post-it, fetch-it というメタファを用いた場合連続メディアオブジェクトの記録、再生、

編集などの処理と自律オブジェクトの振舞いにどのように整合性を持たせることが可能か検討する。

また、情報スケーリングなどの知的処理を人間の意識しない部分で行わせることで人間にかかる負担を軽減させる一方、自律オブジェクト単位の情報交換の結果生じるシステムやネットワーク資源へのコストを可能な限り抑えるよう定量的評価に基づくシステム最適化が大きな課題として残されている。

9 謝辞

本プロジェクトを遂行するにあたり、協力していただいた慶應義塾大学環境情報学部 徳田・村井研究室の皆様、また慶應義塾大学大学院 政策・メディア研究科「オープンメディアと高速ネットワーク」プロジェクトのサブプロジェクト「Ubiquitous Operating System(UOS)」プロジェクトの皆様へ感謝致します。

参考文献

- [1] 望月, 峯尾, 村井, 徳田, “汎用機器の統合制御による日常研究活動支援システムの構築”, 情処研報, Vol. 94, No. 12, 94-DPS-63, 94-GW-5, pp. 17-24(1994).
- [2] 望月 祐洋, “次世代計算機環境におけるコミュニケーション・ツール体系に関する一考察”, 徳田・村井研究会卒業論文, 1993 年度.
- [3] 富田 修平, “知的協調作業を支援するアプリケーション開発環境の実装”, 徳田・村井研究会卒業論文, 1994 年度.
- [4] C.F.Goldfarb, Editor. *The Standard Generalized Markup Language(ISO 8879)*. International Organization for Standardization, Geneva, 1986.
- [5] 川又 浩一, “情報洪水の緩和のためのインフォメーションスケーリングの実現”, 徳田・村井研究会卒業論文, 1994 年度.