

# Keio Media Space Family プロジェクトにおける システムアーキテクチャ†

徳田 英幸<sup>1,2</sup> 石川 直太<sup>1</sup> 望月 祐洋<sup>2</sup> 富田 修平<sup>1</sup> 川又 浩一<sup>1</sup>

<sup>1</sup>慶應義塾大学環境情報学部 <sup>2</sup>慶應義塾大学大学院 政策・メディア研究科

あらまし KMSF(Keio Media Space Family)プロジェクトでは、現在開発されている分散実時間マイクロカーネル、実時間ネットワークプロトコル、統合メディアサーバ、連続メディアベースなどの新しい技術を利用して、1対1の協調作業だけでなく、複数のグループ間での知的協調作業をサポートするKMSF環境を構築している。本稿では、KMSF環境のシステムアーキテクチャ、およびKeio Media Space BoardとMedia Space Navigatorの概要を述べる。

キーワード メディアスペースボード、メディアスペースナビゲータ、OSアーキテクチャ、マイクロカーネル、連続メディア

## System Architecture for Keio Media Space Family Project

Hideyuki Tokuda, Naota Ishikawa, Masahiro Mochizuki,  
Shuhei Tomita, Kouichi Kawamata

{hxt, naota, moma}@sfc.keio.ac.jp,  
{shuhei, kawauso}@sfc.wide.ad.jp,  
Faculty of Environmental Information, Keio University,  
5322, Endoh, Fujisawa-shi, Kanagawa, 252 Japan,

**Abstract** We have been developing distributed real-time microkernel, real-time network protocols, media servers, and continuous media base systems. Based on these technology, we are developing a new media space environment for collaborative groups in the KMSF (Keio Media Space Family) project. In this paper, we describe the system architecture of the KMSF environment and the overview of Keio Media Space Board and Keio Media Space Navigator.

**Keyword** Media Space Board, Media Space Navigator, Micro-kernel, Continuous Media

---

†この研究の一部は、KEIO-IBM パートナシッププログラムのもとに行なわれました。

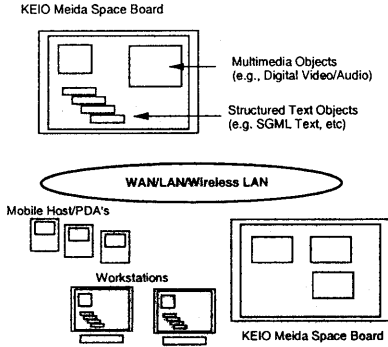


図 1: Keio Media Space Board and its Navigator

## 1 はじめに

現在、慶應義塾大学で開発している分散実時間カーネル、実時間ネットワークプロトコル、メディアサーバ、連続メディアベースなどの新しい技術を利用して、1対1の協調作業だけでなく、複数のグループ間での知的協調作業をサポートする統合環境 [1, 6, 7] として Keio Media Space Family (KMSF) を構築している。

KMSF は、Keio Media Space Board (KMSB) と Keio Media Space Navigator (KMSN) を中心にこれらと協調動作するハードウェア、ソフトウェアから構成される統合環境である。Keio Media Space Board とその Navigator の概念図を図 1 に示す。

Keio Media Space Board (KMSB) は、ネットワークに接続された装置で、入出力機構を持った可搬型の表示装置である。従来の大型ディスプレイ装置と異なり、キーボード、ペンなどで直接データを書き込むことが可能である。例えば、従来から使われているオフィスでのホワイトボードとしてや、大学などで利用されているキャンパス掲示板や、ビデオホールなどで利用されているビデオウォール [1] としても利用可能である。現在は、XEROX 社の Live Board [12] や UNIX ワークステーションなどを利用して開発・実験を始めている。

Keio Media Space Navigator (KMSN) は、一般に使用されている PDA (Personal Digital Assistant) や携帯型のノートブックコンピュータなどのハードウェアと様々な KMSF 用のソフトウェアを総称したものである。ユーザが、メディアスペースを自由に動き回るための知的ヘルパの役目をもち、Mosaic や Netscape などで利用できるナビゲーション機能や様々なエージェントプログラム [5] とともに、ペン入力やスピーチ入力などの機能も利用できることを目標としている。KMSN のハードウェアプラット

ホームとしては、現在 IBM 社の TP230Cs, TP750C などの PC 互換機や Apple 社の Powerbook Duo などのノートブックコンピュータを利用して開発を行なっている。ソフトウェアプラットホームとしては、我々が開発している Real-Time Mach3.0 マイクロカーネル [9] や UNIX, MacOS, Windows 3.1 といったクロスプラットホーム環境での利用をめざしている。また、応用ソフトウェアとしては、PIM (Personal Information Management) に関するソフトウェアやそれらと関係するエージェントプログラムや大学などでのキャンパスライフ支援プログラムとしての Academic Information Management などが開発されている。

本稿では、このような KMSF 環境におけるハードウェアおよびソフトウェアに関するシステムアーキテクチャについて概説し、Keio Media Space Board と Keio Media Space Navigator の概要を述べる。

## 2 KMSF システムアーキテクチャ

KMSF 環境では、人間が知的活動を行なっている従来からの物理的な共同作業空間と、KMSB や KMSN などを通じてのネットワーク上で広がる仮想的な共同作業空間をシームレスに統合した環境の実現を目標としている。従来のワークステーションや PDA を通じての作業環境では、サイバースペースでの共同作業は可能であるが、実際の物理空間を共有しながら共同作業をすることが難しい。共有作業スペースなども一つのウィンドウのサイズなどに制約されてしまい、実時間でインタラクティブな協調作業にはあまり適していない。

KMSF 環境は、いろいろなアプリケーションに活用できるが、基本的には、図 1 に示されているように、ネットワーク上ですべての機器は接続されていることを前提としている。ユーザは、ボード上に表示されている情報を端末機器なしに物理的に読み取ることも可能であるし、自分が利用している携帯端末からネットワークを通じて必要な情報を掲示したり、取り出したり、編集することも可能である。

KMSF では、このようにシステム内に情報 (オブジェクト) を掲示するアクションを *post-it* と呼び、取り出すアクションを *fetch-it* と呼ぶ。KMSF 内にポストされたオブジェクトは、共有オブジェクト空間内に生成/登録され、ACL (Access Control List) によってアクセス権が制御される。また、ユーザが *fetch-it* のアクションをとる場合、単なる読み出し操作だけでなく、検索のための知的フィルタリング操作を行なうことができる。フィルタリングをした後に取り出されたオブジェクトは、KMSN 内のオペレーティングシステムが提供しているプライベート

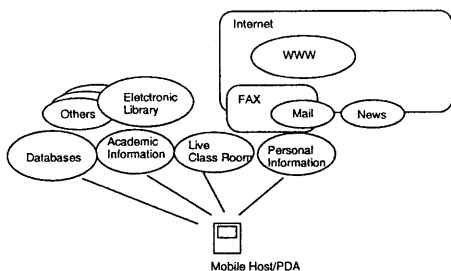


図 2: A Mobile Host with KEIO Media Space

トなオブジェクト空間に移動され、管理される。

KMSFで取り扱っているオブジェクトとしては、テキストのように構造化されたオブジェクトと、デジタルビデオやオーディオといった連続メディアのオブジェクトがある。

連続メディアのオブジェクトに対しては、*continuous-post-it* や *continuous-fetch-it* といったオペレーションを行なうことができる。*continuous-post-it* オペレーションでは、例えば、自分のモバイルホストから取り込んだ動画のある QOS(Quality of Service) レベル [10] でボード上に連続的にポストすることを実現する。連続メディアオブジェクトの QOS レベルは、時間的解像度 (temporal resolution) と空間的解像度 (spatial resolution) などによって指定する。例えば、時間的解像度は、ビデオデータでの毎秒のフレーム数 (fps: frame per second) であったり、オーディオデータでのサンプリングレートで指定する。また、空間的解像度は、ビデオデータでのフレームサイズや 1 画素あたりの色数を指定する。

ビデオウォールを介しての物理空間でのリアルタイム通信に KMSB を使うような場合は、QOS レベルで指定した時間的解像度とともに空間的解像度が十分高くなければならない。一方、モバイルホストからボード上の連続メディアオブジェクトをフェッチする場合など、ポストされた QOS レベルと全く同一のレベルである必要はなく、通信容量の小さいネットワークからでも連続フェッチできるように動的に QOS レベルを調節する必要がある。

KMSF 環境は、大学や企業などの単一組織だけでなく、複数の組織がネットワークで接続された複数組織に広がったメディアスペースを想定しているため、ユーザの認証なども AFS のような形態を必要とする。

この KMSF 環境を大学のような高等教育機関などで利用したとすると、ユーザがアクセスできる情報としては、図 2 に示されるようにさまざまメディアスペース上に広がっている。

### 3 KMSB

Keio Media Space Board は、物理的な共同作業空間を提供できるよう物理的にも大きなボードとなることが可能であり、また、インタラクティブで仮想的な共同作業空間も提供できるよう高速なネットワークと接続されている。

### 4 KMSB のハードウェア特性

KMSB を実現するためのハードウェアとしては、以下の特性を持っていることを目標としている。

- 高い可搬性。たたんで持ち運び可能。
- 最小 17 インチから 100 インチぐらいまでの可変サイズ。
- モジュラ構造で、複数枚を上下左右に接続可能。
- 強靭性。人がボード上に乗っても使用可能。
- 入力方式として、感圧式センサ、光学式センサ、マーカー (ペン)、レーザーペン、スピーチ入力などが可能。
- ディスプレイ部の入出力方式として、ATM ネットワークに直結し、ATM-cell レベルでの入出力も可能。

現時点では、残念ながらこのようなボードハードウェアとして市販されているものが存在しない。しかし、我々は、UNIX ワークステーションと大型ディスプレイの組合せや、ペン入力をもったデバイスとして XEROX 社の Live Board をボードのプロトタイプシステムの開発実験のために使用している。ただし、Live Board システムは、200kg 近くあり、ボードとしての可搬性に劣っているだけでなく、ボードの上で行なうゲームなどの実験には全く利用することができない。

#### 4.1 ボードの利用形態

KMSB は、いろいろな場面での応用が考えられているが、ここでは、代表的なアプリケーションのいくつかを紹介する。

グループ間でのテレビ会議システム: 大型ディスプレイとワークステーションの代わりにボードを各部屋に置き利用する。

ビデオウォール: 語学教室などの部屋の 3 つ以上の壁面を KMSB で覆い、リアルタイムに外国の実際のある場所の映像/音声などを連続ポストし、仮想的な学習環境を創り出す。

ゲームボード: カードゲームやボードゲームを電子化し、

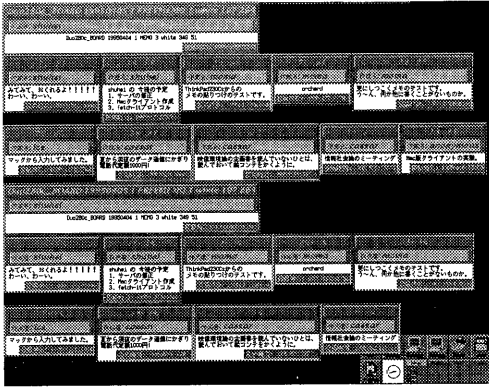


図 3: A Screen Dump of the KMSB Prototype

KMSB 上にカードなどを表示し、タッチ入力機能などによりゲームを実行する。

電子掲示板: いろいろな情報をポストしたり、取り出したりできる物理的にも大きな掲示板。

## 4.2 KMSB プロトタイプ

KMSB プロトタイプでは、想定しているハードウェアが商用では存在しないので、ボードの機能を大型ディスプレイ接続可能なワークステーションを使って実験を行っている。ワークステーション上のボードサーバがオブジェクトの管理をはじめ、通信管理を行なう。

現在のプロトタイプでは、UNIX ワークステーション上の X11R6 ウィンドウシステムを利用して、表示機能の実験、オブジェクトの管理を UNIX ファイルシステム上で行なっている。ボードに post-it されたオブジェクトは、UNIX 上で動いているボードサーバがファイルシステム上に 1 つのファイルとして蓄積し管理している。また、ボードサーバと KMSN プロトタイプの間では、TCP/IP を使ってオブジェクトの交換を実現している。実際には、いろいろなタイプのオブジェクトに対して post-it や fetch-it のオペレーションがサポートされなければならないが、テキスト情報に関しては、SGML[2] のタグをもったデータをファイルとして蓄積し、検索、分類、圧縮などの方法について実験を行なっている。

現在の KMSB プロトタイプのディスプレイ表示方式の 1 つを図 3 に示す。ここでは、いくつかの SGML のタグを持ったテキストオブジェクトが表示されている。

KMSB プロトタイプのソフトウェアアーキテクチャに関しては、以下の図 4 ように表すことができる。

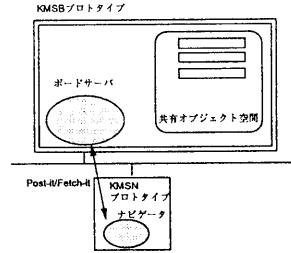


図 4: Software Architecture of the KMSB Prototype

## 5 KMSN

Keio Media Space Navigator は、モバイルホストや PDA などのいろいろなハードウェアプラットフォームと、メディアスペース内をナビゲートするための基盤ソフトウェアと応用ソフトウェアからなる。従来からのウィンドウシステムベースのインタフェースだけでなく、新しい音声入力のインタフェースなどもサポートすることをめざしている。

また、特定の OS に限定された KMSN システムではなく、できる限り多くの異種のハードウェアプラットフォームを支援でき、異種 OS 基盤上で実行できることに加え、次のような特性を持っている。

- マイクロカーネルをはじめ異種 OS に対応したソフトウェアアーキテクチャ
- ユーザレベルプロトコルサーバや他のサーバのサポート
- ワイヤレス LAN のサポート
- 内蔵型ビデオカメラのサポート
- PCMCIA TypeII, III のサポート
- 音声入出力機能のサポート
- カラーディスプレイ

KMSF 環境は、従来からのカラー TFT 液晶ディスプレイを持つノートブックコンピュータやサブノートサイズのシステムが実験用ターゲットシステムとして使われている。上の多くの条件は、かなり満たされているが、内蔵型ビデオカメラやワイヤレス LAN が標準でサポートされているマシンはまだ普及していない状況である。

### 5.1 KMSN プロトタイプ

KMSN プロトタイプは、現在 IBM 社の TP230Cs, TP750C などの PC 互換機や Apple 社の Powerbook Duo

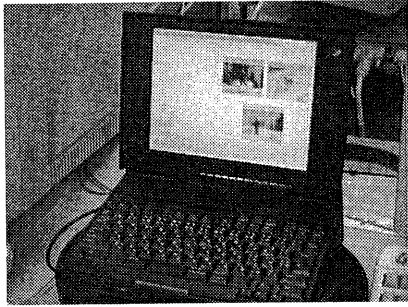


図 5: A Configuration of TP750C

などのノートブックコンピュータを利用して、基盤ソフトウェアとアプリケーションソフトウェアの両面を開発している。

KMSNプロトタイプに使用している IBM TP750C を図 5 に示す。

基盤ソフトウェアとしては、特定のオペレーティングシステムに限定せず、できるだけ個人の目的にあうような様々なソフトウェア基盤環境を整備している。現在までのところ、次のような基盤ソフトウェア環境を整備している。(図 6 参照)

- Real-Time Mach3.0 マイクロカーネル+プロトコルサーバ
- Real-Time Mach3.0 マイクロカーネル+OS サーバ群
- UNIX
- MacOS
- DOS+Windows

Real-Time Mach3.0 マイクロカーネル+プロトコルサーバの環境は、もともとアプリケーション指向に機能を特化した構成であり、ナビゲータ上でのアプリケーションの効率化が期待できる。一方、Real-Time Mach3.0 マイクロカーネル+OS サーバ群の構成は、KMSN のハードウェア資源をもっとも必要とする構成であるが、既存のソフトウェアとの互換性を維持することが可能である。商用の OS である UNIX, MacOS, DOS+Windows の環境では、連続メディアなどの転送をした場合に、システム機能が固定化されているため、QOS の動的変更などアプリケーション指向でいろいろな実験するには、システム変更が難しい問題がある。

KMSN プロトタイプで実験している post-it や fetch-it の操作は、従来から使用されているアプリケーションからも利用できるように実装されている。例えば、電子

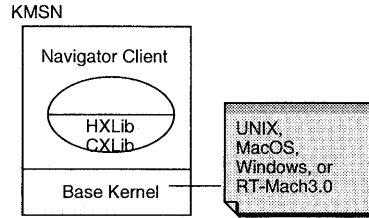


図 6: Heterogenous Operating System Environments for KMSN

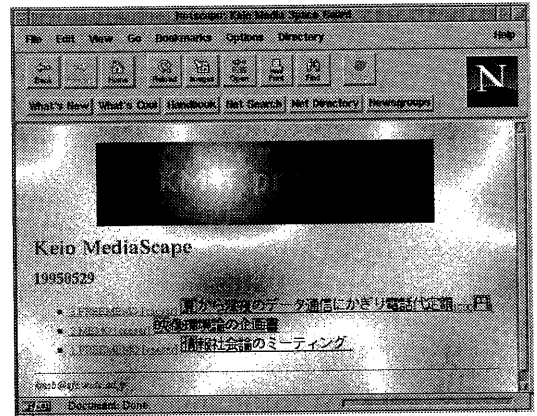


図 7: A Screen Dump of the KEIO Media Space Navigator

メールのプログラム、ニュースのプログラムや Mosaic や Netscape などの WWW のブラウザからも情報にアクセスできるようにしている。図 7 は、ボードサーバに Netscape からアクセスし、最新情報を fetch-it した時のスクリーンダンプである。

## 5.2 クロスプラットフォーム通信機能

各種ハードプラットフォーム上でのプログラム開発を効率良く行なうために、クロスプラットフォーム用の通信ライブラリ群を開発した [11]。低レベルなライブラリ CXLlib は、ボードサーバと KMSN 上のクライアント間でのデータの送受信をサポートし、高レベルなライブラリ HXLlib は、ボードサーバに対する文書データの post-it と fetch-it 機能を提供している。もちろん、漢字コードや改行コードの違いなどもこれらのライブラリによって吸収されている。

### 5.3 情報フィルタリングと情報スケーリング機能

KMSN などのように多数の情報を fetch-it し、それらに限られた物理的なスペース上で表示していくうえでは、いろいろな問題が生じる。情報フィルタリングは、ユーザが指定したフィルタを自動的に実行し、ある限定された範囲の情報だけを選択する機構である。

情報スケーリングは、ディスプレイ上で現在表示可能な情報量を考慮し、ユーザの指定したプロファイルに応じて情報表示の粒度を自動的に調節する機構である。現在、KMSN プロトタイプ上では、テキストレベルでの情報フィルタリングとスケーリングに関する実験が行なわれている [4]。

## 6 まとめ

本稿では、KMSF プロジェクトにおける新しいメディアスペース環境について、そのシステムアーキテクチャに関して解説した。また、KMSF 環境の主な構成要素である KMSB と KMSN についてその概要と現在のプロトタイプの機能を報告した。KMSF 環境については、KMSB や KMSN のプロトタイプシステムを使っての実験がスタートしたが、今後とも、どのように物理的な共同作業空間とネットワーク上で広がる仮想的な共同作業空間をシームレスに統合していくかが、大きな課題である。

なお、KMSB と KMSN のプロトタイプに関する実装結果とその評価に関しては、[8] で報告している。

## 謝辞

KMSF プロジェクトを遂行するにあたり、ご協力いただいた KEIO-IBM パートナシッププログラムのメンバーおよび徳田・村井研究会の皆様には感謝致します。

## 参考文献

- [1] S. A. Bly, S. R. Harrison, and S. Irwin, "Media Spaces: Bringing People Together in a Video, Audio, and Computing Environment," *CACM*, Vol.36, No.1, pp. 28-47 (1993).
- [2] C.F.Goldfarb, Editor: *The Standard Generalized Markup Language(ISO 8879)*, International Organization for Standardization, Geneva, 1986.
- [3] Alan Jones and Andrew Hopper. "Handling Audio and Video Streams in a Distributed Environment".

*Proceedings of the 14th ACM Symposium on Operating Systems Principles*, p231-243. 1993

- [4] 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.
- [5] P. Maes: "Agents that Reduce Work and Information Overload," *CACM* Vol.37, No.7, pp.31-40 (1994).
- [6] 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).
- [7] 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.
- [8] 望月, 富田, 川又, 石川, 徳田, "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", 情処研報, Vol. 95, No. 22, 95-OS-69, (1995).
- [9] H. Tokuda, T. Nakajima, and P. Rao: "Real-Time Mach: Towards a Predictable Real-Time System," *Proc. USENIX Mach Workshop*, pp. 73-82 (1990).
- [10] H. Tokuda, Y. Tobe, S. T.-C. Chou, and J. M. F. Moura: "Continuous Media Communication with Dynamic QOS Control Using ARTS with an FDDI Network," *Proc. ACM SIGCOMM '92*, pp. 88-98 (1992).
- [11] 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.
- [12] M. Weiser, "The Computer for the 21st century", *SCIENTIFIC AMERICAN*, Vol.265, No.3, pp.66-75, September, 1991.