

携帯型パソコン間のマルチメディア通信システムの開発 *

大矢健一 ** ・ 宮下健志 *** ・ 高木和彦 *** ・ 宮澤慶一 ***

Development of a Multimedia Communication System by Notebook PC

Ken'ichi OHYA, Takeshi MIYASHITA, Kazuhiko TAKAGI, Keiichi MIYAZAWA

キーワード: マルチメディア, ノートブック, TCP/IP, Ethernet, プログラミングコンテスト, プロコン

1. はじめに

平成9年度の「第8回 全国高専プログラミングコンテスト」(以下, プロコン)の競技部門は, 2台のノートブックパソコン(以下, ノートブック)をEthernetでつなぎ, マルチメディアの通信することを要求するものであった。

近年, ノートブックは非常に高性能になってきているが, Ethernetを通じたマルチメディア通信の事例は聞いたことがなかった。そのため, 現時点のノートブック間でどの程度のマルチメディア通信が可能かどうかを見極めるという目的で本研究を開始し, プロコンの競技部門に参加することを目指した。

2. 競技部門のルール

プロコン競技部門のルールは毎年6月頃に発表される。これに参加しようというチームは, そのルールをよく理解した上で, 「どのようなシステムにするか, どういう独自の工夫をするか」などを書類にまとめて応募しなければならない。

書類審査により予選が行われ, 予選を通過したチームのみが本選への出場を果たすことになる。6月当初に発表された最初のルールは以下のようなものであった。

「二人一組の競技者が, 森に見立てた競技場に植えられたたくさんの造花の中から制限時間内に指定された造花をいかに早く摘むことができるかを競う。二人の競技者は競技場の内外に分かれ, 審判は外の競技者にどのような造花を摘むかを指示する。競技者はお互いに携えた携帯型コンピュータ間の通信だけを使って情報交換をし, 目的の造花を探して摘む。コンピュータ間の通信には, 競技場に2組独立に設置された10Base-Tの口が利用できる」。

しかしながら, 本番の2週間前になり, 花の他に「年輪」・「布」が追加されて, それについても正しいものを選ぶようにルールが変更された。

3. システム開発の方針

上記のルールに対し, どういうシステムだと競技に勝てるかということがチーム内で議論された。

いろいろな意見が出たが, その結果, 「動画像と音声をお互いにリアルタイムに通信するという手法が, ルールにも適合するし, あらゆる場面にも対応できるだろう」という結論に達した。すなわち, 「動画像+音声のマルチメディア通信」である。

機材の関係から動画像は片方向通信にしたが, 音声については双方向の全二重通信を目指した。

ただ, Ethernetの10Mbpsという帯域の制限を考えると, 実用的に使えるのは数Mbpsという帯域が予想され, おそらくは音声については問題ないものの, 動画像のリアルタイム通信にはある程度の制約が予想された。

4. ハードウェアの選定

システム開発は, ハードウェアに大きく左右される。我々は7月末まではハードウェア選定に専念することとなった。本・雑誌・NetNews・メイリングリスト・WWWなど, 事前にあらゆる情報を調べ, 予算内で我々の目的にうまく合致するハードウェアはないものかを考えた。

4-1 ノートブックとOS

ハードウェアで一番重要なものはノートブック本体であるが, この選定にはOSが大きく関係してくる。

* 本研究は平成9年度教育研究特別経費の助成を受けている

** 電子情報工学科講師

*** 電子情報工学科

原稿受付 1997年10月31日

ノートブックを購入すると Windows95 が搭載されているが、まずはフリーの開発環境が整っている OS をインストールする必要がある。

そのような OS で手軽なものは PC-UNIX であり、代表的なものに FreeBSD と Linux とがあるが、

- メンバーが慣れ親しんでいる
- PAO (モバイルコンピューティングパッケージ) が存在する

という 2 点が決め手となり、FreeBSD (2.2.1R) を用いることにした。

したがって、まずはこの FreeBSD という OS が動くノートブックを選定する必要がある。

4-2 音の入出力

マルチメディア通信として、音の入出力を扱う必要があるため、サウンドボードが必要となる。

ただ、デスクトップ機にはサウンドボードが内蔵されているものと増設できるものがあるのに対し、ノートブックにおいてはサウンドボードを PC カードとして増設できないことがわかった。(近い将来、PC カードとして登場するらしいこともわかった)。よって、サウンドボードを内蔵しているノートブックを選定するしか方法がない。

4-3 動画像の入力

動画像の入力に関してが大問題であった。

非常に高価なノートブックにはビデオ入力端子がついているものの、普通のノートブックにはそんなものはついていない。デスクトップ機でよく行われている方法はビデオキャプチャカードを用いてビデオ入力を行なう方法である。

そのようなビデオキャプチャカードが PC カードとしてもいくつか製品化されていた。そのうち、FreeBSD 2.2.1R がサポートしているのは“IBM Smart Capture Card 2”のみであったので、これを用いることに決めた。

しかし、この PC カードは非常に品薄であり、すでに日本中の店頭および問屋から消えていたが、メーカーに直接問い合わせることにより、なんとか一枚だけ手に入れることができた。

4-4 ノートブック

以上をもとに、ノートブックの選定を行なった。

ノートブックは、競技場内・競技場外の 2 台必要であり、共通して必要な条件は、

- FreeBSD が動く

- 音声の入出力が可能であった。

4-5 競技場内のノートブック

競技場内のマシンとして必要な条件は、「小さい、軽い」であった。調査した結果、以下の製品が候補にあった。

- 東芝 Libretto 50CT
- 東芝 Libretto 60
- 日立 PriusNote 210T
- IBM ThinkPad 310
- 富士通 FMV-BIBLO NC13D

最終的には“日立 PriusNote 210T”に決定した。その理由を以下に示す。

- Libretto 50CT, Libretto 60, PriusNote 210T の 3 つは、FreeBSD での動作が確認できた。
- Libretto には音の入力機能がなかった。
- FMV-BIBLO NC13D の液晶ディスプレイは DSTN であり、他の機種種の TFT 液晶に比べて視認性が劣る。

4-6 競技場外のノートブック

競技場外のマシンとして必要な条件は、

- PCMCIA スロットが 2 つ以上
- 画像の処理に問題のない程度の CPU・メモリである。最終的には、
- 東芝 PORTEGE 300CT
- Panasonic Let's note

が候補に残ったが、画面の広さ (PORTEGE: 1024x600 に対し、Let's note: 800x600) が決めてとなり、“東芝 PORTEGE 300CT”に決まった。

とはいえ、問題点が 1 つだけあり、それは、PORTEGE にはマイク端子がなく内蔵マイクのみということであった。これについては、マシンの改造等により解決しようということになった。

4-7 Ethernet Card

Ethernet Card はそれぞれのノートブックに対し 1 枚ずつ合計 2 枚必要であるが、1 枚はすでに所有していた“IBM Ethernet2”を使用することにした。もう 1 枚であるが、安定性の点で定評のある“3Com Etherlink3”を購入することに決めた。

4-8 ヘッドホンとマイク

歩いたりキーボードをたたいたりしながら会話する必要があるため、ヘッドホンとマイクとが一体となっ

た製品である“SONY DR-200C”を2つ購入した。

4-9 10Base-T ケーブル

実際の競技において必要となるため、10m の10Base-T ケーブルを購入した。

4-10 ノートブックのホスト名

以上のようにして、ハードウェア選定が終了したが、ハードウェアが全てそろったのは、10月12日(日)のプロコン競技部門本番を一月後に控えた9月中旬のことであった。

2台のノートブックのホスト名は、otoko(東芝 Portege), onna(日立 Prius note)と命名した。

5. システム開発

5-1 FreeBSD のインストール

まずは、以下の手順により、新品のノートブックにFreeBSD2.2.1Rをインストールした。PAOについても2.2.1R用のものを使用した。

表1 Windows95におけるスライス

マシン	全容量	FreeBSD	Windows95
Prius (onna)	800MB	500MB	300MB
Portege (otoko)	1.6GB	1GB	600MB

表2 FreeBSDのパーティション

	Prius (onna)	Portege (otoko)
/	30MB	30MB
/home	10MB	60MB
/usr	350MB	740MB
/var	30MB	30MB

表3 XFree86の設定

	Prius (onna)	Portege (otoko)
Mouse	PS/2	PS/2
Keyboard	JIS106	US101
VideoChip	CT65550	CT65554
Display	640x480	1024x600

- (1) 新品のノートブックにはWindows95がインストールされているので、まずは表1のようにWindows95のスライスを縮め、FreeBSDのスライスを確保。
- (2) PAO版ブートフロッピーよりノートブックを起動し、FreeBSD2.2.1R, XFree86 3.2をインストール。パーティションは表2のように切った。
- (3) Kernel等をPAOのものに置き変える。
- (4) Luigi's New Audio Driverを組み込む。

- (5) Portege (otoko) のカーネルコンフィグレーションにSmart Capture Cardに関する記述をする。具体的には、“device scc0 at isa? port 0x240 irq 10 iomem 0xd4000 vector sccintr”の1行を加える。
- (6) 不必要なデバイスを削りkernelをコンパイル。
- (7) /etc/sysconfig および /etc/pccard.conf を適切にエディット。
- (8) /etc/host.conf を書き換え、DNSへのバインドをしないように設定し、/etc/hosts を適切にエディット。
- (9) Portege (otoko) の scc デバイス scc0, sccctl0 を mknod して作成する。
- (10) 表3のようにXFree86の設定。

5-2 Luigi's New Audio Driver

“Luigi's New Audio Driver”とは、Luigi Rizzo氏により開発の進められているLinux OSS 互換の新しいサウンドドライバであり、旧来のFreeBSDのサウンドドライバでは対応していないいくつかのサウンドカード(例えば、ノートブックに搭載されている多くのSound Blaster Pro / Microsoft (Windows) Sound System 互換チップ)にも対応を行なっている。

今回のシステムにおいても、ノートブックに搭載されているサウンドボードがMicrosoft(Windows) Sound System 互換チップであったため、このドライバを用いることにした。

インストールは以下の手順で行なった。

- (1) ftp等によりtar+gzされたドライバを入手し、
- (2) /sys/i386/isa/sndに展開。
- (3) パッチを/sys/i386/isa/isa.cに当てる。
- (4) /sys/i386/isa/sndの中のCファイルについて/sys/i386/conf/files.i386に追記する。その際、古い/sys/i386/isa/soundに関する記述は削除する。
- (5) /usr/include/machine/soundcard.hを、新しいドライバのものと置き換える。
- (6) カーネルコンフィグレーションに、“device pcm0 at isa? port? tty irq 0x530 tty irq 5 drq 1 vector pcmintr”と記述し、カーネルをコンパイルおよびインストールする。
- (7) デバイス snd0 を MAKEDEV する。

以上の作業により、新たなドライバがカーネルに組み込まれ、音声の録音・再生が可能になる。

さて、音を全二重で通信するにはドライバのサポートが必要である。

今回使用したマシン (prius, portege) では、それぞれ CS4236, OPL3-SA_x という Sound Blaster Pro / Microsoft Sound System 互換の音源チップを使用していることがわかったが、この2つのチップに関しては、残念ながら、全二重通信をサポートしていないことがわかった。そのため今回は音を全二重で通信することができないことがわかった。

6. ソフトウェアの開発

ソフトウェアを開発にするにあたり、まずは、目的を満たすソフトウェアとしてソース付きでフリーのものがないかどうかを調査した。

6-1 動画像通信のソフトウェア

動画像通信に関しては、xsc_c というソフトウェアが目的に合致することがわかった。これは、“Smart Capture Card 2” から動画像を読み込み、X 上の画面に表示するものである。

この xsc_c に対して、以下の改良を加えた。

- -display オプションの追加
ネットワーク上の他のマシンの X サーバへ接続して画面を表示する。これにより、2 台のノートブック上に同じ動画像が表示される。
- 静止画像の表示
ボタンをクリックすることにより、新たなウィンドウ上に静止画像を表示する。静止画像は、Prius 側に 4 画面、Portege 側に 1 画面表示される。

また、動画像表示は、30frame/s は無理であったため 2frame/s としたが、実用上は何も問題がなかった。

6-2 音声通信のソフトウェア

音声通信に関しては、Speakfreely-6.1c が使えることがわかった。Speakfreely は

- 音声受信・再生ソフトウェアである sfspeaker
 - 音声取り込み・送信ソフトウェアである sfmike
- の2つから成り、半二重通信をサポートしている。

この Speakfreely に対して、以下の改良を加えた。

音の入出力デバイスである /dev/audio の調子が悪く音に歪みが出てしまったため、入出力デバイスを /dev/dsp に変更するために soundbyte.c の `define SoundFile "/dev/audio"` という記述を `define SoundFile "/dev/dsp"` に変更した。

6-3 その他のソフトウェア

- painter

今回、マウスをクリックすることにより簡単なメッセージを通信するためのアプリケーション painter を自作し、音声・画像通信にトラブルが発生した場合への備えとした。機能を以下に示す。

- (1) Portege 側にはメッセージの一覧があり、競技者はそのなかの適当なものをマウスでクリックする。
- (2) Prius 側では、クリックされたメッセージが“メッセージボックス”の中に表示される。

• okphone

okphone は、ひらがなで talk をおこなうアプリケーションである。これについては改良の余地がなく、そのまま用いることにした。

7. 完成したシステムについて

7-1 システム全体の概観

完成したシステム全体の概観を図 1 に示す。

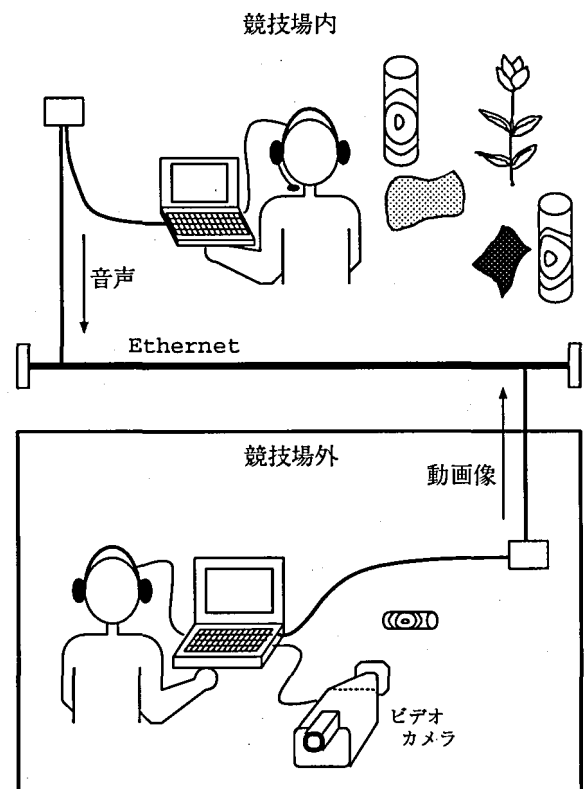


図 1 システム全体の概観

また、双方のノートブックに表示される画面の概要を、図 2、図 3 に示す。

7-2 動画像・静止画像の転送

ビデオカメラに映っている画像データを「競技場外」（以下、「外」）から「競技場内」（以下、「内」）に

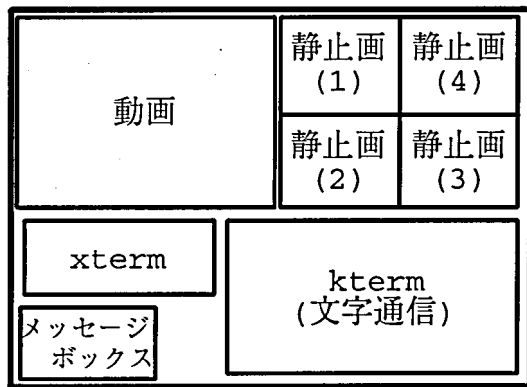


図 2 表示画面の概要 (競技場内)

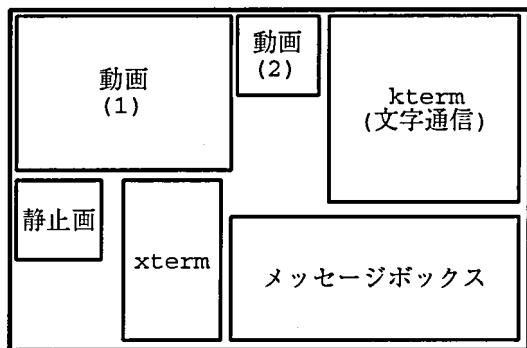


図 3 表示画面の概要 (競技場外)

転送することができる。Ethernet 上の TCP/IP における通信は Xlib のレベルで対応している (図 4)。

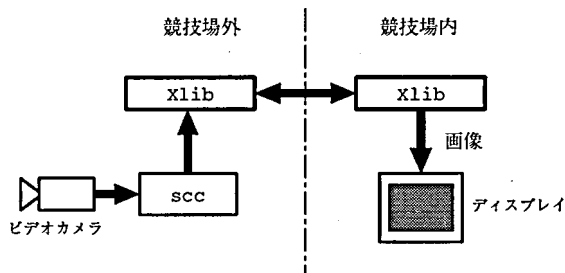


図 4 画像の通信

このとき、動画は「外」と「内」の両画面上に表示されており、「外」の競技者がボタンを押すことでその瞬間の画像を静止画として「内」に送ることもできる。なお、静止画のサイズは動画のサイズ (320x200) の 1/4 (160x100) であり、「内」には最大 4 つ、「外」には最大 2 つの静止画を表示できる。

7-3 音声通信

「内」から「外」に対し、常に音声で情報を伝えることができる (図 5)。

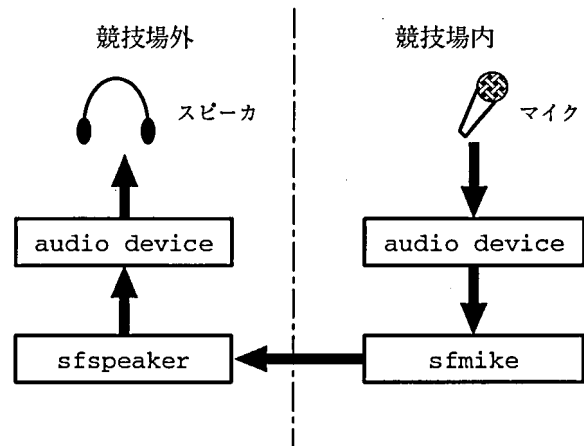


図 5 音声の通信

7-4 文字による会話

文字通信専用のウィンドウがあり、「内」と「外」でひらがなを用いた双方向の文字通信が可能 (図 6)。

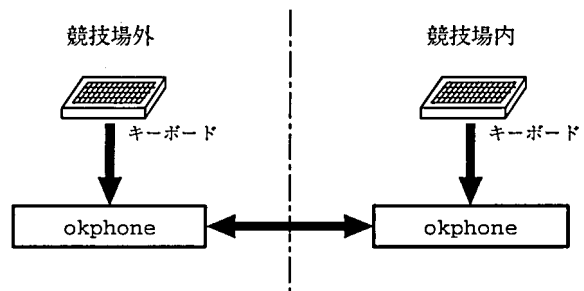


図 6 文字による通信

7-5 簡単なメッセージの転送

メッセージ転送用のウィンドウ内に、あらかじめ複数の簡単なメッセージ (yes や no など) が用意してある。このメッセージをマウスでクリックすることにより、そのメッセージを相手側に転送することができる (図 7)。この機能は「外」から「内」方向のみ可能となっている。

7-6 実際の動作例

実際の動作例を図 8、図 9 に示す。

8. システムの使用方法

システムの使用法について、簡単に述べる。

- (1) Portege, Prius を起動し、X を立ち上げる。
- (2) 双方をハブに接続。
- (3) Portege から Prius に、okphone で呼びかける。
- (4) Prius は okphone を起動し、同時に xhost を実行して画像・メッセージ双方の受け入れ準備。

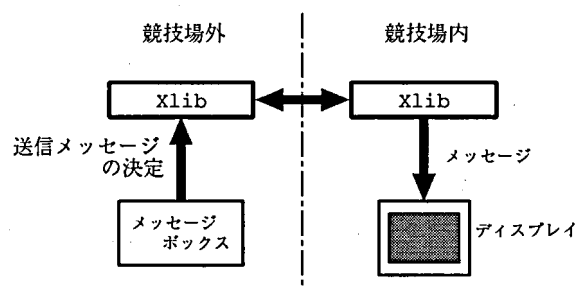


図7 メッセージの通信

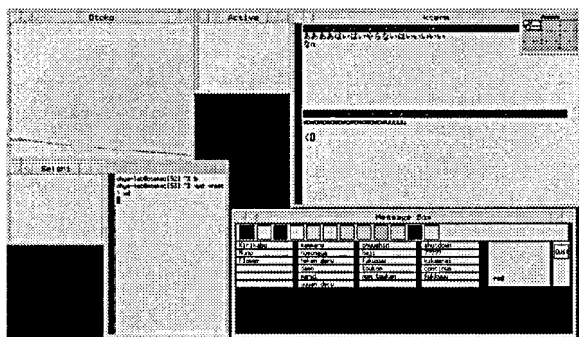


図8 実際の動作例（競技場外）

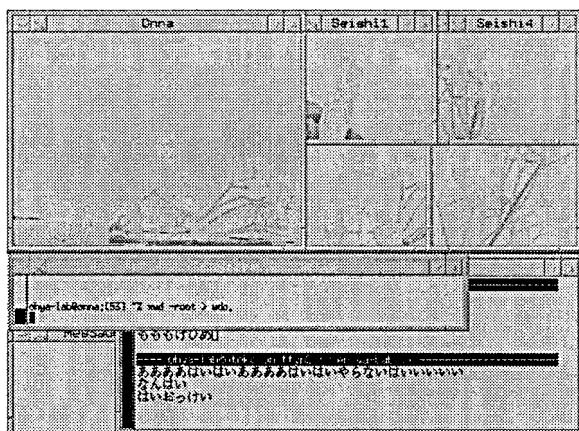


図9 実際の動作例（競技場内）

- (5) okphoneの接続完了が確認できたら、Portegeはxsc, painterを起動して画像の送信を開始。また、同時にsfspeakerを起動し、Prius側からの音声等待。
- (6) Priusでは、画像が送られてきたのを確認したら、sfmikeを起動して音声の送信を開始する。

9. 問題点と今後の対策

以上のようにシステムは完成したが、ときとしてシステムの動作が不安定な状況に陥ることもあるので、

以下はそれについて述べる。

● scc の不安定性

xscを終了して再起動した場合、xscの画面が乱れることが頻繁にあった。この原因については解析中である。復旧するにはSmart Capture Cardを差しなおすなどしてカードを初期化するしか方法がなく、その際の所要時間は約45秒である。

また、すでに音声通信が行なわれている状態でxscを起動しようとする、そのままマシン自体が止まってしまうことがよくあった。これについても原因を解析中である。この場合はマシンの電源を切ってからまたブートしなおすという作業が必要である。

● sfmike の不安定性

動画・音声Ethernet上で通信する場合には、ネットワークに強力な負荷がかかるため、音声通信においてもパケット再送が繰り返されることになる。そのため、時間の経過により音声の遅れが大きくなり、最後にはsfmikeが停止してしまうこともあった。

このことから、動画・音声をリアルタイムに快適に通信するためには、両者を統一して考慮する必要があるように思われる。

今後の対策としては以下のことが考えられる。

- 安定した通信のため、動画・音声を統一的に扱う環境の構築
- 音声の圧縮による通信

10. あとがき

競技本番の1週間ほど前から、本番を想定したリハーサルを毎日1時間ほどずつ行なった。

リハーサルでは実際に、布・年輪（競技場内、外）・造花を十数種類ずつ用意し、競技場内外それぞれ別の部屋に分かれて本格的な練習を行なった。なお、それぞれの布・年輪には、極めて似通ったものを用意した。

また、リハーサルによって問題点を洗い出し改善を計った。さらには、PCを持ちやすいような専用器具も製作し、時間短縮に努めた。

以上、万全の態勢を整えてはいたが、3回戦の本番中に突然、競技場内側のPriusがハングするというトラブルが起きた。これは、長時間におよぶリハーサルにおいて一度も起きなかった出来事である。これにより、本番中に再起動を余儀なくされ、3回戦で敗退してしまった。目指していた優勝はおろか入賞も果たすことができず、極めて残念である。