

ELIS 8200 における Xウィンドウシステムの開発

7N-1

松山 憲治^{*} 中松 芳樹^{*} 前野 芳史^{*} 山田 勝彦^{*} 宇田川 智之^{**} 長坂 篤^{*}

^{*} 沖電気工業(株)総合システム研究所 ^{**}(株)沖テクノシステムズラボラトリ

1 はじめに

我々は、Lisp マシン ELIS 8200 上の Xウィンドウシステムを開発した。ELIS 8200 では、FEP(Front End Processor)として、OS/2 が稼働する IF800/EX (CPU は 80386) を採用している。これは、今後、ビジネス分野で主流になるであろう OS/2 と、Lisp マシンとしての機能の両方をユーザに提供している。

しかしながら、ワークステーションとしてはネットワーク環境が重視されるため、OS/2 上で、ネットワーク (TCP/IP)、および、Xサーバーが動作し、UNIX 系のマシンとも融合できるように設計されている。

本稿では、ELIS 8200 Xウィンドウシステムの構成、および、OS/2 上への Xサーバーの移植について述べる。

2 ELIS 8200 Xウィンドウシステムの構成

ELIS 8200 Xウィンドウシステムは、ELIS CPU 側で動作するクライアントと、FEP 側で動作する Xサーバーとに分けられる(図1 参照)。ELIS CPU と FEP 間の通信には、通常の Xウィンドウのクライアント-サーバー間の通信と同じように Xプロトコルを用いている。

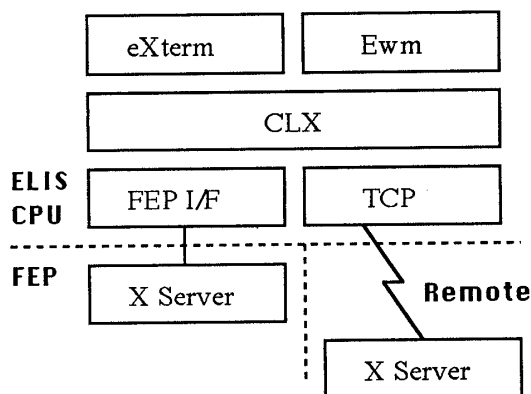


図1

ELIS CPU 側では Xlib に相当する CLX の上に、ターミナルエミュレータ (eXterm)、ウィンドウマネージャ (Ewm) が独立したクライアントとして動作しており、これらは、ネットワークを通して他の Xサーバーへもアクセス可能である。どの Xサーバーにアクセスするかは、CLX で吸収しており、FEP 上の Xサーバーとのインターフェース部分を CLX に追加した。

3 OS/2 上での Xサーバー

ディスプレイは、カラー (4096色中16色) またはモノクロ (16階調) を選択可能であり、解像度は両方共 1120×754 である。

OS/2 上で、Xサーバーは独立した1つのセッションとして動作する。このため、他のセッションで別のプログラムを走らせながら、Xウィンドウが使用可能である。また、PM(Presentation Manager) も1つのセッションで動作するため、マルチセッション、マルチ・ウィンドウシステムという形態を取ることが可能である(図2)。

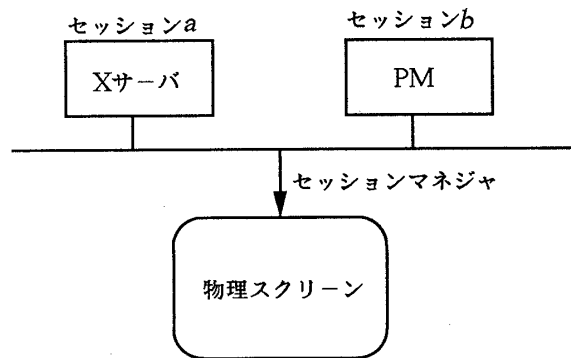


図2

以下に、移植上の主な変更点、問題点をあげる。

(1) OS 依存部分

提供されているものは UNIX 系のものしかないので、OS/2 対応に作成する必要があった。

Development of X Window System on ELIS 8200

Kenji MATSUYAMA^{*} Yoshiki NAKAMATSU^{*} Yoshifumi MAENO^{*} Katsuhiko YAMADA^{*} Tomoyuki UDAGAWA^{**} Atsushi NAGASAKA^{*}

^{*}Systems Laboratories Oki Electric Industry Co., Ltd. ^{**}Oki Technosystems Laboratory, Inc.

(2) Cコンパイラ

マイクロソフト社のコンパイラ MS-C が 80386 をまだサポートしていないため、32ビット CPU としての 80386 の機能が生かせていない。また、X のプログラムで暗に 32ビットマシンを仮定している所があり、その部分の変更が必要であった。

(3) 描画部分の変更

本 X サーバでは、VRAM (4MB) 上に直接描画するという形態を取るため、描画の基本的な関数は書き直した。ここで、IF800/EX がもつ、VRAM 上のデータをラスターオペレーション付きで転送する機能(以下 DMT と呼ぶ)を生かし、大部分の描画には DMT を用いた。DMT は 80386 とは独立に動作するため、CPU の負荷の軽減にもなっている。特に、フォントはオープン時に VRAM 上にロードし、描画は DMT で行う方式を取っているため、文字の描画は高速である。なお、X ウィンドウで提供されている漢字フォント (K14) の他に、内蔵 ROM の漢字フォント (28×26) もサポートしている。

4 クライアントの実現

ELIS 8200 でのクライアントの実現においては、UNIX 上でのクライアントとの相違から、以下の点を考慮する必要があった。

(1) プロセス

クライアントを新たなプロセスとして実行する場合は、TAO の関数 process-fork により、子プロセスを生成し、実行する形態をとった。

(2) ポインタのボタンの個数

ELIS 8200 のポインタは 2 ボタンであるため、3 ボタンをデフォルトにしているクライアントは 2 ボタン対応に変更した。

4.1 ターミナルエミュレータ

ELIS 8200 では、X ウィンドウのターミナルエミュレータ xterm と同等の機能をもつ eXterm を実現した。

eXterm の実体は、入力ストリームかつ出力ストリームであるオブジェクトである。eXterm は TAO のストリームのクラス階層の中の1つのクラスとして実現されており、既存のクラスの上に、eXterm 固有のメソッドを追加している。

eXterm が起動されると、ストリームの変数 (*standard-output*, *standard-input* など) にクラス eXterm のインスタンスがバインドされ、トップレベルが動作する。エディタなど通常のターミナ

ルで動作するプログラムは変更なしに eXterm 上でも動作する。

また、ELIS では漢字コードとして EUC コードを採用しているが、ネットワーク上の EUC コードと異なる漢字コードを採用しているマシンを使用する機会が多いため、eXterm では、EUC コードの他に、JIS、および、シフトJIS の漢字コードをサポートしている。この変換は、追加したメソッドの中で行っており、ストリームの拡張機能である。

4.2 ウィンドウマネージャ

UNIX マシン上の X ウィンドウシステムで広く使われているウィンドウマネージャとして uwm がある。ELIS 8200 では、uwm と同等の機能を持ち、Lisp 用に拡張したウィンドウマネージャ Ewm (ELIS Window Manager) を実現した。

uwm では、メニューから UNIX のシェルコマンドが実行可能であるが、Ewm では任意の Lisp 関数が実行可能である。なお、Ewm の初期設定ファイルの構文は、uwm のそれが継承できるように、同一にしている。

Ewm は、Common Lisp にはない機能(プロセスフォークなど)を除き、Common Lisp で記述しており、他の Lisp 処理系への移植も容易である。

5 おわりに

本稿では、ELIS 8200 上のウィンドウシステムについて、まず全体構成を示し、次にクライアントとサーバに分けてそれぞれ述べた。

現在、ELIS CPU 側では CLX を使ったクライアントを開発しているが、今後、言語に CLOS が実現されるのに続いて、ツールキットの開発、さらに、統一されたユーザインタフェースの実現を行う予定である。

参考文献

- [1] Sheifler, Gettys, Newman: "X Window System", Digital Press, 1988
- [2] Jones: "Introduction to the X Window System", Prentice Hall, 1989