

## 3 J-2

## YyonX: 概要設計

井田昌之<sup>†1</sup>, 古坂孝史<sup>†2</sup>, 田中啓介<sup>†1</sup>  
 青山学院大学情報科学研究センター 研究教育開発室

## 1 始めに

YyonX は青山学院大学において現在 X-window 上に試作を進めている Yy Window Tool Kit である。その要求仕様については [1] に示した。

YyonX は、Yy ウィンドウのパイロットシステムであり、それにより Yy の評価データを得ることも目的としている。現在、SUN-Common Lisp, Allegro Common Lisp, KCL の上で動作している。

YyonX では Yy の持つ次の特徴を実現する。1) CLOS の全面的な採用、2) output recording 機能及び文脈依存型のマウスセンシティブリティ。また、CLOS の Meta Object Protocol により実行性能を向上させる。

## 2 Yy の実現 - サーバクライアントモデル -

Yy は YYAPI (Yy Application Program Interface), YYWS (Yy Window System), NWSI (Native Window System Interface) の三つのレイヤから成る。YYAPI は、利用者が使用するライブラリプログラム群である [4]。YYWS は Yy ウィンドウを特徴付けるオブジェクトを管理する部分である [2]。NWSI はベースとなるウィンドウシステムとのインタフェース部である。

これを X window 上に、Yy-server と Yy-client の二つのプロセスとして実現する。

Yy-server は、二つの部分よりなる [3]。デバイス依存部は Yy の NWSI にあたる。デバイス独立部は Yy YYWS のうちカーネル部に当たる。

Yy-client は、二つの部分よりなる。YYWS のアプリケーション部と YYAPI である。

この関係を図 1 に示す。Yy-server と Yy-client はそれぞれ X のクライアントとして実現する。それらは同一マシン上でもまた、ネットワーク上の別マシンでもよい。

## 3 動作環境

YyonX は、次の要件を満たすシステムで動作する。

1) フル Common Lisp, 2) X Window R11 (CLX は用いない), 3) CLOS 処理系 (当面は PCL)。

## 4 YyonX の構成上の特徴

## 4.1 Meta Coordinate system

X window それ自身は左上を原点とする座標系を持っている。一方、グラフィックシステムの多くは左下を原点としている。アプリケーションによって原点をウィンドウ生成時に選択できるようにした。継承するクラスの変更によって実現するので実行速度の低下はない。

Design of Yy on X, by Masayuki IDA<sup>†1</sup>, Takashi KOSAKA<sup>†2</sup>, Keisuke TANAKA<sup>†1</sup>

<sup>†1</sup>Aoyama Gakuin University, <sup>†2</sup>Aoyama Gakuin University/CSK Corp.

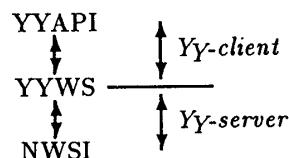


図 1: サーバ/クライアントモデルによる YY 三階層の実現

## 4.2 Output Recording とその動的制御

[1] に示した解析に従って Output recording 及び presentation を導入する。

Presentation は、CLOS のクラス概念により構築した。表示されたオブジェクトは present した時点のクラスに属するものとし、暗黙のクラス継承は行なわないものとした。また、記録するオブジェクト数の上限を設定できるようにした。

## 4.3 Common Windows レベルの機能水準

用意する描画機能の数は、極力、合理的な範囲で多く持つこととした。また、既存のアプリケーションソフトへの便利さと Common Lisp 上での共通性を考えた結果、Common Windows レベルの機能を用意した [4]。

## 4.4 インプットエディタとマルチナショナル機能

一つには実装フォントの問題がある。YyonX では選択可能な様にしている。

また、日本語変換のためのフロントエンドの問題がある。フロントエンドとしては、二通りを考慮している。Wnn Jserver と独自の交換機構の組み込みである。

フォントは Common Lisp において character 型の構成要素として既に組み込まれている概念であるが、現在、ANSI X3J13 で出されている最終案に従う。

## 5 基本クラス

## 5.1 Position 及び Region

もっとも基本的なオブジェクトとして Position と Region を置く。Position は、XY 座標を表す基本オブジェクト、Region は、矩形領域を表す基本オブジェクトである。

region は論理的に六つのスロットを持つ。:top, :bottom, :left, :width, :height, :right である。これにより LTRB でも LBWH でも扱えるようにした。

## 5.2 active-region

マウスセンシティブリージョンである。enter-region と exit-region およびその中でのマウスメソッドを指定できる。

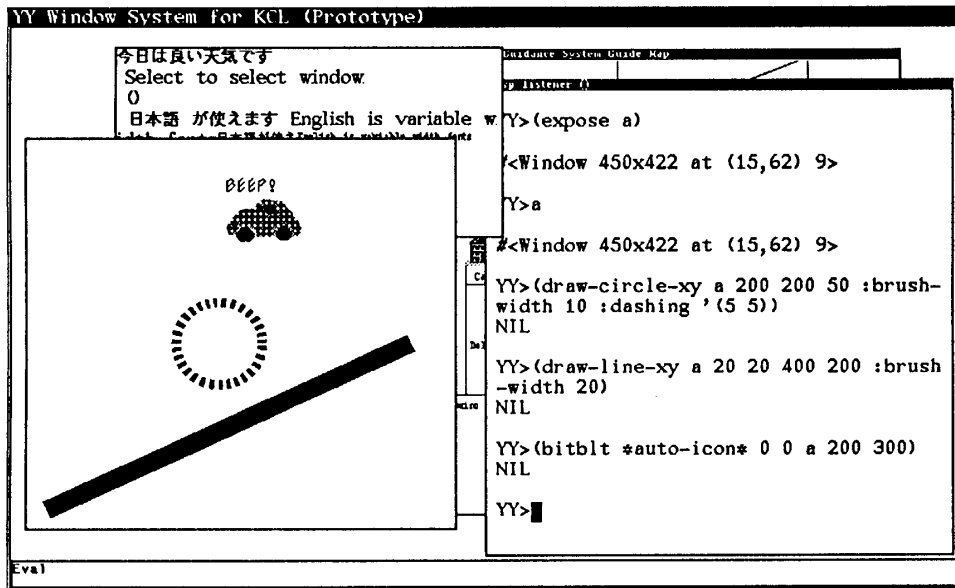


図 2: YY window 表示例

Presentしたオブジェクトだけをセンスできる presentation とは別に、利用者が領域を指定できるような active-region をおいた。

### 5.3 Stream オブジェクト

Yy への入出力はストリームの概念による。これは X3J13 の修正案に沿った stream type の拡張であり、CLOS により実現する。

graphic-stream、そのサブクラス window-stream と bitmap-stream が導入するストリームオブジェクトの中心である。

一方、非同期に発生する入力イベントに対するストリームとして Event-stream を用意する。Event-stream にはマウスカーソル/ボタンのクリックに関連するイベントステータス、キーボード割り込みに関連するイベントなどが通る。

### 5.4 World 及び Viewport

表示対象を World と呼ぶ。そのうち、画面に現われている部分を Viewport と呼ぶ。

### 5.5 Stipple

bitmap-stream 操作の対象として存在するビットマップを Stipple と呼ぶ。Stipple は、world との間でやり取りをすることができる。

### 5.6 Window の構造

Window は、title と frame より構成する。Title は無くとも良い。title 及び frame には、アクセサリが付けられても良い。frame は viewport あるいは page である。frame の縁には幅があって良い。frame はスクローリングをサポートする。viewport scrolling は上下左右に移動できる。page scrolling は上下のみに移動できる。

休止状態のウィンドウはアイコンとなり得る。

ルートウィンドウの中に任意に複数のウィンドウを表示できる。この例では、四つのウィンドウを表示している。複数のワールドを同一のルートウィンドウの中で扱える。また、同時に複数のルートウィンドウを起動しても良いが、それらは独立したプロセスとなる。それらの間の通信は利用者が陽に指示する。ルートウィンドウにはその最下部にガイドウィンドウが必ず付けられる。これは、input editor、システムからのガイド/警告メッセージの表示、日本語フロントエンド等で使用される。

## 6 使用感覚

Yy は、Initialize-yy により起動する。これにより、ルートウィンドウと呼ぶウィンドウがあらわれる。ルートウィンドウに対しては X-window の制御が有効となる。ルートウィンドウにはその最下部にガイドウィンドウが必ず付けられる。また、自動的に一つの Lisp Listener ウィンドウがとられる。エディタの呼び出しでは、その利用者が設定した初期化ファイルで GnuEmacs を呼び出す。図 2 にプロトタイプでの画面の例を示す。

各ウィンドウに対してスクロールバーと標準ポップアップメニューが用意される。Viewport frame に対しては、左と下にバーが出る。Page frame に対しては、左にバーが出る。標準ポップアップメニューには、Shrink (Expand), Move, Reshape, Clear, Expose, Bury 等の基本的な操作が含まれる。ルートウィンドウには、更に全体に対して必要な基本操作を含むメニューが用意される。

## 7 まとめ

YyonX の開発は 1989 年 4 月に開始し、10 月に一応のプロトタイプが動き出している。評価を行ないながら、完成を目指した開発を継続しているところである。また、今後は、X-window 以外への Yy の実現を踏まえた再評価も行ないたい。

## References

1. M.Ida, et. al. : "A Requirement Analysis for a Portable Window System on Top of Common Lisp" IPSJ annual conf. 1990 March
2. T.Kosaka, M.Ida, K.Tanaka: "YYWS : the window toolkit for YyonX" ibid.
3. K.Tanaka, T.Kosaka, M.Ida: "Design of Yy-server for YyonX" ibid.
4. M.Ida: YYAPI External Specification Manual v1.0 1989 Dec.