

MathBlackBoard からの OpenXM の利用

出口 博章

神戸大学大学院 国際協力研究科*

DEGUCHI HIROAKI

GRADUATE SCHOOL OF INTERNATIONAL COOPERATION STUDIES, KOBE UNIVERSITY

1 はじめに

MathBlackBoard[1] は、1997 年に松嶋 [2] が Java を用いて作成した黒板アプレットを、改良・発展させた Java プログラムである。

黒板アプレットは簡単な数式処理システムであり、数式の入力と操作、数式処理、そしてグラフの描画などの機能を備えている [3]。黒板アプレットの最も特徴的な点は、数式を直感的に操作するためのユーザインタフェースである。

MathBlackBoard では、黒板アプレットの優れた GUI を受け継ぎながら、数式を入力・編集・操作・処理するための、さらに使いやすいユーザインタフェースの実現を目指している。GUI 操作の細かな変更や新たな操作の追加、また、外部の数式処理エンジンに接続するための機能の追加などを行なったが、ここでは OpenXM の利用について述べる。

2 OpenXM サーバの用意

OpenXM サーバとしての ox_asir については、cygwin を利用して Windows で動くようにコンパイルされたものを福岡教育大の藤本氏からテスト用にいただいた。そのファイルなどを元にして、サーバ側のファイルを以下のように構成した。

CYGWIN1.DLL, OX.EXE

変更なし。

OX_ASIR.EXE

元は ASIR.EXE という名前のファイルが、OX_ASIR.LNK というシンボリックリンクから呼び出されていたが、拡張子が EXE でないとうまく動かなかったため、ASIR.EXE 自体を OX_ASIR.EXE とリネームした。

OX_ASIR.BAT

アイコンのクリックで ox_asir が起動するように、以下の内容を記述したバッチファイルを用意した。

```
ox -ox ox_asir.exe
```

*deg@kobe-u.ac.jp

3 OpenXM クライアントの用意

MathBlackBoard に OpenXM クライアントとしての機能を持たせるために、OpenXM のソースコードに含まれる Java のファイルを利用した。以下に簡単な手順を述べる。

まず、OpenXM HEAD ページ¹⁾内の「Source Distribution (Download)」の「2. Semi-HEAD branch is available from our ftp site: [Source (Shift+Click)].」からリンクされている `openxm-head.tar.gz` をダウンロードして、展開する。

そして、展開したファイル内の `OpenXM/src/OpenMath` に移動する。`OpenXM/src/OpenMath` には `ORG` フォルダがあるので、その位置で以下のコマンドを実行する。

```
javac -target 1.1 ORG\openxm\tam\OpenXM.java
del ORG\openxm\tam\*.java
jar cvf OpenXM.jar ORG
```

`javac` コマンドのオプション `-target 1.1` は、JDK1.1 をターゲットとしてコンパイルされている MathBlackBoard にあわせるために付加している。

最後に、上の手順で得た `OpenXM.jar` を適当な位置に置き、その位置をシステムの変数 `CLASSPATH` に追加する。以上の手順によって、Java プログラム中から `import` して OpenXM を利用することが可能となる。

`OpenXM.jar` を使用して OpenXM サーバを利用する方法は、`OpenXM\src\OpenMath\` の `testclient.java` を参考にした。

4 問題点と解決方法

既にあるソースコードを利用することにより、容易に OpenXM サーバへアクセス可能となったが、`OpenXM.jar` の利用においては問題点もある。

OpenXM サーバとの通信で得られた文字列を取り出そうとすると、MathBlackBoard の処理には不要な情報も含んだものが返される。これは `OpenXM.jar` 内のプログラムの仕様上仕方のないことなので、MathBlackBoard 側で調整することとした。

例えば、 $(x-1)^2$ を実行した結果は `(CMO_STRING,9,x^2-2*x+1)` という文字列で返される。利用するために必要な情報である $x^2-2*x+1$ 以外に、情報の種類と文字数についての情報が含まれている。他の例では、 $x/2-2/3*x$ に対して `(CMO_STRING,6,-1/6*x)` が、また `fctr(x^2-1)` に対して `(CMO_STRING,23,[[1,1],[x-1,1],[x+1` が返される。

これらの結果から、「二つめのコンマの次の文字から、最後から二番目の文字まで」を抜き出す文字列操作を行なうことによって必要な情報を得られることが分かる。

本来であれば `OpenXM.jar` の中に含まれるソースコードに手を加えて、結果が取り出しやすくなるように変更すべきなのだが、今回は「二つめのコンマの次の文字から、最後から二番目の文字まで」というパターンのみで対応できるため MathBlackBoard 側で対応することとした。

ただし、因数分解の場合は `[[1,1],[x-1,1],[x+1,1]]` のままでは MathBlackBoard で扱えないため、別途 `asir` に関数を読み込ませるなどの工夫が必要となる。

¹⁾<http://www.math.sci.kobe-u.ac.jp/OpenXM/1.2.2/index.html>

5 今後の課題

まず、因数分解の結果を MathBlackBoard で利用可能な形式で出力するための関数を起動時に読み込むように変更しなければならない。起動時に外部の関数を読み込むように設定しておけば、将来的に他のパターンにも対応可能となる。

また、現状では OX_ASIR.BAT をダブルクリックして ox_asir を起動させておいた状態で MathBlackBoard を実行する必要があるため、MathBlackBoard でのカーネル切り替え操作に連動して OpenXM サーバが起動するような仕組みを整えたい。

それらの基本的な仕組みが整ったところで、利用可能な操作の種類を増やすことや、グラフィックス出力を取り込むことについての検討を行ないたい。

参 考 文 献

- [1] 出口博章: MathBlackBoard, 数式処理 Vol11 No.3,4, pp.77-88, 2005.
- [2] 松嶋純也: Java を用いた使いやすい数式処理システム, 神戸大学大学院教育学研究科 修士論文, 1998.
- [3] 出口博章: 黒板アプレット, 数式処理 Vol9 No.1, pp.32-37, 2002.