

プログラマのデスクトップ再訪

佐原伸

URL: <http://www.sra.co.jp/people/sahara>

URL: <http://www.parkcity.ne.jp/~sahara>

E-Mail: sahara@sra.co.jp

E-Mail: sahara@parkcity.ne.jp

昨年のSEAMAILに「プログラマのデスクトップ」を書いたが、最近、僕のデスクトップ環境（持ち歩いているPowerBook 3400/240と家のPowerMac 9500/120）が激変しはじめたので、報告する。
仮想的なOSや言語の改良が進み、ユーティリティ・ソフトも便利なものが増えてきて、今、Macが旬である。

1.0 関数型言語のその後

昨年紹介した、関数型言語Concurrent CleanのPowerMac対応版コンパイラである1.2.4版が発表され、さっそくダウンロードしてみた。

4年前の玉井さんのSun3/60上のK C Lによるパズル¹のプログラム²は、最初の解で停止するもので約100秒かかった。昨年の僕のPowerMac 9500/120上のMCL版のプログラムは全部の解を求めるのに51秒で、玉井さんのプログラムに比べ約100倍速くなっていた。この時の、PowerMac 9500/120上のConcurrent Clean 68000版は41秒で、PowerMac用のコンパイラであるMCL版よりやや速かった。

同じプログラムを、PowerMac 9500/120上の最新1.2.4版で（PowerMac対応）実行すると7秒で、PowerBook 3400/240では5秒になった。玉井さんのプログラムに比べ、約1000倍速くなっていることになる。来週には、PowerMac 9500のCPUを200MHzにアップグレードするので、さらに速くなるはずである。

実は、このプログラムはもっと速くできる。バージョンアップしたことで一部言語仕様が変わったため、リストのi番目の要素を参照するやり方が変わったようで、ざっとマニュアルを見たところでは新しいやり方が分からなかった。そこで、一旦リストから配列に返還してから計算を行っているので、このプログラムは多少効率が悪いのである。それに、ヒープ領域やスタック領域なども調節できるのだが、そのあたりもあまり考えずに実行している。

プログラムは以下のようにした。

1. パズルの問題は、次の に1から9までの数字を重複無く入れて、等式を成り立たせるようにするというものである。

$$= 1$$

2. SEAMAIL (Volume 8, Number 4) 参照

```

module tamaiReal
import StdEnv, StdList, ExtendList

Start :: [[Int]]
Start = filter isIt (allPermutations [1..9])
where
  allPermutations:!.[a] -> [[a]] / 順列生成 */
  allPermutations [] = [[]]
  allPermutations [x:xs] = concat (map (interleave x) (allPermutations xs))

  isIt :: [Int] -> Bool / 解判定 */
  isIt x = (mySum (ToArray x) >= 0.99999999) && (mySum (ToArray x) <= 1.00000001)
  where
    mySum :: {Int} -> Real / 3項の和の計算 */
    mySum x = (addThreeRationals x.[0] x.[1] x.[2]) +
              (addThreeRationals x.[3] x.[4] x.[5]) + (addThreeRationals x.[6] x.[7] x.[8])
    where
      addThreeRationals: Int Int Int -> Real / 項を一つ計算 */
      addThreeRationals x y z = (toReal x) / (10.0 * (toReal y) + (toReal z))

  ToArray :: [e] -> .(a e) | Array a & ArrayElem e 配列へ変換 */
  ToArray list = {e \\ e <- list}

```

図 1. パズルのプログラム (Concurrent Clean 1.2.4版)

2.0 OS

昨年の僕のデスクトップでは、Mac OSしか動いていなかったが、今は日本語版Windows 95とLinuxがPowerMac 9500上で動いている。

Windows 95はペンティアムをエミュレートするVirtualPC上で動いていて、Office97上のMS WordやExcelで作成された九大病院のドキュメントを手入れするのに役立っている³。九大病院では、1台のPowerBook 3400/240上で固定IPアドレスのMacと、DHCPサーバー利用のVirtualPCとが、一本の10BASE-Tのケーブルで、論理的には2台別々にネットワークに繋がっている。

VirtualPCは単にMacのアプリケーションであり、VirtualPCからMacのフォルダーやファイルとSoftWindows形式のディスクが扱え、しかも、実行中にマウスをVirtualPCのウィンドウ外に動かせば、MacとWindowsの環境を行き来でき、両方のアプリケーションを同時実行もできるので便利である。

LinuxはMkLinuxというPowerMac用のものだが、まだデバッグモードで動いているにも関わらず、昔VAXやソニーのNEWSでUNIXを動かしていた身からすれば、とても快適に動く。MkLinuxは起動時にMacOSとして起動するかMkLinuxとして起動するかを指示するので、VirtualPCほど便利ではないが、GNUのソフトがほとんど標準でインストールされているし、その他のフリーソフトやシェアウェアも多いので、総合力は一番ある。LinuxDiskというシェアウェアを使えば、MacOS側からLinuxやBeOSやA/UXのファイルを簡単にアクセスできるので、気の多いプログラマには都合がよい。

3. といっても、本当に実用になるのは、CPUを200MHzにしメモリーを208MBにする来週のはずである。今は、まだ何とか動く段階である。ただしメモリーは128MB程度で十分であろう。

将来は、古いPowerMacはLinuxとして各種のサーバーに使おうと思っている。また、FreeBSDの系統のUNIXが68000系Macで動くので、古い68000系Macも捨てられない。

また、形式仕様言語やツールでLinux上で動く物が多いので⁴、それらを動かすことも狙っている。

3.0 ネットワーク環境

近所のケーブルテレビ会社がインターネット接続サービスを始めたので、さっそく繋いだ。実効速度で30Kbpsから60Kbpsでファイル転送などができるのでありがたいが、接続する機器のMACアドレスを固定されるので、家のPowerMac 9500しか接続できない。イーサネットに繋がっている他の3台のMacとは、人力で10BASE-Tのケーブルを切り替えるという石器時代の手法を用いている。他のMacも繋ぐにはPowerMac 9500上にもう1枚のEthernetカードを装着し、インターネット・ゲートウェイ・ソフトウェアをインストールしなければならぬので、大蔵大臣に内緒で裏金作りに励んでいる⁵。

まあ、そうは言っても速いことは良いことで、MkLinuxのファイルも、CD-ROMファイルの内容をFTPでダウンロードするという暴力的操作ができるようになった。この際には、NetFinderという、FTPによるファイル転送をあたかもMacのファイルコピーのように行えるソフトの世話になった。

しかし、ネットワーク環境は、全体としては良くない。僕が出発するところは、東京ではSRAの四谷オフィスと自宅、博多では九大病院とシステム情報技術研究所とホテルである。ホテルを除く4カ所で専用回線を通してインターネットに接続できるのだが、いずれもファイアウォールの内側にいて、データ交換が自由にできない。結局、あるデータがどうしても必要になったとき、電話回線を利用してPPP経由でTCP/IP接続し、ゲートウェイに入り、さらにそこから自分のワークステーションに入って、一旦FTPなどでファイルをゲートウェイに転送し、さらにそこから自分のMacにファイル転送することになる、という書いているだけでもイライラする手続きを踏まなければならない。

「隠すべき秘密がないことを隠している」としか思えないような過度な秘密保護を行っているのは、ネットワークは高速化し自由化が進んでいるのに、日本の社会と会社だけが閉鎖的であるためだろう。このような秘密主義は反動を招き、かえって秘密が露呈する危険性を増すことは、かつてシグマ本部で起こっていた事象を見れば明かであるのに...

「ファイアウォールの外にサーバーを起きたいのだが...」

「ビジネス上どのような利点があるのですか？」

「電話と同じようなものでしょう？」

「電話は役に立っています。」

4. とは言っても、今はまだペンティアム系CPUにしか対応してないものが多いが...

5. その後、生命保険の解約がばれて、家庭内紛争が持ち上がっている。生命保険会社は、顧客のプライバシーを無視しているので、気を付けた方がよい。

というようなネットワークに強いはずの会社内での不毛な会話に時間を費やしたくないものである。

4.0 DTP環境

Adobe Acrobat 3.0Jを買って、すべてのソフトの出力はAdobe PDFファイル形式にした。これによって、Mac・Windows・UNIX⁶を問わずに同じファイルを持ち歩けばよいし、今回も南山大学の全く異なるUNIX環境にMacを持ち込み、PSプリンターで資料を講義直前に印刷することができた。

また、今までセミナー資料はAldus Persuasion 3.0J、文書はNisusWriter⁷という不安定なソフトを使っていたが、FrameMaker5という快適なドキュメントプロセッサ上で作成できるようになった。この原稿や日記もFrameMaker5⁸で書いている。僕のホームページも、FrameMaker5で作成し、PDF化したものを増やす予定である⁹。

文書を読み書きするときには、Jamming 2.2 PPCというシェアウェアが手放せなくなっている。これは、電子ブックの辞書を総なめして検索してくれる。例えば、「まんしゅう」という単語を英和辞書と広辞苑と知恵蔵と昭和史データベースと仏教辞典で一網打尽に検索するといったことができる。おもしろすぎて、文章を書いている途中の脱線が多くなる副作用がある。もっともこの機能は、CATALOG Maker 1.0というフリーウェアで、電子ブックのCATALOGを一つにまとめ、辞書をひとつのDisk Volumeに纏めておかなければ使えない。僕は1GBのJazディスクに辞書類を入れて持ち歩いている¹⁰。

今までに作成したセミナー資料や本や文章は、CD-ROMに焼いて持ち歩いているので、どこでも加工できる。ISO形式なら、追加して焼くこともできるので、バックアップにも良い。

5.0 おわりに

プログラマの能力とデスクトップ環境の強力は特に相関関係はない。しかし、同じ能力を持ったプログラマならば、デスクトップ環境が強力なほど能力が発揮できることも間違いない。そういうわけで、僕は能力を磨く前に、デスクトップを磨く癖が付いてしまっている。

6. 宣伝ではUNIXでも同じように読めるはずだが、まだ日本語版が出ていないという情報もある。

7. NisusWriterも、本を書くのに使っていた68000版は結構安定していた。NisusWriterはもはやディスク上に存在していない。

8. 別にAdobeの関係者ではないが、同社の製品は信頼性が高いものが多いので、つい高くて買ってしまう。

9. つまり、この前買ったばかりのクラリスホームページはもう不要だし、もっと前に買ったPageMillやSiteMillも不要になったということ。

10. つまり、Jazディスクは東京と博多の両方に置いてある、という自慢である。ついでに言えば、100MBのJipディスクは3ドライブ持っている。