

# ユーザー・インターフェース・シミュレータとその開発ツール "RINCS"

黒 沢 隆

(日本アイ・ビー・エム 藤沢研究所)

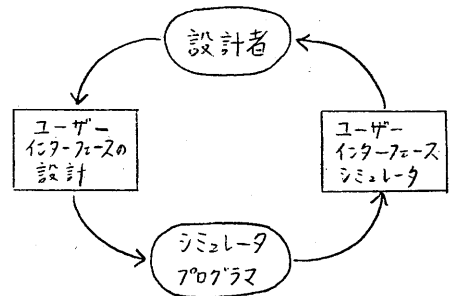
## 1. はじめに

今日、テクノロジーの急速な進歩により、コンピュータはより小型で安価なものとなり、特にオフィス環境への著しい進出を見せている。それに伴いコンピュータの主なユーザーも、もはやデータ・プロセッシングの専門家ではなくなりつつある。こうした状況においては、コンピュータ・システムの開発に携わる人たちは、システム内のマン・マシン・インターフェースの『ユーザビリティ』あるいは『使い勝手の良さ』というものを、より重要視していく必要がある。ユーザーにとって、あるシステムが魅力的かどうかは、機能の多様性よりは使い易さによって決まると言ってもよいであろう。ソフトウェアの設計者も、当然の事ながら、ユーザビリティの高い設計を心掛けなければならない。ユーザビリティはソフトウェアの『品質』を左右する重要な要素と考えられるであろう。

しかし、ソフトウェア開発の設計の段階で、ユーザビリティの評価をすることは、なかなか難しいことである。というのは、机上で考えられたマン・マシン・インターフェースは、実際に作って見た時にしばしば予想とは異なるものになってしまうことがあるからである。設計段階においてユーザビリティのより正確な評価を行なえるように、我々は英文ワード・プロセッサを開発する際に、ユーザー・インターフェース・シミュレータを作成した。そのシミュレータは、マイクロコンピュータをベースにしたもので、あるディスプレイ・ターミナルの画面上に、ワード・プロセッサの画面をシミュレートするよう作られていた。実際にシミュレートされたソフトウェアは、マン・マシン・インターフェース上特に重要と考えられた二つの機能：メニュー/プロンプト (Menu/Prompt) 機能及びオンライン・オリエンテーション・ダイアログ (On-line Orientation Dialogue) 機能であった。これらの機能については又章で説明したい。

我々がソフトウェアの設計において採用したアプローチを図示すると (図1) のようになる。

まず設計者が設計のガイドラインを作り、それをシミュレータ・プログラムが最初のシミュレータを作ってみる。これで設計者は自分の設計のユーザビリティをシミュレータ上でテストすることができる。そのテストの結果を基に、設計者は自分の最初の設計に変更を加え、プログラムもシミュレータを変更する。

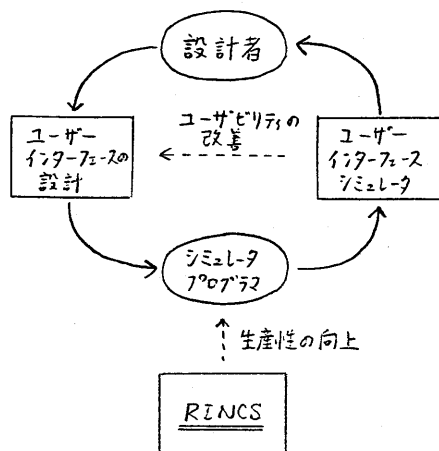


(図1) ユーザー・インターフェースの反復設計

この『反復設計』のサイクルが有効に働くためには、シミュレータ・プログラムの高い生産性が要求される。我々は、シミュ

レータ・プログラムが、設計者の要求に応じて、シミュレータを簡単に作成あるいは変更できるように、RINCS (Recursive Interpreter for Nested Character Strings) という小型のインタプリタをツールとして用いた。

本稿では、2章において、ワード・プロセッサのユーザー・インターフェース設計のユーザビリティを改善していくことに役立つシミュレータを紹介し、3章においては、シミュレータ・プログラマの生産性を向上させるために用いられたツール RINCS の概略について、そして4章において、結論として我々のアプローチの有効性について述べたいと思う。本稿においては、ユーザビリティの評価法を議論することは意図していないので、注意していただきたい。



(図2) 反復設計におけるシミュレータ及びRINCSの役割

## 2. ユーザー・インターフェース・シミュレータ

ここで、我々がシミュレーションを行なったワード・プロセッサの2つの機能について説明したい。

### メニュー/プロンプト機能:

まず、メニューとは、オペレータが現在行なっている仕事の段階で、どのような選択ができるかを、ワード・プロセッサの画面上にリストし、(例えば、テキストの作成、編集、プリントなど)オペレータに選択させるものである。またプロンプトとは、オペレータがメニューから選択したオペレーションを、どのように行なったらよいかを、次々にメッセージをディスプレイしながら、オペレータを導いていくものである。どちらの機能も、まだ訓練されていないオペレータにとって便利な機能である。

### オンライン・オリエンテーション・ダイアログ機能:

初めてワード・プロセッサを使うオペレータに、ワード・プロセッサの機能をアニメーション的に紹介することによって親しみを持たせ、またオペレータが実際の仕事をスムーズに始められるように、いくつかの使い方の例を示す機能である。

これらの機能は、人間工学的にみて重要な役割を果たす機能であり、非常に多くのレイアウトの画面(アニメーション的なものも含めて)がシミュレートされ、改良されていた。実際には、全部で500枚程の画面が設計され、シミュレートされた。シミュレーションを用いた反復設計の結果、メニュー/プロンプト用に87枚、オンライン・オリエンテーション用に72枚の画面が残された。

反復設計の途中で改良されていた主な点は;

- 画面のレイアウト
- メッセージの文章

- メニューから選択する方法
  - オペレータへのフィード・バック
  - スクロールリングの速さ
- などである。

### 3. シミュレータ開発ツール "RINCS"

#### 3-1. RINCSとは?

RINCSは、キャラクタ・ディスプレイへの出力データ量が、そのディスプレイからの入力データに比べて非常に多いようなアプリケーション用に開発されたインタプリタである。アプリケーションの例としては;

- ・ディスプレイを用いた教育コース
- ・アニメーション
- ・シミュレーション
- ・ゲーム

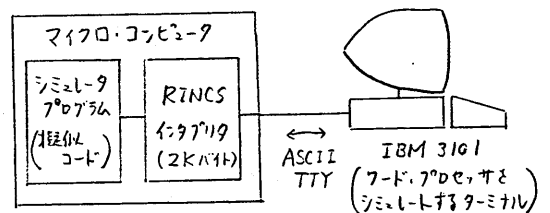
などが挙げられる。

RINCSの特長は次の2点である。:

- (1) プログラムはRINCSの命令を使って、望むレイアウトの画面(アニメーション的なものも含めて)を簡単にディスプレイできる。
- (2) RINCSのデータ構造の特性によって、蓄積すべき出力データ(画面データ)を圧縮できる。

RINCSを用いたプログラムの作成は次の様に行なわれる。まずプログラムは自分の望む画面をRINCSの命令(マクロ命令)を使ってソース・プログラムの形にコーディングする。RINCSアセンブラは、そのソース・プログラムから(インタプリタへの入力としての)擬似コードを出力する。この擬似コードがRINCSインタプリタにより解釈実行され、ディスプレイとの入出力(主に出力)が行なわれるのである。この擬似コードの意味については、3-2で説明したい。

現在、藤沢研究所においてRINCSは、IBM3101ディスプレイ・ターミナルを接続したマイクロ・コンピュータ・システム上で稼働し、今回のシミュレーションにもこのシステムが使われた。(図3)参照)



(図3) シミュレータ・システム

#### 3-2. データ構造と擬似コード

RINCSのデータ構造について説明する前に、RINCS設計の際に(3-1)に挙げたアプリケーションを考えて)定めた三つの条件について言及しておきたい。:

- ① 速く簡単にプログラムが作れること。
- ② 出力データを圧縮できること。
- ③ システム全体が簡単に開発できること。

RINCSのデータ構造は(図4)で示す様なNEST(入れ子)構造をしている。このデータ構造は共通なデータを共有することで出力データを圧縮し、またスタックさえあればデータの出力は容易に行なえるので、条件②及び③を満た

すものである。

ところで、一般に、或るデータ構造にはそれを解釈する為の(広義の)インタプリタが存在すると考えることができるであろう。RINCSのデータ構造にも出力用のインタプリタが必要であり、すでに(図4)に示される様に二つのコントロール・データが含まれている。RINCSの目的とするアプリケーションはディスプレイへのデータの出力が大部分なのであるから、コントロール・データの種類を増やし、一種のディスプレイ出力用の仮想的なプロセッサとしたものが、RINCSインタプリタである。

次に擬似コードについてであるが、RINCSインタプリタは、とり出した1バイトの擬似コードのMSB(Most Significant Bit)によって次の様に動作する。即ち、もしMSBが0ならば、それを7ビットASCII文字としてディスプレイに出力し、1であれば何らかの機能を実行するのである。

また条件①と③を満たすために、RINCSアセンブラがVM/CMS上のクロス・アセンブラとして、既存のマクロ・アセンブラのマクロ機能を利用して作成された。

### 3-3. RINCS命令セット

RINCSの命令セットは主に、スクリーンの場所指定などのディスプレイ・コントロール用の命令、及びサブルーチン・コール、リターン(図4)中の☒、☑に対応)などのフロー・コントロール用の命令から成り、プログラマが自分の作りたい画面を速く簡単に表現できるように工夫されている。(ここでは詳細な説明は省略する。)

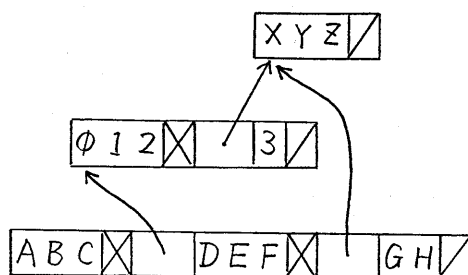
## 4. 結論

### 4-1. ユーザー・インターフェース・シミュレータの有効性

机上で設計されたユーザー・インターフェースは、それが果たして本当に高いユーザビリティを持っているかどうかテストする必要がある。我々の、シミュレータを使った反復設計というアプローチは、非常に有効であった。その理由は次の2点である。:

- 設計者はユーザビリティのみならず、設計の一貫性もチェックすることができた。
- 設計者以外の人々もシミュレータを使うことにより、明確にその設計を評価することができた。

ということである。



(図4) RINCSデータ構造

(このデータは 'ABC 012 XYZ  
3 DEF XYZ GHI' を表わす。  
尚、☒と☑はコントロール・データ  
で、スタックを備えた計算機の、  
それぞれサブルーチン・コール、リ  
ターン命令に似たものである。)

## 4-2. RINCSの生産性

シミュレータの設計者への有効性も、RINCSという手助けがなければ得られなかったであろう。他のツールとの比較ができないのは残念であるが、RINCSの生産性を示すデータを挙げておきたい。設計者から要求されたレイアウトの画面一枚をコーディングするのに要する平均時間(作成された全画面数約500枚での平均)は次の通りである。:

- a. 動きのない画面(文字のみ) ..... 約 5分
- b. 動きのない画面(文字と絵を含む) ..... 約 10分
- c. 動きのある(アニメーション的な)画面 ..... 約 15分
- d. キーボードからの入力待ちをする画面 ..... 約 20分

もちろん、このデータはプログラマのRINCS使用経験に依存するであろうが、通常のプログラミングに必要な命令数は少なく(約10)、しかもそれらは覚えやすいものである。初心者の場合でも大差はないようである。また、シミュレータ作成に要した時間全体のうち、コーディング及びデバッグに必要とした時間はそれぞれ約10%であった。今後は、より一般的なシミュレータ開発用のツールとして、RINCSをさらに改良していきたいと思っている。

## 5. 参考文献

1. Clark, I.A. "Software simulation as a tool for usable product design"  
IBM Systems Journal, Vol. 20 No.3, 1981 pp. 272-293
2. Helander, G.A. "Improving System Usability for Business Professionals"  
IBM Systems Journal, Vol. 20 No.3, 1981 pp. 294-305
3. Bethke, F.J. "Improving the Usability of Programming Publications"  
IBM Systems Journal, Vol. 20 No.3, 1981 pp. 306-320
4. Shackel, B. "Man-Computer Communication"  
Infotech State of the Art Report, Vol. 1, 1979  
Maidenhead, UK, Infotech International Ltd.
5. Martin, J. "Design of Man-Computer Dialogues"  
Prentice-Hall, Inc., Englewood Cliffs, NJ, 1973
6. Kasuga, H. "An Approach for Designing A Friendly and Flexible User  
Interface for An Office Workstation"  
Conference Proceedings, 1982 International Zurich Seminar on Digital  
Communications
7. McIntire, T.C. "Software Interpreters for Microcomputers"  
John Wiley & Sons, Inc., 1978
8. Gaines, B.R. & Foley, P.V. "Some experience in interactive system development  
and application"  
Proc. IEEE, Vol. 63.6, 1975
9. Johnson, E.M. & Huber, G.P. "The technology of utility assessment"  
IEEE Trans. Syst. Man & Cyb. SMC7-5, 311-325, 1977
10. Meister, D. & Rabideau, G.F. "Human Factors Evaluation in System Development"  
New York, Wiley, 1975
11. Rouse, W.B. "Design of man-computer interfaces for online interactive systems"  
Proc. IEEE, 63.6, 847-857, 1975