

この資料は日本エム・テクノロジー学会員専用です。
この資料を学会員以外がコピーしたり、学会員以外に配布することを禁じます。

Copy right : M Technology Association - Japan

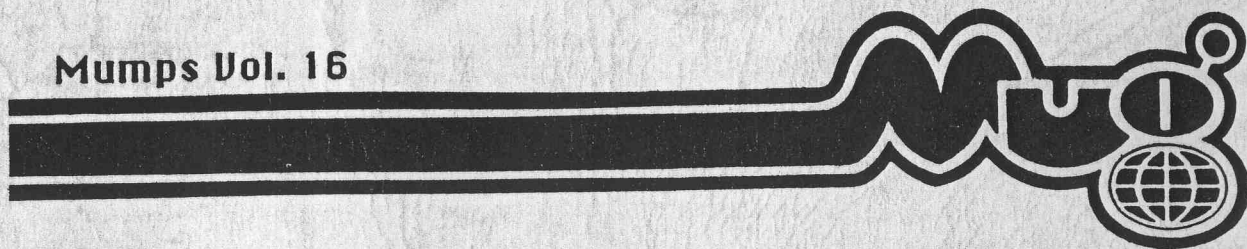
日本エム・テクノロジー学会事務局
〒259-1193 神奈川県伊勢原市望星台
東海大学医学部・基礎医学系
大櫛陽一

Tel: 0463-93-1121 ext. 2140

Fax: 0463-96-4301

Email: youichi@keyaki.cc.u-tokai.ac.jp

Mumps Vol. 16



**Proceedings of the 16th
Mumps Users' Group of Japan
Meeting**



November 22-24, 1989
Kyoto

平成2年12月28日(金)

第16回日本MUMPS学会大会 参加者 各位

日本MUG会員 各位

第16回日本MUMPS学会大会事務局

〒606京都市左京区聖護院川原町54

京大病院 医療情報部内

TEL: 075-751-3211

拝啓、年の瀬もいよいよ近づきつつある今日、会員の皆様には益々
ご清栄のこととお慶び申し上げます。

さて、この度「Proceedings of the 16th Mumps Users' Group of
Japan Meeting」が出来上りました。ここに同封致しますのでどうぞ
ご一読下さいますよう、お願い申し上げます。

末筆ながら、皆様方のご健勝とご活躍をお祈り申し上げます。

良いお年をお迎え下さい。

敬具

目次

Japanese CCSM-2 for the NCR Tower — Features and Capabilities	1
David B. Brown、Futoshi R. Imanaka、Sumiati Aliwarga	
既存アプリケーション・プログラムへの知識ベース・システムの組み込み方式	3
今泉 幸雄、小形 稔	
ニューラル・ネットワーク学習モデル — 誤差逆伝播学習 —	9
今泉 幸雄	
Non-MUMPSとの結合	17
大槇 陽一、田久 浩志	
電子教科書の日本語入力に関する考察	26
岡田 好一、遠藤 晃、高橋 隆	
病理組織標本ラベル発行	30
加藤 寿夫、三宅 和夫、杉崎 登、長田 裕次、馬場 謙介	
学会報告：北米MUG第18回大会	37
小森 優	
反復切断補正法による血液検査項目の臨床参考値の設定の試み	40
志賀 修一、小柳 逸子、江見 安一、神奈木 玲児、小森 優	
大量漢字グローバルのバッチ転送	42
田久 浩志、大槇 陽一、馬場 謙介	
呼吸機能線画データベースの構築について	48
寺村 昌文、古林 榮次郎、野口 弘、大杉 隆史	
仕様書自動作成ツールの開発	52
野口 弘、古林 榮次郎、寺村 昌文	
機械式カード検索機からパソコンマンプスへ	56
馬場 謙介、杉崎 登、三宅 和夫、鈴木 孝始、加藤 寿夫、石名田 洋一	
小型文字をスクリーンに書くスモールコンセプト	68
馬場 謙介、木村 一元、杉崎 登	
地域がん登録におけるMUMPSの応用	74
林 恭平	
患者さん自身が外来受け付けで操作するパソコン外来予約	80
林寺 忠、西小森 隆太、廣田 陽代、西角 淳、東 洋子	

ワークベンチにおけるプログラム作成理論	86	藤江 昭
看護勤務表作成支援システムの開発	102	本多 正幸、岩尾 幸子、大塚 清子、鶴岡 藤子、里村 洋一
病院に置ける内線ファクスの効用	106	三宅 和夫、馬場 謙介、杉崎 登、小玉 隆一
MUMPSの国際化への展望	113	若井 一郎
電子カルテの開発	121	山下 芳範、遠藤 晃、河野 清司、笹山 哲、岡田 好一、桜井 恒太郎、高橋 隆
DASLの試用	123	山下 芳範、遠藤 晃、高橋 隆
ユーザー側からみたアクセルワークベンチ	125	山本 和子、須藤 正克
MUMPS/4GL 日本語DASL	131	日本DEC
VAX/DSMとDECWindowsの結合について	135	太井 寿幸

Japanese CCSM-2 for the NCR Tower Features and Capabilities

David B. Brown
Futoshi R. Imanaka
Sumiati Aliwarga

MGlobal Singapore Pte. Ltd.

Abstract: Features and Capabilities of Japanese CCSM-2/NCR is the version of CCSM designed specifically for the NCR Tower. All the requirement of the US ANSI standard are met in addition to MDCC-Japan's specifications for Japanese extensions. Special implementation features such as multiple buffer pools for different priority users insure extremely fast interactive support. Extended debugging and real-time features are also described.

Keywords: CCSM-2/NCR MUMPS implementation
Japanese Extensions Image Processing
Real-time trace/monitor/edit/journaling Multiple buffer pools

CCSM-2/NCR is a standard MUMPS language implementation for the NCR/Tower 32 series computers. All the requirements of the US ANSI standard are met including the additions specified by the MDCC-J for Japanese Extensions. This version of CCSM runs on all the NCR/Tower 32 processors that support the UNIX System-V operating system.

As CCSM-2/NCR is a direct descendant of the CCSM-2 for the PC, all programs written on the PC version will run without modification on CCSM-2/NCR. In addition to this compatibility, CCSM-2/NCR also offers several new debugging and journaling features whilst improving on the run-time execution speed of its predecessors by maintaining separate buffers pools for each user.

This implementation is also language independent. In other words, only one version will be released for both the Japanese and English languages. There is, however, no need to configure CCSM-2 as it automatically recognizes and differentiates between Kanji and ASCII characters. This version is also capable of storing all the ASCII symbols, including graphic and European characters together with Kanji. At MGlobal, we have developed an algorithm that allows us to store ASCII characters as 1-byte and Kanji as 2-bytes whilst still ensuring proper collating (or sorting) sequence. CCSM-2 also supports the MDCC's proposal for large objects. This large record length is especially suited to image processing.

203 Henderson Road Wing A, #04-03 Henderson Industrial Park, Singapore 0315

The powerful new debugging facilities offered in this version of CCSM-2 provide a programmer with the ability to trace, monitor and even alter the execution of a routine. Break points may be preset or generated on certain condition changes. Local or global variable contents may also be modified during run-time. Finally, the debugging information may be re-directed to a file or printer, thus preserving the screen output. All these new debugging functions are standard for CCSM-2/NCR.

On almost all computer systems file data is threatened by hardware or power failures. CCSM-2/NCR provides 2 forms of journaling that may be employed either separately, or in conjunction with each other to improve storage integrity. This journaling is in addition to other system provided backup mechanisms, such as a UPS or disk mirroring.

In the first type of journaling, all disk blocks that are about to be modified are saved in a journal file. These disk blocks will remain in the journal until a flush or shutdown is performed. If a power failure occurs before a flush and a global file is damaged, the journal may be used to restore the global file to its last "flushed" state. This journaling is automatic and transparent to both the programmer and user. It may however be suppressed, for efficiency's sake, through CCSM's configuration file.

In the second form of journaling, all changes to data are backed-up in real-time to a second system via a communications link. This implies that the backup system does not need to be located geographically near the journaled system. Although this method of journaling requires an additional system, it provides the highest level of security for up-to-the-minute changes.

CCSM-2/NCR implements an efficient system of buffer pooling for routines and globals thus increasing execution speed. Generally, there are 2 sets of buffers, namely, a System-Partition Area and a User-Partition Area. The System-Partition Area is common to all users and contains information like shared programs, disk blocks and other common data. The User-Partition Area holds mostly pointers and data that is changed frequently by the particular user.

CCSM-2/NCR also maintains and stores integer values of all variables to reduce the conversion overhead when executing program loops. This optimization may be extended further to include a 2-digit fraction for accounting-related data. These optimizations ensure users a fast interactive response.

On a closing note, even though CCSM-2/NCR includes all these new features, it is still as simple to install and use as the CCSM-2 for the PC. This makes it the ideal choice for all MUMPS applications whether on the PC or on the NCR/UNIX platforms.

既存アプリケーション・プログラムへの
知識ベース・システムの組み込み方式

An embeded method of the knowledge base system
for the existed application program.

今泉 幸雄 、 小形 稔

アップジョン・ファーマシューティカルズ・リミテッド
筑波総合研究所 管理部研究データ管理課 開発化学部

Y.Imaizumi , M.Ogata

Upjohn Pharmaceuticals Limited Tsukuba Research Laboratories
Lab.Adminstration Lab.Data Control Control Development

データ・ベースを経由して、知識ベース・システムの世界にデータを取り込む考えは、数年前より試みられているが、ここでは既存のアプリケーション・プログラムの世界に、知識ベース・システムを組み込む方式を報告する。相互に共通なるインターフェイスを設定して、既存アプリケーション・プログラムより知識ベース・システムの推論部を起動する。インターフェイスの入力としては、推論に必要なデータ項目、データ値とフレーム名、出力としては、推論結果である。

知識ベース・システム、アプリケーション・プログラム、インターフェイス、フレーム名、推論、安全性試験システム、エキスパート・システム

The method of getting the data into the knowledge base system via a data base has been tried for these years. We report method of embedding the knowledge base system for the existed application program.

The common interface accesses to the inference parts of the knowledge base system from the existed application program and fire out it. The input items to interface program are data item , data value and frame name of the interface of knowledge base system and output items from it are results of interface.

knowledge base system , applicatiopn programs , interface ,
frame , inference , stability test system , expert system

〒300-42 つくば市和台23番地

23 WADAI , TSUKUBA - SHI , IBARAKI - KEN , JAPAN 300 - 42

1、はじめに

知識ベース・システム（以下KBS）[1]とデータベース（以下DB）、既存アプリケーション・プログラム（以下APP）の関連については次の様な歴史がある。第一世代 DBの後に実用化されたKBSは、DBと完全分離、第二世代 KBSへDBがデータ変換より参照可能への融合性と歩んできたが、まだKBS主導型システムであり、次に期待されるのはAPP主導型である。演者らは別々に開発されたAPPとKBS相互にインターフェイスを設定して、APPの世界にKBSを組み込んだ方式を開発したことを報告する。

2、知識ベース・システム、データ・ベースと既存アプリケーションの関係

2.1 DB, APP, KBSの関係

DB, APPとKBSの関係は知識ベースの発展と共に変化し、特にAPPの中にKBSへの役割も大きくなってきている。

表1. 知識ベース・システム、データ・ベースと既存のアプリケーションの関係

	世代の定義	DBを利用する 時間差 *1	既存アプリケーション プログラムの変更	備考
第一世代	KBSと DBの分離	時間差あり	なし	
第二世代	KBSと DBの融合	同時の 可能性あり	少しあり ^{*2}	*2 DBがAPPとKBSの どの位置によるかに 依存する
次世代	APPにKBS を組み込む	同時	追加あり ^{*3}	*3 APPとKBSに 共通インターフェイスを 追加する

*1 トランザクションデータを主とする

2.2 組み込むことによる期待度

APPの修正（追加、変更、削除）を出来るだけ少なくして、大きな期待を得るには、変更を少なくし追加中心にするとよい。

(1) APPでメンテナンス処理が多いブロックを、KBSに置き換えることにより、知識ベース（以下KB）を修正するだけでDBに依存した色々なアルゴリズムが展開できる。

(2) APPに対して、起動すべきKBSを追加、呼び出しをすることにより、定常処理に加えて経験的判断をする処理を展開できる。

我々は後者の処理を実施した。

3、組み込み方式

3.1 組み込み設計方針

(1) APP及びKBSは共にDSM（VAX版MUMPS）で作成済みとする。

(2) APPにKBSを呼び出す部分は、経験的な判断を必要とする処理部分とする。

(3) ローカル変数は

APP側 : 従来の変数

インターフェイス（APP側とKBS側）: "SOCKET"

KBS側: ワーキング・メモリー（プロダクション・ルールの推論域）インスタンス（知識フレームの推論域）
推論処理に必要な変数

APP, KBS共にローカル変数はルーチン内でクローズされて、ルーチン間のは決められている。

(4) ソケット（SOCKET）には、推論に必要な起動定義フレームとデータ名、データ属性、データ値を入力として推論結果を出力とする。

(5) 起動定義フレーム、知識フレームとプロダクション・ルールは、各々のエディターを用いて作成済みとする。起動定義フレームは最初に起動する知識フレーム名とプロダクションの推論戦略などより構成されている。

3.2 組み込み後の処理

- (1)データを見てKBSを呼び出すかの判定をする。
 - (2)APP側がKBSを呼び出す直前、ソケット（共通インターフェイス）に起動定義フレーム名（KBSを起動する最初のフレーム名）、データ項目名、データ属性とデータ値を設定する。
 - (3)KBSの呼び出し後の制御はKBS側に移る。
 - (4)ソケットの内容と既に作成済のKB（フレームとプルダクション・ルール）を利用して推論を実行する。
 - (5)推論結果をソケットに設定する。
 - (6)KBSの処理を終えて、制御はAPP側に移る。
 - (7)APP側はソケットの内容を提示して、次の処理を実施する。
 - (8)従来の処理をする。
- 以上の部分を追加することに実施できる。

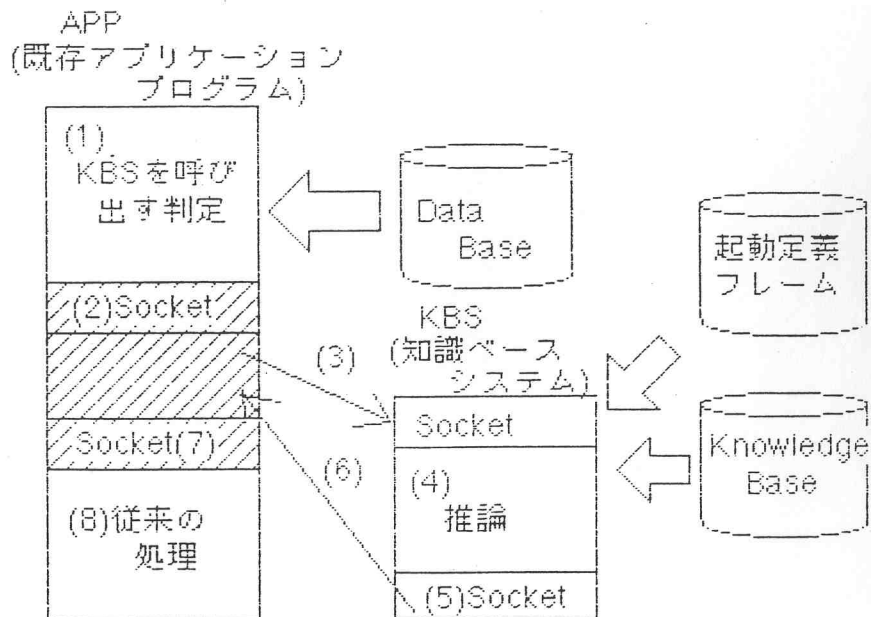



図2. 既存アプリケーション・プログラムの知識ベース・システムの組み込み概念

3.3 毒性試験システムへの組み込み

APPとしては毒性試験システム("JURAS", DSMで作成)、KBSは実験動物データ解析のエキスパート・システムのシェル版("BWV846", DSMで作成) [2], [3]を融合した。毒性試験システムは尿定性の個別表出力ジョブを対象として、ソケットの入力となる尿定性のデータ項目としては8項目(動物番号, PH, Protein, Glucose, Bilirubin, Keton, Occultblood, Urobilinogen)、データ値、データ属性、起動定義フレームである。(起動定義フレームより起動される)知識フレームは前記の8項目と推論に必要なデータ項目のスロットより構成される。8項目以外のスロット(データ項目)の値は、スロット推論のデーモン処理となる。

SOCKET = 8; OUDANBRD 

SOCKET(1) = 5002; X; AID[^]

SOCKET(2) = 7; 9; PH[^]

SOCKET(3) = 3; 9; PRO[^]

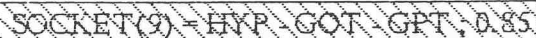
SOCKET(4) = -; 9; GLU[^]

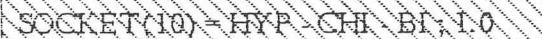
SOCKET(5) = 1; 9; BIL[^]

SOCKET(6) = 0; 9; KET[^]

SOCKET(7) = -; X; OCC[^]

SOCKET(8) = 0; 9; URO[^]

 SOCKET(9) = HYP - GOT - GPT; 0.85

 SOCKET(10) = HYP - CHI - BI; 1.0

 出力値

4. おわりに

APP, KBSともに同一言語かつMUMPS言語なので融合処理が容易であった。KBSが推論するにおいて必要となるデータ項目のデータ値はソケットより入れずにスロット推論のデーモン (IFNO_MAP-PED) 処理により対話形式にて実施しているが、今後はバッチ形式にての処理を検討する予定でいる。

[謝辞]

機会を与えてくださっている小林社長兼所長、富永氏、長谷川氏及び本研究に当たって多大なる助言を頂いた病理毒性研究統括部の三浦研究員に深く感謝いたします。

(参考文献)

- [1] Edward Hance Shortliffe :
"Computer - Based Medical Consultations : MYCIN"
American Elsevier Publishing Co. (1976)
- [2] 今泉幸雄、藤本浩司、小形稔、金子泰久、小池敏 :
"実験動物データ解析の支援システム (3)"
情報処理学会 第37回 (昭和63年後期) 全国大会論文集
pp1196 - 1197 (1988)
- [3] 今泉幸雄、金子泰久、小形稔、小池敏 :
"MUMPSによるデータ解析のエクスパート・システム"
PC world 11月号 pp68 - 74 (1988)

ニューラル・ネットワーク学習モデル
— 誤差逆伝播学習 —

A neural network learning model
— Back propagation error learning —

今泉 幸雄

アップジョン・ファーマシューティカルズ・リミテッド
筑波総合研究所 管理部研究データ管理課

Y.Imaizumi

Upjohn Pharmaceuticals Limited

Tsukuba Research Laboratories Lab. Admistration Lab. Data Control

脳はニューロンを基準単位とし、それらがシナプスにより密に結合して、ニューラル・ネットワークを形成している。ニューラル・ネットワークの応用分野に教師あり学習のパーセプトロンモデルがある。そのモデルの階層は3つより成立して、入力層—データ入力部、出力層—外界に出力されて、教師信号と比較される、隠れ層—入力層、出力層以外の中間層である。パターン認識モデルにおいて誤差逆伝播学習による、訓練パターンを用いる学習前と逆伝播とテストパターンを用いる学習後には多くの変化があることを報告する。

ニューロン、シナプス、ニューラル・ネットワーク、パーセプトロン、教師あり学習、誤差逆伝播学習、多層構造、パターン認識

A neuron is a basic unit of brain. A neural network consists of these neuron connected tightly each other by a synapse. A perceptron model of supervised learning is in a part of a neural network area. These are three layers model

Input layer : data input

Output layer : data output in outside. It is compared with supervised signal.

Hidden layer : except for input and output layer.

In this respect, I report on many progress between before learning of back propagation with training pattern and after learning with test pattern by back propagation error learning in pattern recognition.

neuron, synapse, neural network, perceptron, supervised learning, back propagation error learning, multilayer structure, pattern recognition

〒300-42 つくば市和台23番地

23 WADAI, TSUKUBA - SHI, IBARAKI - KEN, JAPAN 300 - 42

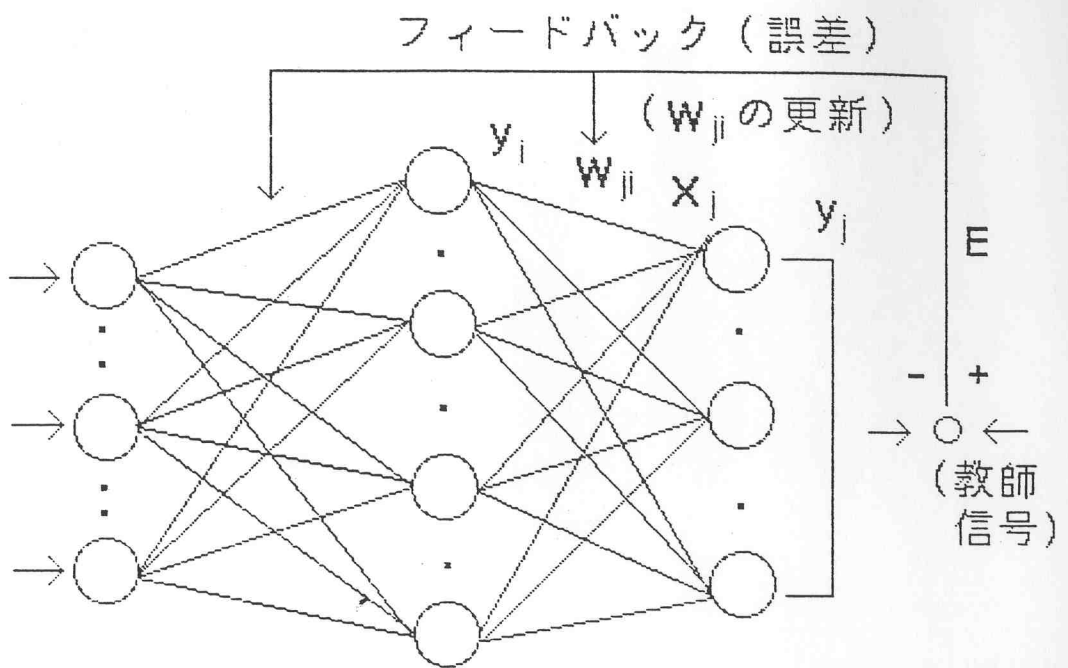
1. はじめに

人工知能をアプローチするためには、(1) 知能をどう捉えるか(2) 実現手段として何を考えるかである。(2) の実現手段としては2つあり、1つは今まで試みられている知能を記号で表現し形式的アルゴリズムとしての実現、他方は、知能は脳によって発現される機能であり、その本当の実現のために脳の神経回路網の結合とその重みづけのメカニズムを学ぶことである。後者の1例として誤差逆伝播学習プログラム([1],[2])をC言語よりMUMPSへの変換・追加機能と学習前、学習後の文字認識モデルについて報告する。

2. 誤差逆伝播学習

2.1 誤差逆伝播学習の概要

歴史的には甘利(1967,[3])によって提案され、Rumelhart, Hinton and Williams(1986,[4])によって認知情報処理モデルに応用されたのが誤差逆伝播学習アルゴリズム(back propagation error learning algorithm)である。入力層、出力層とその間にある隠れ層の階層型の3層モデルであり、理想出力値(教師信号)と実際の出力値(出力層)の誤差関数の値を局所的に最小にするように結合の重み W_{ij} を求めることによって、理論的に可能である。各層はユニットという基本単位の集合であり、1ユニットは多入力の1出力のいき値処理をして、伝播は、入力層→隠れ層→出力層の1方向である。



- 総力 : $X_j = \sum y_i W_{ji}$
- 入出力関係 : $y_j = 1 / (1 + e^{-x_j})$
- 全誤差 : $E = 1/2 \sum \sum (y_{j,c} - d_{j,c})^2$
- 学習(重みの更新) : $\Delta W = -\epsilon \delta E / \delta w$

図1. 誤差逆伝播学習

2.2 文字認識処理のアルゴリズム

入力データとして、文字データ（5×7の配列）を0と1の組み合わせの入力層として、隠れ層は16ユニット、出力層は4ユニットとして、学習をさせるために4種類の入力データと教師信号となる教師データ4つをグローバル・ファイルより読み込み、入力層、教師層の値とした。学習した後に結合の重み W_{ij} を利用してテストデータを実行すると、学習前と学習後の違いが現われる。



入力データ



テストデータ

- (1) 初期化処理
- (2) 入力データ（文字）、教師信号、テストデータ（文字）をグローバル・ファイルより読み込む
- (3) 結合の重み W_{ij} （荷重の初期化）
- (4) 伝播処理 （入力→隠れ→出力）
- (5) 伝播処理 （入力→隠れ→出力）
- (6) 逆伝播処理 （ W_{ij} 荷重の再計算）
- (7) テストデータの実行

学習前処理 (1)、(2)、(3)、(4)

学習処理 (5)、(6)
(エラー誤差が少なくなるまで(5)、(6)を繰り返す)

学習後処理 (7)

2.3 学習前と学習後

出力層と教師信号の2乗誤差の値がある以下 (= 終了誤差) になったときに学習をしたと認識させ、学習処理を終了させた。終了誤差が 0.08 以下の時は、約 15 回以下の逆伝播で収束し、文字認識処理での小さなあいまいデータでは有効であり、繰り返し処理が可能のためにWij荷重のファイル出力を試みた。

```

**** Before Learning ****
pattern  output1  output2  output3  output4
1 -> 0.475  0.368  0.575  0.395
2 -> 0.490  0.405  0.596  0.394
3 -> 0.464  0.385  0.542  0.418
4 -> 0.468  0.399  0.557  0.405
ErrorFunc : 1.885

Learn start time is Wed Sep 20 15:12:32 1989

**** Start to Learn ****
counter  pattern  output1  output2  output3  output4
1 1 -> 0.163  0.193  0.095  0.071
2 2 -> 0.176  0.282  0.102  0.075
3 3 -> 0.148  0.272  0.109  0.096
4 4 -> 0.141  0.259  0.101  0.087
ErrorFunc : 1.556
5 1 -> 0.079  0.109  0.065  0.058
6 2 -> 0.048  0.217  0.069  0.069
7 3 -> 0.061  0.140  0.085  0.085
8 4 -> 0.057  0.168  0.072  0.081
ErrorFunc : 1.62
9 1 -> 0.156  0.177  0.131  0.118
10 2 -> 0.072  0.429  0.135  0.156
11 3 -> 0.099  0.224  0.178  0.172
12 4 -> 0.082  0.287  0.143  0.178
ErrorFunc : 1.349
13 1 -> 0.452  0.178  0.281  0.213
14 2 -> 0.164  0.492  0.267  0.303
15 3 -> 0.228  0.220  0.383  0.307
16 4 -> 0.199  0.325  0.302  0.347
ErrorFunc : 1.031
17 1 -> 0.663  0.113  0.285  0.181
18 2 -> 0.154  0.671  0.247  0.301
19 3 -> 0.250  0.156  0.469  0.314
20 4 -> 0.199  0.259  0.329  0.385
ErrorFunc : 0.792
21 1 -> 0.701  0.081  0.209  0.124
22 2 -> 0.078  0.762  0.167  0.235
23 3 -> 0.131  0.115  0.520  0.288
24 4 -> 0.097  0.724  0.290  0.393
ErrorFunc : 0.578
25 1 -> 0.839  0.061  0.176  0.108
26 2 -> 0.078  0.813  0.128  0.219
27 3 -> 0.118  0.075  0.652  0.315
28 4 -> 0.091  0.166  0.297  0.499
ErrorFunc : 0.396
29 1 -> 0.895  0.056  0.128  0.091
30 2 -> 0.083  0.855  0.084  0.188
31 3 -> 0.111  0.056  0.736  0.301
32 4 -> 0.090  0.135  0.248  0.613
ErrorFunc : 0.261
33 1 -> 0.908  0.055  0.091  0.071
34 2 -> 0.078  0.882  0.055  0.145
35 3 -> 0.097  0.048  0.784  0.225
36 4 -> 0.082  0.114  0.187  0.694
ErrorFunc : 0.163
37 1 -> 0.907  0.053  0.077  0.060
38 2 -> 0.067  0.895  0.043  0.123
39 3 -> 0.080  0.043  0.831  0.166
40 4 -> 0.070  0.094  0.154  0.773
ErrorFunc : 0.103
41 1 -> 0.907  0.050  0.068  0.053
42 2 -> 0.059  0.904  0.036  0.109
43 3 -> 0.067  0.041  0.861  0.128
44 4 -> 0.061  0.081  0.130  0.829
ErrorFunc : 0.071
45 1 -> 0.904  0.056  0.062  0.054
46 2 -> 0.824  0.098  0.068  0.061
47 3 -> 0.131  0.855  0.035  0.072
48 4 -> 0.073  0.877  0.043  0.089
49 1 -> 0.060  0.054  0.773  0.187
50 2 -> 0.135  0.023  0.791  0.122
51 3 -> 0.296  0.075  0.060  0.476
52 4 -> 0.112  0.085  0.091  0.750
ErrorFunc : 0.071

after Learn start time is Wed Sep 20 15:15:17 1989

**** After Learning ****
pattern  output1  output2  output3  output4
1 -> 0.907  0.050  0.068  0.053
2 -> 0.059  0.904  0.036  0.109
3 -> 0.067  0.041  0.861  0.128
4 -> 0.061  0.081  0.130  0.829
5 -> 0.904  0.056  0.062  0.054
6 -> 0.824  0.098  0.068  0.061
7 -> 0.131  0.855  0.035  0.072
8 -> 0.073  0.877  0.043  0.089
9 -> 0.060  0.054  0.773  0.187
10 -> 0.135  0.023  0.791  0.122
11 -> 0.296  0.075  0.060  0.476
12 -> 0.112  0.085  0.091  0.750

```

図2. 実行結果

3. C言語よりMUMPSへの移植

3.1 C言語とMUMPSの違い

移植の環境としてはC言語 (FM16β, 日本語MS/DOS V3.3, Microsoft C V3.0) にての動作を確認後MUMPS (VAX11/750, 日本語V-AX/VMS V4.6J, DSM 3.0J) への移植を実施した。移植をするに当たって、両言語の主な差違を調査した。

表3. MS-CとDSMの主な比較

		MS-C (V3.0)	DSM (V3.0J)
変数	属性	属性宣言あり(double, float, integer, longなど)	属性宣言なし、文字型を基本
	長さ	32文字	8文字までの認識可能
	文字タイプ	大文字,小文字別々である	大文字,小文字同一である
	宣言の必要性	実行前に属性宣言をする	特になし
	有効範囲	外部レベルと内部レベルとに分れる。実行制御の範囲の有効宣言が可能 (1つのサブルーチンで幾つにも階層宣言が可能)	外部レベル (全プログラム共通) に共通となる。(特定にする時はNEWコマンド)
	初期化	<ul style="list-style-type: none"> 外部レベルでの宣言変数はCompile時に初期化される。 Auto変数、register変数は、実行制御のブロック単位で初期化すること。省略すると不定になる。 	引用される時までには値が入っていればよい。
	グローバルとローカル (有効範囲と関係あり)	<ul style="list-style-type: none"> グローバル変数を宣言すると各関数 (サブルーチン) のどこでも利用可能。 ローカル変数はそのブロック単位で有効であり、抜けた後も保存される。 	どのプログラムにも有効である。(MS-Cで述べるグローバル変数扱いとなる)

		MS-C (V3.0)	DSM (V3.0J)
二次元配列	サイズ宣言 インデックス スパース	必要 0 ~ (n-1) 整数 無し	無し 使用時 (整数、小数、文字) 有り
サブルーチン	呼び名 データ渡し 起動方法	関数 (function) グローバル変数 ポインター変数 引数渡し 関数値 関数記述	(サブ)プログラム 特になし DOコマンド
ファイル	ファイル編成 入出力 その他	ユニークなファイル編成 File構造体により提供される (例: fopen, fclose, ...)	木構造ファイル (一般的にグローバル・ファイル) SET, \$ORDER VAX/VMSの3ファイル編成も入出力可能
式	式の評価	*, /, は+, - より優先高い	*, /, +, - 同等で左より評価

3.2 移植方法

C言語よりMUMPSへの移植において特に注意した点を記述する。

- (1) ローカル変数
異なる関数でローカル変数と使用している同一変数名は、ユニークになるように変数名を変更した。
- (2) 関数の引数
関数引き渡し引数名はプログラム全体の共通変数として用いた。
- (3) ファイル
Cの1レコードと該当するのをグローバル・ファイルの (BPDATA (n)) のノード名に合わせた。
- (4) 初期化
外部レベルで用いる配列は、全て初期化ルーチンにて初期値 (=ZERO) を設定しておいた。

4. おわりに

- (1) MUMPSとC言語との実行時の時間の比較はしなかったが、少しC言語の方が速く感じ、収束回数はほぼ同じであった。
- (2) 問題解決において入力データと、結果などのデータ間においてアルゴリズムが不明な問題において誤差逆伝播学習も1つの解決案と認識できた。
- (3) 今後は、今回作成した荷重(Wij)の出力化を利用して既存システムへの応用を計る予定である。

[謝辞]

多大なる助言を頂いた慶応義塾大学の安西研究室、及び機会を与えてくださっている小林社長、富永氏、長谷川氏に深く感謝いたします。

(参考文献)

- [1] Robert Jay Brown :
"C言語による神経細胞シミュレーション"
I/O, Vol. 12, No.8, 1987, pp151 - 158
- [2] 安西祐一郎 :
"10.ニューラルネットによる学習", 認識と学習 (岩波書店)
- [3] S. Amari :
"A theory of Adaptive Pattern Classifiers"
IEEE Trans. EC, 16, 3, pp299 - 307 (1967)
- [4] D. E. Rumelhart, G. E. Hinton and R. J. Williams :
"Learning Representations by Back - Propagating Errors"
Nature 323, pp533 - 536 (1986)

Non-MUMPSとの結合
Interface between MUMPS and non-MUMPS

大櫛陽一、田久浩志
Yoichi Ogushi, Hiroshi Takyu

東海大学医学部 医学情報部
Department of Medical Information
School of Medicine, Tokai University

要約

MUMPSと他システムとの橋渡しとしては、同一CPU内での場合とCPU間の場合とに分けられる。まず、同一CPUの場合では、専用のOSから汎用のOSで動いたことが重要である。これにより他言語との共存が可能となった。他言語とのインターフェイスとしては、シーケンシャルファイルを介する、\$ZC、\$ZIの3通りの方法がある。CPU間通信としてはRS232Cによる無手順以外に確立された方法がない。MUMPSからSQLによりRDBをアクセスする、他言語からDDPシンタックスを発生するなどの方法の確立が望まれる。通信路としては、当面イサネットが使われるケースが多いものと思われる。このようなコミュニケーションによりMUMPSと非MUMPSの相互の良さが生かされるであろう。

There are 2 cases where MUMPS communicates with non-MUMPS. One is the case where the communication is carried out in the same CPU and the other between 2 CPUs. In the former case it is important that MUMPS runs on a general OS. We can use MUMPS and non-MUMPS simultaneously and communicate with non-MUMPS by using sequential files, \$ZC and \$ZI. So far asynchronous method by RS232C is the only way to communicate in the latter case. It is desired that we are able to access RDB database from MUMPS by SQL and use DDP syntax in the non-MUMPS languages. Ethernet will be used for the hardware. The interface between MUMPS and non-MUMPS will extract attractive points of the both sides.

Keywords : non-MUMPS, OS, SDP, HFS, \$ZCALL, \$ZINTERRUPT, TCP/IP

1. はじめに

1969年、G. O. Banett氏により発表されたMUMPSは従来のコンピュータ言語に比較してユニークな点を多く持っていた。旧来の堅い言語(COBOL、FORTRANなど)では医療のデータの多様性と流動性に対応が困難であったのに対して、MUMPSはこれらの問題によく答えてくれた。特に、医療データの多次元性とスパース性に対してよく設計されている。また、旧来の言語によるシステム開発は、その作成と保守に時間がかかるため、改善要求に対する対応が遅れ、医療のニーズを十分に吸収したシステムになり難かった。MUMPSは言語およびデータベースの柔軟性に加えて、開発保守時間が短いため、1970年代に入って、MUMPSによる医療システムが数多く作られた。MUMPSの持つ特徴を今一度整理すると次のようになる。

〒259-11 神奈川県伊勢原市望星台
Bouseidai Isehara, Kanagawa, 259-11 Japan

- 1) 使いやすさと生産性の高さ
 - *少ないコーディング量
 - *取得が容易
 - *設計時間が短い (スキーマ定義不要、可変長、宣言文なし)
 - *開発時間が短い (TSS、インタープリタ)
 - *保守時間が短い (オンライン保守、データベースの構造追加時に再編成不要)
- 2) 適応性
 - *汎用言語
 - *広いCPUレンジ (パソコンから汎用機まで)
- 3) 使いやすいデータベース管理
 - *ノードレベルの深さは制限なし
 - *添え字付き変数の形でアクセスできる
 - *添え字に日本語を含む文字列が使える
- 4) 効率性の高さ
 - *高速なデータベース (B+Tree、ディスク・キャッシング)
 - *効率の良いディスク使用 (スパース、可変長、自動ガーベジコレクション)
- 5) 移行性
 - *標準言語 (ANSI X11.1-1984、日本語標準)

しかし、MUMPSはその誕生から20年を経ており、この間に旧来言語の機能強化、第4世代言語や人工知能言語の登場、リレーショナル・データベースの発表などがあり、MUMPSの新鮮さや輝きが少し後退しているように見える。これらの新しいシステムが持つ特徴は以下のようである。

- 1) SQLに代表されるように、既存のシステムとの融合性
- 2) 構造化プログラムなどの見やすさの改良
- 3) RDBのように、分かりやすさとスマートさ
- 4) データベースとプログラムの独立
- 5) ノンプログラム環境の充実 (画面定義、Queryシステムなど)
- 6) 新しい利用方向への対応 (知識、画像、音声、公開ネットワークなど)

年	メーカーなど	MUMPS名	OS名
1969年	Banett	MGH/MUMPS	専用OS
1970年	DEC	MUMPS-11, 15	専用OS
1980年	Walters	microMUMPS	CP/M
	TANDEM	T/MUMPS	Guardian
1981年	DEC	VAX/DSM	VMS
1984年	住友電工	U-MUMPS	UNIX
1985年	住友電工	SP-MUMPS	MS-DOS
1988年	IBM	MUMPS/VM	VM

表1 代表的なMUMPS用OS

MUMPSは自らの改良を続けると同時に、こうした新しい流れとの連携を行い、MUMPSの特徴を生かした使い方がされるようにすることも重要となっているものと思われる。

2. MUMPSと他システムとのインターフェイス

他システムとのインターフェイスは同一CPU内の場合とCPU間の場合がある。まず、同一CPUではMUMPSが専用OSから汎用OSの上で動くことから始まった。表1にMUMPSの動いた代表的なOSを年代順に示した。

汎用OSの元で動くことにより、Multi-languageが可能となり、各言語の特徴を生かした使い方ができるようになった。この時の他言語とのインターフェイスとしては、現在次の3つの方法が可能となっている。(残念ながら、これらは標準外であるため、すべてのMUMPSシステムに装備はされていない。また、使い方も異なっている。)

1) シーケンシャル・ファイルを紹介して

Sequential Disk Processor (SDP)、またはHost File Servor (HFS)と呼ばれている。MUMPSから他言語へ、他言語からMUMPSへデータの受け渡しを可能としている。大量のデータの受け渡しには便利であるが、データ出力後MUMPSを抜ける、またはMUMPSへ入り込む必要があるためオンラインでの利用は無理である。

```
(例) Open "TEST.DAT":NEWVERSION For I=1:1:10 Write "DATA"_I,! ; VAX/DSM
      Open 51:("TEST.DAT") Use 51 For I=1:1:10 Write "DATA"_I,! ; SP-MUMPS
```

2) \$ZCALL

MUMPSから直接OSのコマンドまたは他言語で書かれたプロセスの実行が可能である。実行後は自動的にMUMPSの世界に戻ってくる。これによりMUMPSで記述することが困難な処理をサブルーチンの他言語で書くこともできる。

```
(例) OSコマンドの実行
      Set X=$ZCALL("%COMMAND.COM /CDIR") ; SP-MUMPS
      Set X=$ZCall(%SPAWN,"DIR") ; VAX/DSM
      他言語サブルーチンの実行
      Set X=$ZCall(%SQRT,2) ; VAX/DSM
```

3) \$ZInterrupt

割り込みレベルの処理をMUMPSから直接起動できる。これにより細かいI/O制御や、特殊なI/O機器の使用をMUMPSから行える。

```
(例) グラフィックの例
      S A=$ZI(#43,"GLINE",X1,Y1,X2,Y2,Color,0) ; SP-MUMPS
      マウスの例
      S M=$ZI(#45,5),X=+$E(M,1,5),Y=+$E(M,6,10),C=$ZI(M,11,15) ; SP-MUMPS (寺村昌文)
```

CPU間通信としてはMUMPS間にはDistributed Data Processing (DDP)がある。これは分散データベースシステムとしては非常に使いやすい機構である。しかし、non-MUMPSとの道としてはRS232Cによる通信か、先に述べたSDPまたはHFSで一旦シーケンシャル・ファイルに落とし、KERMITなどの汎用ツールを使い転送するぐらいである。

3. 今後のnon-MUMPSとのインターフェイス

同一CPUでの他のシステムの利用ということではMUMPS言語へのSQL組み込みが急がれる。これができればMUMPSの高速性、コンパクトさとRDBのインデックスの自由さの相互の良さを使い分けたシステム作りが増えてくることと思われる。言語レベルでの装備ができるまでは、昨年本学会でP. M. Kuzumak氏が発表したようにMUMPSシステムからSQLコマンドをRDBシステムに送り、結果をメッセージとして受け取る方法が考えられる。彼の説によるとショートデータの場合はDDPの方が早い、SQLで扱えるような大量のセットデータのアクセスの場合には彼のシステムの方が早いとのことである。他システムからMUMPSへのアクセスとしては色々考えられるが、DDPシンタックスによる方法が最もMUMPSの特徴を生かした方法になるものと思われる。しかしこの場合、non-MUMPS側でDDPシンタックスを解釈する必要がある。また、各

社独自のDDPプロトコルから共通のプロトコルへ移行する必要がある。

東海大学医学部病院システムで計画している例を次に示す。ホストコンピュータはNECのACOS 630/20で、医事会計システムはこの上で動く。また、患者情報、病名、診療情報などのデータベースがホスト上に構築される。診療サイドと医事以外の部門業務用の端末は17台のUstationにつながる。診療のテンポラリーデータと部門業務用データはUstationに持たれる。このコンピュータ間はEthernet (NEC Branch 4680) で結ばれる。(図1) 通信プロトコルはTCP/IPである。Ustation間はDDPであるが、ACOSとUstation間は独自のプロトコルによる通信を行う。その時の通信ソフトウェア体系は図2のようになる。VISはオンラインデータベースシステム管理システム、MCS-IIは内臓型通信制御プロセッサを用いたメッセージ管理システム、TSSAPSはTCPコネクションおよびアプリケーションレベルのセッション制御である。メッセージフォーマットを図3に示す。標準のヘッダー54バイトの後にNEC独自のヘッダーおよびコントロールコード14バイトが続く。今回の通信の共通部が63バイト、各アプリ独自の部分が最大4,024バイトとなっている。1つのメッセージが1つのセグメントに収まらない場合は最大3つのセグメントを使って転送する。共通項目の内容については図4に、病名の問い合わせの例でのトランザクションデータの内容を図5にそれぞれ示す。MUMPSでのコーディングとしてはアプリ独自のヘッダー情報をローカル変数にセットし、リクエストジョブを起動するサブルーチンを実行し、特殊関数により自分のイベントフラグをウエイトする。ここを抜けると汎用受信グローバルに目的のデータがセットされているので、このグローバルを使い必要な処理をしていく。このシステムは90年1月から段階的に稼働を始めていく予定である。

4. おわりに

MUMPSは大変使いやすいシステムであり、それだけでも十分な場合が多い。しかし、コンピュータのハードウェアおよびソフトウェアの進歩は急激に進行しており、ユーザーニーズも多様化している。MUMPS自体もこれに適合するよう進化をすべきであるが、すべてのニーズをカバーすることは難しいであろう。MUMPSの基本的な特徴を生かしつつ、non-MUMPSの世界と共同して良いシステムを作り上げていく必要性はますます高まるものと思われる。最後に現在、東海大学医学部のシステム開発に協力いただいております。今回の資料を提供していただいたNEC医療情報システム部の永井肇、藤枝正秀両氏、住電システム応用システム部の伊藤貴章氏に感謝致します。

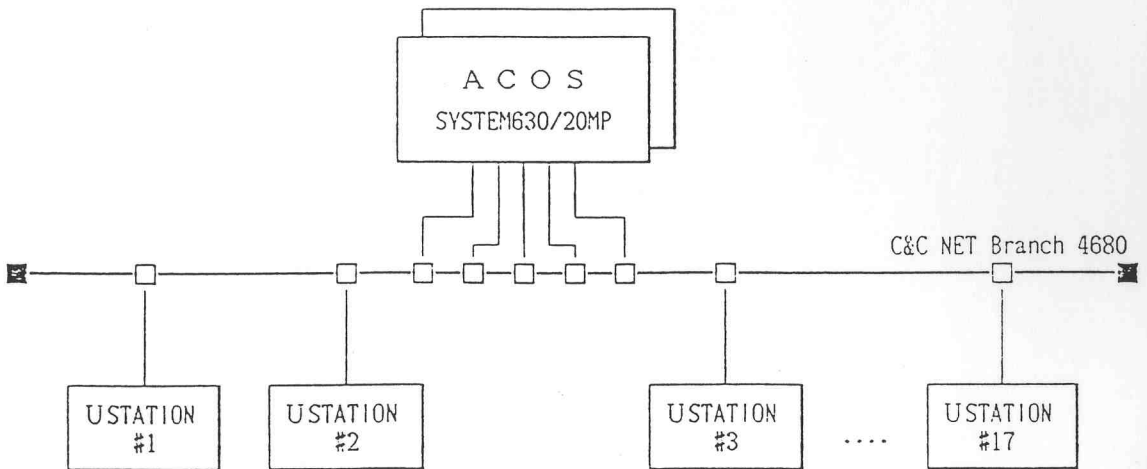


図1 ネットワーク・ハードウェア構成

ACOS-4/MVP XE

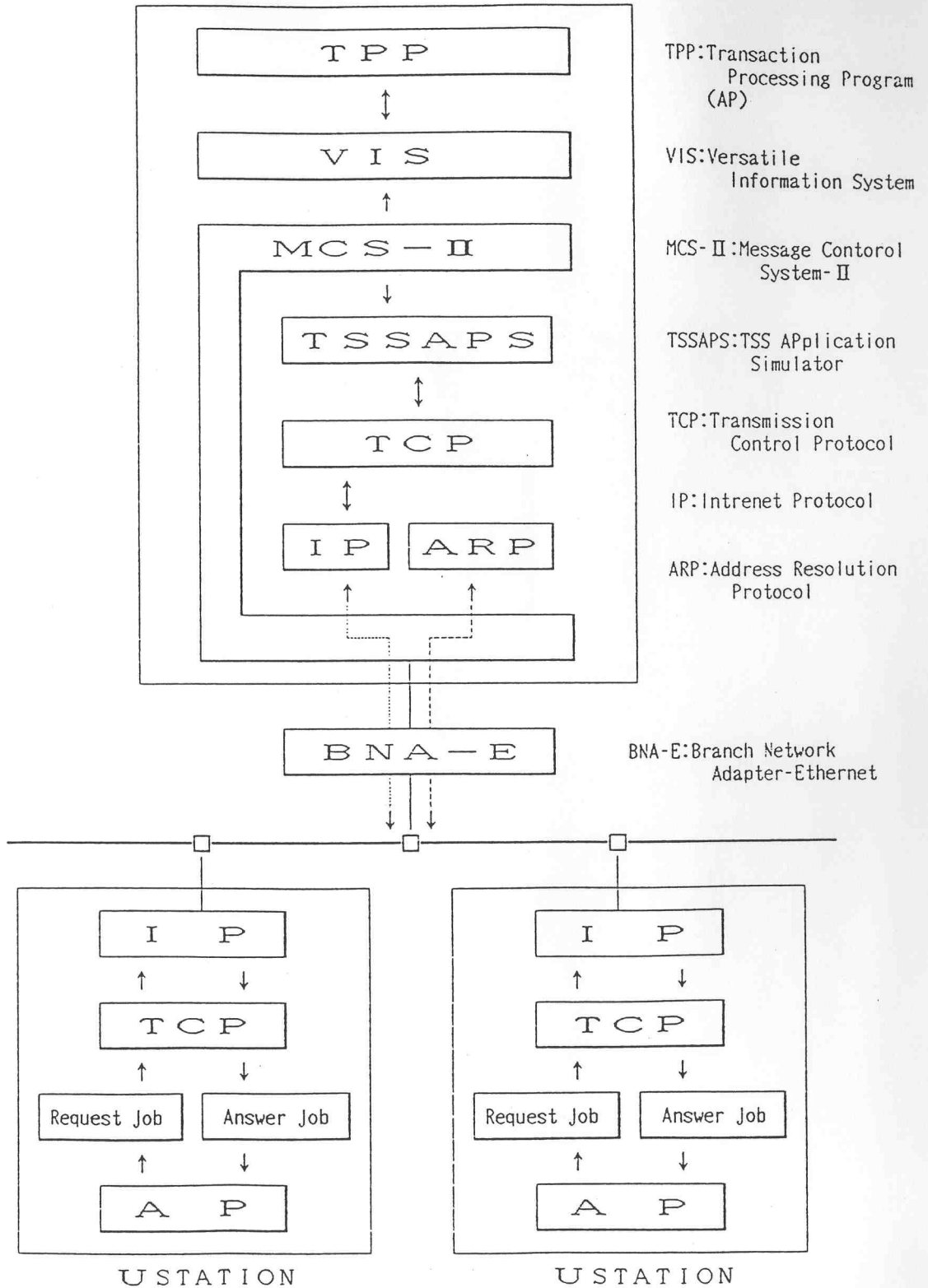
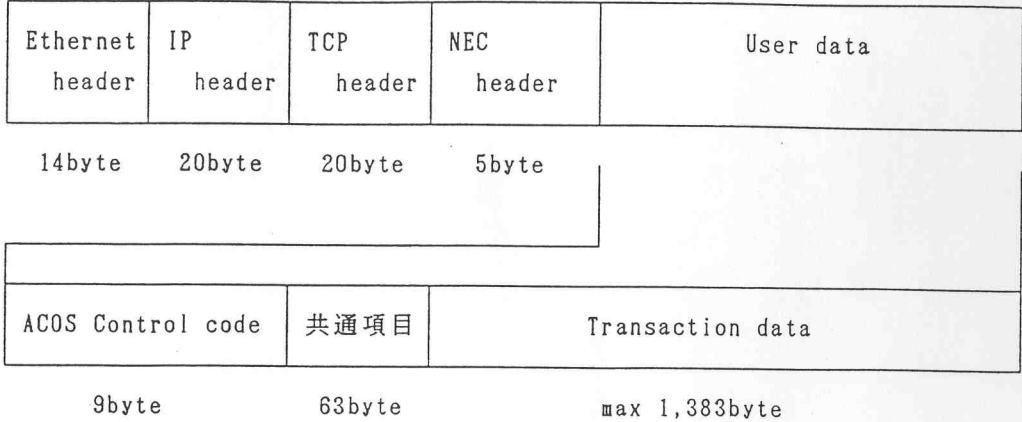


図2 ネットワーク・ソフトウェア構成

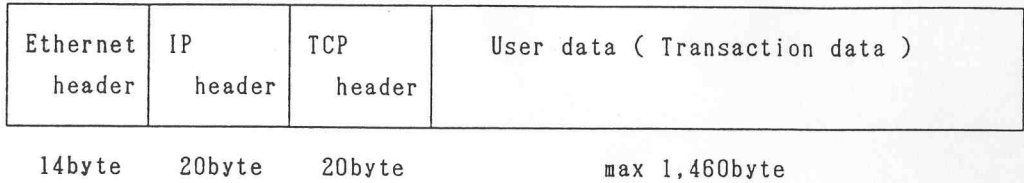
①第1セグメント

<----- 1 segment << max 1,514 byte >> ----->



②第2セグメント

<----- 1 segment << max 1,514 byte >> ----->



③第3セグメント

<----- 1 segment << max 1,235 byte >> ----->

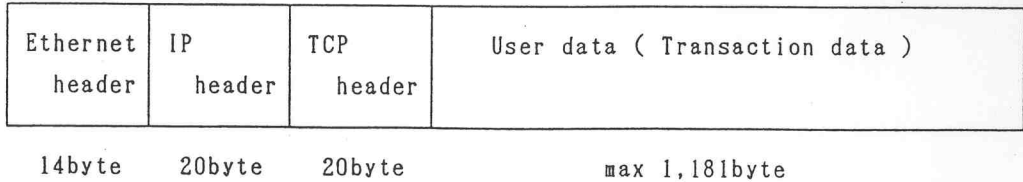
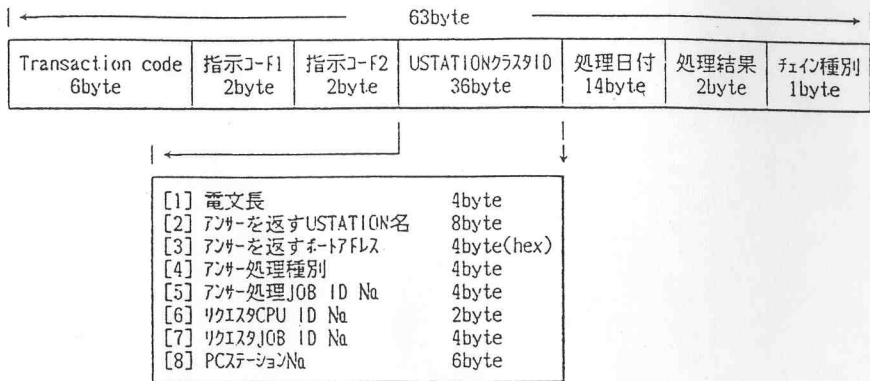


図3 通信メッセージフォーマット (全体)

各メッセージ全てについてこの共通項目を付加する。



《各項目の説明》

No	項目名	length	方向	内容	
1	Transaction code	6byte	US→AC	ACOS側処理プログラム名称	
2	指示コード-F1	2byte	US→AC	Transaction codeに対する細分類コード	
3	指示コード-F2	2byte	US→AC	"	
4	USTATION側クラスID	電文長	4byte	US↔AC	文字数(2byte系文字は1文字とカウント)
5		アンサーを返すUSTATION名	8byte	US→AC	ACOS側VDL記述中の端末名 ex. UST012
6		アンサーを返すポートアドレス (hex)	4byte	US→AC	原則としてUSTATION側受信ポートは'2002'
7		アンサー処理種別	4byte	US→AC	ACOSでは上りメッセージ中の下りメッセージに コードするのみ。
8		アンサー処理JOB ID No	4byte	US→AC	
9		リクエストCPU ID No	2byte	US→AC	
10		リクエストJOB ID No	4byte	US→AC	
11		PCステーションNo	6byte	US→AC	
12	処理日付	14byte	US↔AC	YYYY年MM月DD日hh時mm分ss秒, 年は西暦	
13	処理結果	2byte	US↔AC		
14	チェイン種別	1byte	US↔AC	メッセージ種別を表す(先頭・中間・最終)	
計		63byte			

US:USTATION
AC:ACOS

図4 通信メッセージフォーマット (共通項目)

伝文No:	USTATION クラスタID											処理結果	小冊番号			
ACOS-U-STATION	制御コ-F	トランザクション コ-F	指示 コ-F1	指示コ-F2	伝文長	送信先 宛先名	送信先 ポート	MUMPS 処理指示 コ-F	MUMPS JOB ID No	送信元 USTATION CPU ID	送信元 USTATION JOB ID	PC STATION No	処理日付	チェーン 種別	処理結果	小冊番号
(8BYTE)	X(09)	X(06)	XX	XX	9(04)	X(08)	X(04)	X(04)	X(04)	X(02)	X(04)	X(05)	X(14)	X	XX	X(8)

伝文No:	科コ-F	X(2)
-------	------	------

伝文No:	
-------	--

図5-1 通信メッセージフォーマット (トランザクションデータ)
----- 病名問い合わせの例 -----

伝文No: _____

USTATION クラスターID										処理日付	チャイニング種別	処理結果	検索件数	
制御コード	トランザクションコード	指示コード	指示コード	伝文長	送信先宛先名	送信先アドレス	MUMPS 処理指示コード	MUMPS JOB ID No	発信元 USTATION CPU ID	発信元 USTATION JOB ID	PC STATION No			
X(09)	X(06)	XX	XX	9(04)	X(08)	X(04)	X(04)	X(04)	X(02)	X(04)	X(06)	X(14)	X	9(4)
(1006BYTE)														

伝文No: _____

OCCUR B										科目コード	科名	開始日	終了日	転帰区分	入力非氏名	科目コード	科名	病者コード	付記	
X(2)	N(3)	X(8)	X	X(4)	X(4)	N(30)	9(6)	9(6)	9(6)	X	N(10)				X(2)	N(3)	X(8)	X(8)	X		
----- 病名報告の例 -----																					

伝文No: _____

17分組コード	300分組コード	病名	開始日	終了日	転帰区分	入力非氏名	未使用
X(4)	X(4)	N(30)	9(6)	9(6)	X	N(10)	X(84)

図5-2 通信メッセージフォーマット (トランザクションデータ)

電子教科書の日本語入力に関する考察

京都大学医学部付属病院医療情報部 岡田好一、遠藤晃、高橋隆

1、はじめに

京大病院では英文の内科電子教科書を作成し、病棟、外来の端末に提供している。電子教科書は、教科書を計算機に入れ、内容による検索システムを付けたものである。

内科電子教科書には作成当初から、日本語化の要望があった。入力部分は、外来でよく見られる158の症状を14の部位別に分け、選択で入力する方法を試作している。

この日本語入力部分はパソコンでの使用を前提としていたが、端末の体系が違う、IBM社の大型機の日本語MUMPSである日本語MUMPS/VMに、極めて容易に移行できた。

パソコンMUMPSからMUMPS/VMへの移行手順を報告し、あわせて、電子教科書の日本語入力について考察する。

2、日本語入力部分のMUMPS/VMへの移行

MUMPS/VMは日本語化されており、イ草案に準じた標準的な日本語の処理が可能である。しかしながら、日本ではASCII端末でなく、大型機固有の端末を使用するので、ここで問題が生じている。

内部的には問題ないが、端末の体系の違いがプログラミングに重大な影響を及ぼす。特に、カタカナや漢字を使った場合や、エスケープシーケンスで画面に工夫を凝らしたプログラムが影響を受ける。エスケープシーケンスの非互換は過去に問題となったが、ユーザーは端末の高度な機能を使う傾向があり、特に、新しく開発されたプログラムに著しく非互換部分が使われている。

大型機の端末をMUMPSに使用する場合に予想される問題点を表1に示す。

しかしながら、予想に反して、今回作成した電子教科書の日本語入力部分はほぼ手直しなしで、図1の手順により移行できた。

3、症状の日本語入力の問題点

内科電子教科書は、入力した症状で疾患を検索する。

表1 大型機の端末をMUMPSに使用する場合に予想される問題点

1、英小文字と半角カナが同時に利用できない

2、コード体系の相違

[→	φ
]	→	
	→	!

3、エスケープシーケンス

ANSIエスケープシーケンスの一部のみ使用できる

4、READ命令

エスケープシーケンスでカーソルを動かしてから入力すると、以前に表示した文字の入力が起こりうる

入力フィールドとプロンプトの間に、1個以上の空白が必要。無い場合は、自動で1個の空白が入力時に出力される。

R *X ができない

画面がスクロールしない。表示を消さないと、次に行かない

半角、全角文字が混在する入力ではSI/SOキャラクタが入る(表示では入らない)

症状に英語を使う場合は、症状を利用者に直接タイプさせている。これが可能なのは、英語の症状入力では、直接入力時にカナ漢字変換が不要であること、適当に考えた症状で適当に疾患が当たり、次のキーワードが考えやすいこと、が原因であろう。

直接に日本語入力すると、変換に手間取り、電子教科書の対話性が保てないとの判断で、選択入力による症状の入力部分を試作している。

選択入力では、症状の多さから、キーワードを分類して表示させるのが必須となるであろう。その際、一通りの分類では偏見が入り得る。

電子教科書は広い分野の検索を目指しているので、偏見はなるべく排除したい。対策として、何通りかの分類を取り揃えることが考えられる。

実際には、部位別、性状別など、2、3の分類しか考えられず、利用者を混乱させるには至らないと思われる。

選択入力では、選択ために1画面(1プログラム)必要となるが、その分、付加的な価値を盛り込むことができよう。例えば、類語、関連後、反対語、連想される語の提示が考えられる。

4、まとめ

パソコンで作成した電子教科書の症状の日本語入力部分は、大型機のMUMPSに容易に移行できた。

電子教科書の症状の日本語入力部分について考察した。

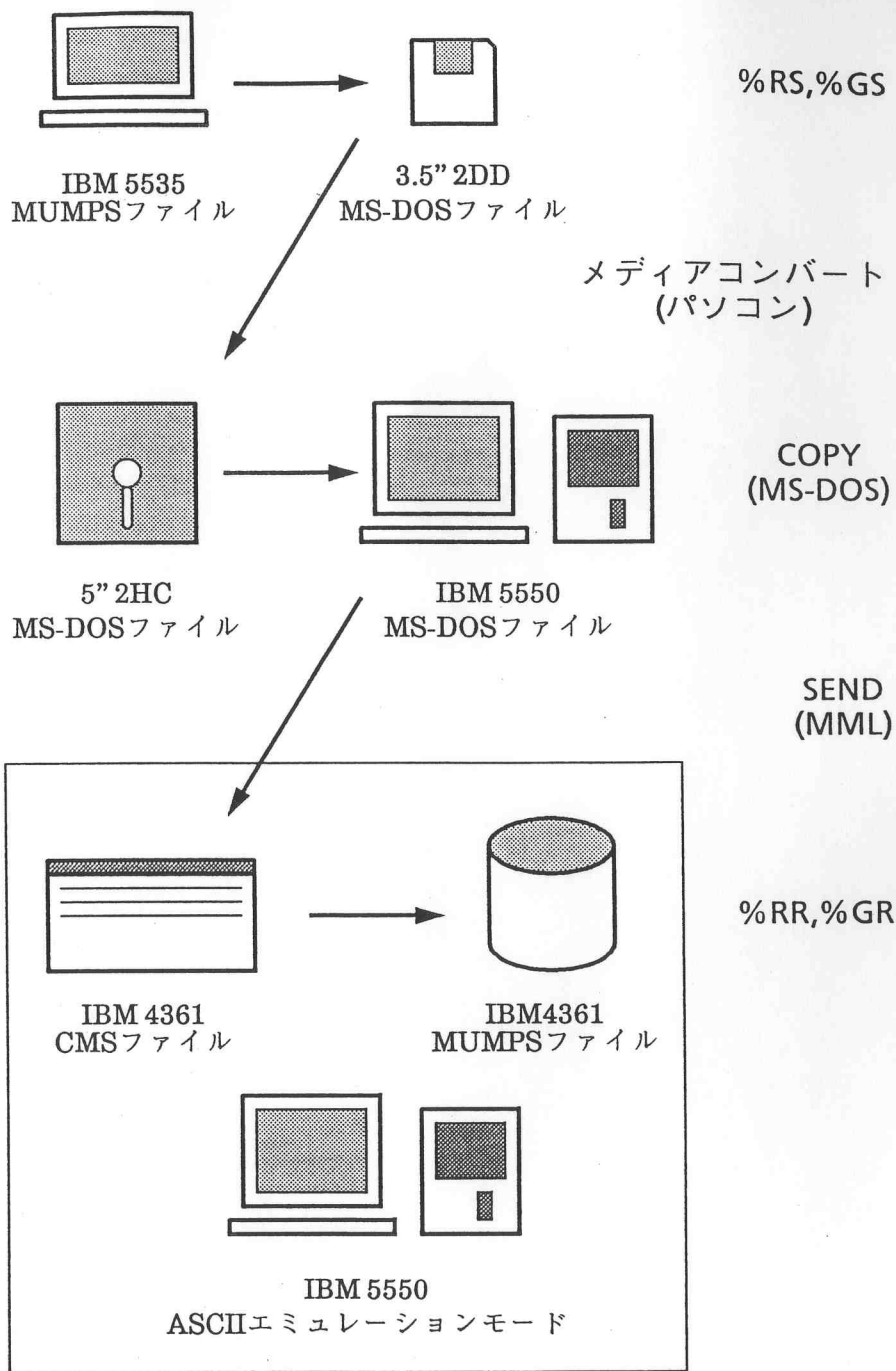


図1 内科電子教科書の日本語入力部分の移行作業
(パソコンから大型機へ)

病理組織標本ラベル発行

Histological Slide Label Issue System in Pathology Laboratory

加藤寿夫^{*}, 三宅和夫^{*}, 杉崎登^{*}, 長田裕次^{*}, 馬場謙介^{**}
Toshio Katou^{*}, Kazuo Miyake^{*}, Noboru Sugizaki^{*}, Yuuji Osada^{*}, Kensuke Baba^{**}

^{*}国立埼玉病院臨床検査科、^{**}同臨床研究部
^{*}Clinical Pathology Laboratory, National Hospital of Saitama
^{**}Department of Clinical Research, National Hospital of Saitama

ABSTRACT

A MUMPS application to issue labels of microscopic slide for histopathological examination and diagnostic cytology was composed by SP-MUMPS (Sumiden) within PC-9800 (NEC). In order to get nice input operation, Xecute[®] small concept which has been established by one of the authors was efficiently utilized in this application. In this application, strings on each label can be edited on the 2D printing image with pseudo-cursor operating via ten-key.

はじめに

最近、病理組織標本スライドのラベルを発行する専用機が、市販されようになったが、これを購入出来る恵まれた施設は少ない。今回、我々は、検査室に設置されていたパソコンに、ラベル発行の機能を持たせ、本病理システムのサブシステムとして、実際の業務に使用可能となったので報告する。

Toshio Katou, Laboratory of Clinical Pathology, National Hospital of Saitama, 2-1 Suwa, Wako, Saitama 351 Japan (TELEPHONE: 0484-62-1011 ext: 362, FAX: 0484-64-1138)

システム構成等： 使用したパソコンは、極く普通の物で、この目的のために何等の、部品の追加もしていない。使用したマンブスは、SP-MUMPS Version 2.2A(住友電工)である。ラベル発行システムは、受付台帳作成作業に始まる本院の病理システムの一サブシステムとして設計されている。

環境対応への配慮： システムが置かれている環境は、次の4つに大別出来る。

- (1) 親コンピュータがない
- (2) 親コンピュータから患者情報が得られない
- (3) フロッピーディスクを介して、デイリーに患者情報が得られる
- (4) 親コンピュータと電氣的に結合されている

我々のシステムは、上記のどの場合でも働く様になっている。本院の現在の環境は、上記(2) から上記(3) に移行しているところである(図2)。本院では、医事用のコンピュータ(親コンピュータ)は、単独で作動しているため、患者情報等を、電気信号として病理が直接受け取る事は出来ないが、医事課の好意でフロッピーディスクを介して患者固有情報(患者登録番号、氏名、生年月日、性別等)を病理室で受け取る事が可能となった。しかし、フロッピーへのコピー作業や、これを病理室へ運ぶ時間が間に合わない事もしばしばあり、上記(2) と (3) が混在した状態にある。

入力操作性の配慮： 年齢入力を例に、入力操作に対する配慮を説明する。年齢のプロンプトに対してする入力は、年齢の他に、和歴の生年月日、西歴の生年月日でも受け付ける。年齢、和歴、西歴の区別は、下記のように、その書式でマンブスが判断するようにした。

- 3桁以下の数字であれば、年齢
- M, T, S, H で始まる数字であれば、和歴
- 8桁の数字であれば、西歴
- 6桁の数字であれば、'略式'西歴

編集入力の利用： 随所に、Xecute ^%e スモールコンセプトコンセプト^{1,2)}を使って、「前入力をデフォルトし不都合ならカーソルキーで編集するようにする」等により入力作業の効率化を図っている。例えば、病理番号は、特に指定の無い限り、前回のラベル発行で作業が終了されている番号の次の番号が、自動的に表示され、不都合なら、カーソルを移動させ修正のできるようになっている。

印刷イメージでの再編集：（その日のラベル発行の）一連の入力作業が終了して印刷に移る前に、（その日の）ラベルの印刷イメージが表示される。その表示に対して、カーソルキー操作で、疑似カーソルが移動しあるラベルを選択するとそのラベルの内容が編集できる。

操作の概要

病理システムからの起動： ラベル発行システムは、標準的には、親システムである本院病理システムのメニュー画面から起動し、同画面で終了する。図1aは、親システムで[受付台帳記帳]にカーソルがある状態、図1bは、[初回ラベル発行]にカーソルがある状態を示している。

病理番号の入力： 病理番号をパソコンに与えると、リターンで先に進むかたちで、患者の氏名、年齢、性別、次々に表示される。（表示しれた患者の氏名、年齢、性別に万が一、誤りがあつたらここで修正できる）。

ラベル発行枚数： 症例についてのラベル発行枚数の入力の要求に答える。この際、前の症例の発行数を修正する形で入力及要求されますので例えば、バイオプシーの症例が続いて、暫く同一枚数を続けて入力したい場合など、操作が快適である。

連番付加： 病理番号の右に発行枚数までの連番を、印刷するかどうかをYかNで答える。この場合も、前の症例の、Yes なら Y、No なら N が、デフォルトになる。

ラベルに印刷する文字： 次に、(先に指定した)発行枚数分だけ、次々に、ラベルに印刷する文句を要求してくる。年齢と性は、いつも自動的に表示されるが、カーソルキーを操作して、また別の文句に変えることもできる。もし、次々にリターンキーを押した場合は、年齢と性だけが印刷される。年齢、性以外については、前のラベルのその行の内容がデフォルト表示され、不都合なら編集できる。図2に、一例の入力が完了したスクリーンの状態を示した。

入力ループとその停止： 以上の様に指定枚数分の入力が終了すると、自動的に、病理番号が次の番号に移り、以下同様な作業が繰り返す。病理番号の要求対して、ピリオドを返すと、入力ループは停止し、次の印刷イメージの表示に移る。

印刷イメージの表示： 入力された内容がラベル印刷時の書式に合わせて表示される(選択モード)(図3)。ここで、入力内容のチェックと修正がもう一度できるようになっている。青色疑似カーソル(図3aでは色が表現されていない)は、テンキー操作で修正したいラベル位置に移動させることができる。最上段または最下段を越すようなテンキー操作をすると、全体がスクロールする。

印刷イメージで編集： 上記の方法で、ラベルを選び、リターンキーを押すと、カーソルは赤に代り(図3bでは色が表現されていない)、そのラベルの印刷内容の編集が可能になる(編集モード)。また、この画面に於ても、新規ラベル発行の入力ができるようになっている。図4では、入力作業の第2行の第2桁に文字編集カーソルがある。

ラベルの印刷： 選択モードで、ピリオドを入力すると、図5のようなラベルを印刷する。印刷が終了すると、親システムのメニュー画面に戻る。尚、ラベルの最下段の「国立埼玉病院」の文字は、システムインストールのときに他の施設名に自由に変更できる。

ラベルステッカーについて： 図4のラベルは、特注のステッカーに印刷した状態を示している。このステッカーは、市販されていないが、照会することはできる。

むすび

従来、標本ラベル発行作業は、標本が出来上がって初めて、開始されていた。しかし、このシステムを使えば、検体受付時から、標本が出来上がるまでの手のあいた時に、必要な情報を入力できて、作業が合理化れ、以前より奇麗なラベルが得られるようになった。今日では、ほとんどの検査室に、大抵1台や2台のパソコンがり、このシステムは、その様な施設に容易に導入出来ると考えられる。普及を期待する。

文 献

- 1) Baba K: On small concept Mumps 14(suppl): 73-76, 1988
- 2) Baba K: Small concept. Mumps 14: 33-43, 1988.

受付台帳記帳	受付台帳記帳
初回ラベル発行	初回ラベル発行
初回ラベル印刷	初回ラベル印刷
特染時入力	特染時入力
特染ラベル発行	特染ラベル発行
特染作業表発行	特染作業表発行
報告内容記帳	報告内容記帳
各種リスト発行	各種リスト発行
各種検索	各種検索
環境設定	環境設定
終了	終了

病理番号: S-3033

患者名: リカチ キロ
AGE: 65Y
SEX: F

LABEL 発行枚数: 11 番号 <Y>:

(1)	66YF HE	1
(2)	66YF HE	3
(3)	66YF HE	5
(4)	66YF HE	6
(5)	66YF HE	8
(6)	66YF HE	10
(7)	66YF HE	12
(8)	66YF E/W	6
(9)	66YF PRS	6

(a) (b)

図1 親システムのメニュー

- (a) 選択カーソルが受付台帳記入にある状態
- (b) 選択カーソルが初回ラベル発行にある状態

図2 ラベル情報入力画面

説明は本文参照

S-30336 (4)

タカハシ リン
59YF PAS

埼玉病院病理

S-30336 (5)

タカハシ リン
59YF E/V

埼玉病院病理

図4: 特製シールに印刷した状態

学会報告：北米MUG第18回大会

Report on the North America MUG 18th Annual Meeting

小 森 優

Masaru Komori, Ph.D.

京都大学附属病院分子病診療学

Dept. of Clinical Molecular Biology, Kyoto Univ. Hospital

◎要旨

1989年5月に開催された北米MUG第18回大会の報告とネットワークサービスG En i e上で展開されているMUMPSフォーラムの状況を述べている。北米MUG大会は事前参加登録数で596名を数える盛況なものであった。参加者の過半数は依然医療分野の関係者で、特にVAからの参加者が多くみられた。テーマとしてはオブジェクト指向、外部DBMSとの結合、ソフトウェアテストというup-to-dateなものも現れている。

◎キーワード：北米MUG、BBS、MUMPSフォーラム

◎北米MUG大会

北米MUGの第18回大会は1989年5月15日から19日にわたってWashington州SeattleのConvention and Trade Centerで開催された。VA(Veteran's Administrative Medical Centers)でのMUMPSによるシステムの成功や1988年に10億ドルの契約が成立したS A I C(Scientific Applications International Corp.)のDoD(国防省)医療機関群のシステム化プロジェクトなど北米でのMUMPS市場は活況を呈している。この状況を反映したものが大会の主題は、「MUMPS: The Language that Makes a Difference」を掲げていた。また、会場のあちこちで「MUMPS is business」というキャッチフレーズが見られたことから、この状況と関係者の期待がうかがえる。

大会参加者数は当日登録者を除いても596名あり、実際にはさらに多数が参加した。参加者の内訳を事前登録者のリストから分類すると、医療関係者が354名、その他が242名となっており、依然として医療関係者が多数を占めている。特に、全国のVAからの出席者が126名を数え、一大勢力をなしていた。その他の分野ではコンピュータ・情報産業や大学が多いが、金融5名、図書館2名、運輸4名をはじめとして公共機関、製造業などの業種からの参加も少なくなかった。

〒606 京都市左京区聖護院川原町53 京大病院分子病診療学 小森 優

Tel : 075-751-3645、FAX : 075-751-8098

大会の内容では、一般講演(Technical Papers, 37)よりワークショップ(Discussion Session, 66)の方が多くとられ、特定のテーマでより具体的な議論を交わす方向にあった。また、講習会(Tutorials)や教育セミナーも数多く(31)行なわれた。大会プログラムから演題内容をその分野といくつかのトピックスで大別すると表1のように分けることができた。

分野別演題数

	一般講演	ワークショップ
医療分野応用	11件	27件
非医療分野応用	1件	2件
装備・ソフトウェア	21件	38件
国際化	4件	0件

トピックス別演題数

VA関係	10件	20件
COSTAR関係	4件	6件
オブジェクト指向	3件	0件
教育	1件	3件
ソフトウェアテスト	1件	1件
MUMPS/非MUMPS	2件	4件
システム管理	2件	8件
ネットワーク	2件	8件
SQL、DBMS	1件	4件
メインフレーム	1件	3件

表1 北米MUGでの演題数の分類

この他に展示に30社(団体)が出展している。古参のDEC、DGといったミニコンメーカーにIBMが加わり、PCベースのMUMPSベンダーが数多く出品していた。

見学(Site Visits)としては、地元Seattle Veterans Administration Medical CenterでのVA kernelによるシステムやBoeing Employees Credit Unionのシステム、Laboratory of Pathology of Seattle、ワシントン大のDepartment of Laboratory Medicineの4カ所で行われた。

デモンストレーションはMUMPS tutor, COSTAR, CAI, MUMPS/VM, Technicalの5つのセッションがあり、特に、DECの会場内ネットワークMUGnetやIBMの9375-60に多数の端末/PCを接続したMUMPS/VMを公開するMUG Discovery Roomなどが大きなスペース

を占めていた。

期間中にはMUG内のユーザーズグループ集会在6件 (COSTAR, DASL, IBM, MSM, M/USERS, NAVSEA) が催されている。

◎G E n i eにおけるMUMPSフォーラム活動

わが国においてもBBSによる活動が盛んになってきたが、米国ではMUMPSを対象としたフォーラムが活動を続けている。これはGE社の主催するBBSのG E n i eである。ここでは、MUMPSに関するQ & Aや製品紹介、MUGに関するニュース、PDSなどを扱っている。以下にこのフォーラムでのタイトルを示す。日本MUGでも1989年12月よりN i f f t y S e r v e上でMUMPSフォーラムを開始する。活発な活動を期待したい。

GEnie MUMPS Page 550

MUMPS RoundTable

Library: ALL Libraries

1. MUMPS Bulletin Board
2. MUMPS RT Conference
3. MUMPS RT Libraries
4. About the RoundTable
5. RoundTable News 880401

MUMPS Bulletin Board

- Category 1 MGlobal Products ***
- Category 2 VAR (Value Added Reseller) Products
- Category 3 MDC (MUMPS Development Committee) Forum
- Category 4 News on MUMPS and the MUG (MUMPS Users' Group)
- Category 5 MUMPS and CCSM Language forum
- Category 6 Forum for the Veteran Administrations Database package.
File Manager.
- Category 7 PICK
- Category 8 Technical Support (CCSM-PC & MacMUMPS)

反復切断補正法による血液検査項目の臨床参考値の設定の試み
On a Decision of Clinical Reference Values of Blood Tests
using Iterative truncation Method with Correction

志賀修一、小柳逸子*、江見安一*、神奈木玲児*、小森 俊**

S. Shiga, I. Koyanagi*, Y. Emi*, R. Kannagi*, M. Komori**

京大病院検査部*、分子病診療学**

Dept. of Clinical Laboratory*, Dept. of Clinical Mol. Biology**

Kyoto Univ. Hospital

近年における臨床検査の普及とともに、各病院において大小様々なコンピュータが導入され、システム化に力が注がれている。当院においてもマンブス (MUMPS) を言語としてDEC社のコンピュータを長期間にわたり使用されてきた。私達の検査室では約8年前にPDP 11/23が最初に導入され、現在はマイクロPDP 11/73が使用されている。

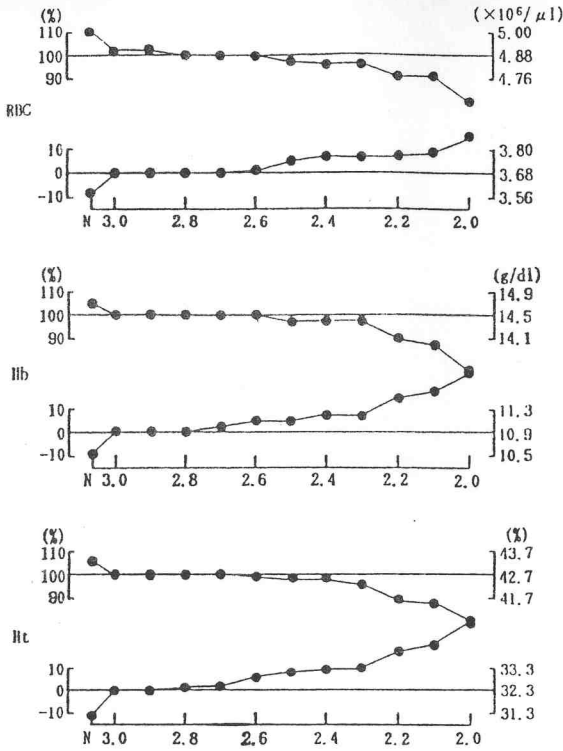
コンピュータ導入に伴い、検査データのデータベース化によって多量のデータをプールできるようになった。そこで私達は当院における参考値を求める計画をたて、参考値を求めていく上で必要となる度数分布表の作成、歪度 (skewness) の算出および反復切断補正法 (臼井法) を自分達でプログラミングしてデータ解析を進めることとした。今までの統計解析では血液学的な参考値の設定に反復切断補正法の利用はきわめてまれであった。私達は数年前より血液学的な参考値の設定方法について検討し、以下のような手順を採った。

まず、NR (Normal Range) というファイルに血液学的なパラメータ (WBC, RBC, Hb, Ht, MCV, MCH, MCHC, PLT, RDW, MPV, PDW, LYP(%), LYN(#)) の計数値をプールし、それぞれの項目における年齢別、男女別について度数分布表を作成する。その度数分布表より確率紙上に累積分布曲線をプロットし、それと同時に歪度の検定を総合して、それぞれの項目について正規性を示すと思われる変換方法を決定する。正規性を示す変換について、切断係数を2.0から3.0まで0.1ずつ変化させて、それぞれを用いて反復切断補正法を行い、最もデータの安定した切断係数から平均値 (\bar{x}) と標準偏差 (S. D.) を使って、上限値 ($\bar{x} + 1.96S. D.$)、下限値 ($\bar{x} - 1.96S. D.$) を求めた。求められたデータが変換されておれば逆変換を行った。分布型が決定できた場合はノンパラメトリック (パーセントイル法) で上限値、下限値を求めた。

このような手順での参考値の設定の試みに対して、最近では種々のソフトを利用し、データ解析を進めているところが多い。しかし、解析方法や解析結果について理解できないケースが往々にしてみられる。その点、自分達でプログラミングしていると、いろいろな失敗がある反面、得るものも多いと思われる。

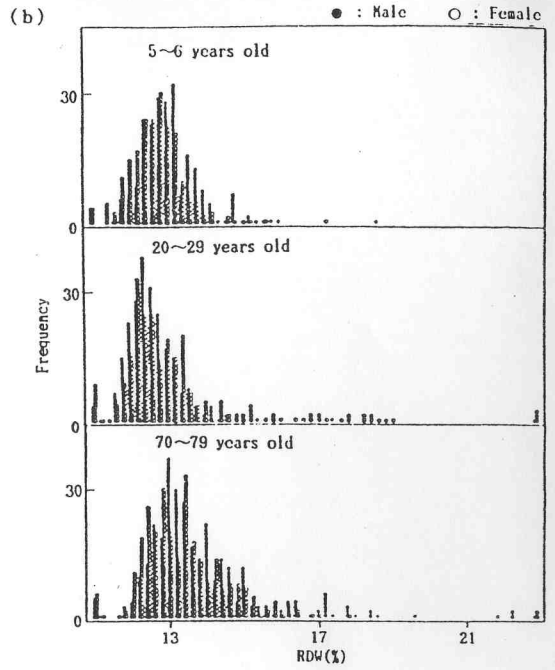
(参考文献)

志賀修一、小柳逸子、神奈木玲児：臨床病理、vol. 8、No. 1、pp. 93-103、1990



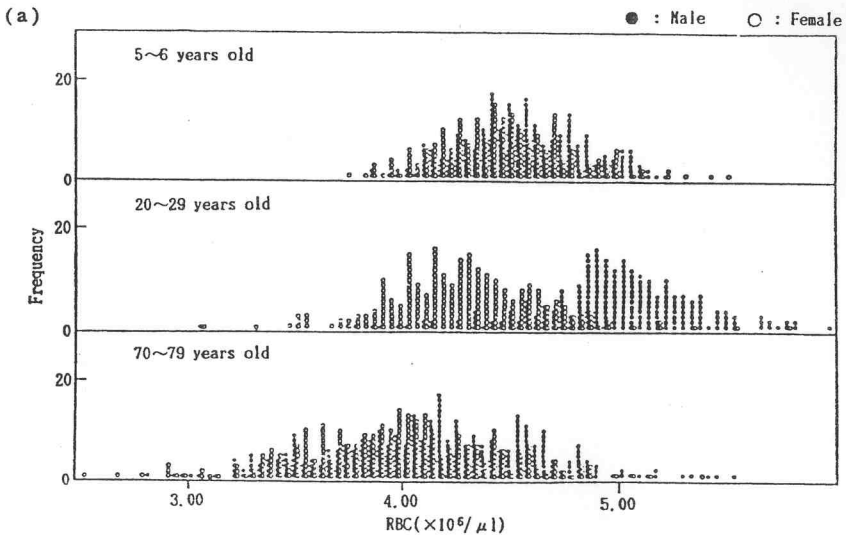
↑

☒ 1 Effect of the truncation coefficient on the reference values. The truncation coefficient for the iterative truncation was varied from 2.0 to 3.0. The calculated reference value was plotted against the truncation coefficient. One example taken from the data on 20~29 year old female population is shown.



↕

☒ 2 Age-dependent change in the distribution patterns of RBC (a) and RDW (b). Closed circles indicate the histogram of male, and open circles indicate that of female. Upper panel, 5~6 year old population; middle panel, 20~29 year old population; lower panel, 70~79 year old population respectively.



大量漢字グローバルのバッチ転送

Batch transfer of amount of Kanji globals

* 田久 浩志、大櫛 陽一、** 馬場 謙介
Takyu Hiroshi, Ogushi Yoichi, Baba Kensuke

* 東海大学医学部 医学情報部
* Department of Medical Information
School of Medicine, Tokai University

** 国立埼玉病院研究部
** Department of Clinical Research
National Hospital of Saitama

要約

VAX/DSMの漢字で記述されたデータベース(120Mbyte)を、PC9801に転送し保存する事を試みた。この際、漢字コードの変換、転送速度の短縮、MS-DOSのファイルで扱える大きさへのグローバルの分割等を考慮しなくてはならない。このため、PC側でTEAC社RS-2を使用して、データの記録を試みた。RS-2はDATテープ上を61.2Mbyteごとに区切り全体で1Gbyteのデータを、MS-DOSのファイルとして記録することができる。PC9801とVAXの接続にはDECNET-DOSを用いた。

本報告では、各種のデータ転送手法の速度の比較を行い、最適な転送手法の選定、転送データの容量の推定などについて考察した。

Author tried to transfer 120 Mbyte Kanji globals from VAX/DSM to NEC PC9801 via DECNET. Kanji-code conversion and transfer speed up must applied to the transfer procedures. Because MS-DOS could treat the filesize up to 61.2Mbyte, TEAC RS-2 was used to storage the data. This RS-2 record the data on the partitioned 61.2Mbyte file on the DAT tape, and can storage up to 1 Gbyte.

In this report, the optimization of transfer procedures, estimation of data size of transfered data was discussed.

キーワード：大量データ、バッチ転送、DECNET、転送速度

Key Words: Kanji Globals, Batch Transfer, DECNET, Transfer speed

〒259-11 神奈川県伊勢原市望星台

Bousei-dai, Isehara, Kanagawa, 259-11 Japan

1. はじめに

今回我々は、VAX/DSMで構築した大量の漢字グローバルデータを、PC-9801上のSP-MUMPSで使用する必要が生じた。そこでSP-MUMPSでデータを扱う前に、大量の漢字グローバルデータをDECNETを経由してPC9801へ転送する試みを行った。以前の報告(1、2)は、単純な英数字のみのデータであったが、今回は、漢字コードの変換、転送速度の短縮、MS-DOSのファイルで扱える大きさへのグローバルの分割等を考慮しなくてはならない。このため、PC側でTEAC社RS-2を使用して、データの記録を試みた。RS-2はDATテープ上を61.2Mbyteごとに区切り全体で1Gbyteのデータを、MS-DOSのファイルとして記録することができる。PC9801とVAXの接続にはDECNET-DOSを用いた。

本報告では、各種のデータ転送手法の速度の比較を行い、最適な転送手法の選定、転送データの容量の推定などについて考察した。

2. 対象と方法

転送対象とするデータは、漢字で記述された一種の辞書である。以前、このデータによるデータベースを、VAX8300の200Mbyteのディスク上のDSMのボリュームセット内に120Mbyteまで構築した。

このデータを今回転送するにあたり、以下のような問題が存在した。

- *使えるVAXの空きディスク容量は80MBYTE程度のため、一度グローバルをVMSのファイルにして転送する手法は採用できない。
- *DSMの漢字コードをPC9801用のシフトJISコードに変換する必要がある。
- *使用したPC9801のMSDOS3.1では扱うディスクの大きさが最大61.2MByteであるため、グローバルを分割して転送する必要がある。
- *転送時間がかかるため、人手の介入はできるだけ排除したい

これらの条件のため、以前の報告(1、2)のように、一度グローバルをディスク上のシーケンシャルファイルに落し、その後漢字コード変換ユーティリティーKCODEを使用して、VMSの漢字コードをシフトJISコードに変換し、その後転送することは難しい。以下に、これらの問題にどのように対処したかを示す。

転送するデータが大量のため、手持ちのハードディスクで全データを記録するのは不可能な事がわかった。そこで、TEAC社RS-2を使用して、データを記録することを試みた。RS-2はDATテープ上を61.2MBYTEごとに区切り全体で1GBYTEのデータを記録することができる。本機器を使用するとパソコン側では、RS-2をあたかもディスクのように見なしてデータの入出力をすることが可能となる。

PC9801とVAXを接続するには、DECNET-DOSを使用した。このシステムはNFT COPYとFALの2種類のファイル転送手法を持つ。両者の特徴を表1に示す。

一連のデータの転送方法は、表2に述べる項目を考慮して対処することより何種類かの方法が考えられる。今回試みたデータ転送手法の内容を表3-表5に、また転送時間のコントロールとして用いたグローバルの操作手法を表6にしめす。今回、VAXとPC9801の処理速度が異なり、両者の間の同期をとるためにVAX上でMail Boxを使用する方法を、転送手法の一つとして採用し

表1 DECNET-DOSでのファイル転送手法

手法	特徴
NFT COPY FAL	PC側で起動し、VAXとPCの間でファイルの転送を行う PC側をVAXのディスクと見なせるように設定する。

表2 データ転送にあたっての検討事項

検討事項	略号
DSM漢字コードよりVMS漢字コードへの変換	DSM→VMS
VMS漢字コードよりシフトJISへの変換	VMS→SJIS
VAX側でのデータの出力方法	データ出力
PC9801側でのデータの入力方法	データ入力
61. 2Mbyte以上のデータ転送の可否	大容量データ

表3 MAIL BOXを使用したデータ転送手法

検討事項	対応
DSM→VMS	\$ZCALL (%KJVMMS)
VMS→SJIS	NFT COPYの実行時PC側で自動的におこなう
データ出力	MAIL BOXへVMSのファイルとして出力
データ入力	NFT COPYでVMSで作られたファイルをRS-2に転送する
大容量データ	可

表4 FALでRS-2をDECNET上のディスクとしたデータ転送手法

検討事項	対応
DSM→VMS	\$ZCALL (%KJVMMS)
VMS→SJIS	VAX側で出力をシフトJIS漢字デバイスとして扱い コード変換をする
データ出力	RS-2をDECNET上のディスクとみなしてファイルを 出力する
データ入力	FAL
大容量データ	不可

表5 FALでRS-2に拡張子. KANの形式でファイルを出力する方法

検討事項	対応
DSM→VMS	シーケンシャルファイル名の拡張子を. KANとすることによりDSM-VAXの漢字コード変換を自動的に行う。
VMS→SJIS	VAX側で出力をシフトJISの漢字デバイスとして扱い、コード変換をする
データ出力	RS-2をDECNET上のディスクとみなしてファイルを出力する
データ入力	FAL
大容量データ	不可

表6 比較に用いたグローバル操作手法

コントロール名	内容
コントロール1	グローバルを\$O関数で単純にサーチする
コントロール2	ボリュームセットと同一のディスクへのシーケンシャルファイルとして、グローバルを出力
コントロール3	磁気テープ上のシーケンシャルファイルとして、グローバルを出力

た。この手法は、プログラムで各種の制御ができるが、その分転送速度が低下することが予想されるが、どの程度他の手法と比較して転送時間が異なるかを検討した。

表4に示すFALを用いたデータの転送手法はRS-2をVAX上の単なるディスクと見なせるため、転送速度は早いことが期待される。しかし、MS-DOSで扱えるファイル容量の制限を受けるため61.2MByte以上のデータを扱う事ができない。表5に示す拡張子. KANを使用する方法とは、DSMが. KANのファイルにデータを出力する場合、内部で自動的にDSM-VMSの漢字コード変換を行う事を利用する手法である。

3. 結果と考察

3-1) 転送時間の測定

今回は、資源の制限等もあり実際の120Mbyteの漢字グローバルの転送は実行せず、試験的に1Mbyteのデータを転送する場合、前述の手法を用いてどの程度の転送時間がかかるかを測定した。測定にあたっては、他のユーザーのいない深夜を選んで実験を行なった。転送時間と転送したバイト数あたりの関係を図1に示す。

図1で、ディスクとMTへ、グローバルをシーケンシャルファイルとして書き込むのに時間があまり差が無かった。これは、グローバルの存在するボリュームセットと同じディスクにシーケンシャルファイルを書き出したためと思われる。

図2にDECNETを経由して、RS-2にデータを転送したときの転送時間を示す。Mail

Boxを使用する方法は、VAX上のMail Boxを介してプログラムで各種の制御ができるが、その分転送速度が低下するため、今回の転送手法の中ではいちばん時間がかかる手法となっている。

表4のFALによる手法と、表5の拡張子、KANを使用する方法とは、後者がDSMが、KANのファイルにデータを出力する場合、内部で自動的にDSM-VMSの漢字コード変換を行なっているため僅かではあるが、前者より時間がかかっている。しかし、DSM側で漢字コードの変換を実行しているため、同様の処理を行うMail Boxを使う手法より比較すると、大幅に転送時間が短縮できていると言えよう。

図1と図2を比較すると、VMSのファイルにグローバルを出力する時間と、RS-2にデータを転送する時間がほぼ同程度である。これは、転送処理の多くをMUMPS内部での処理にとられているためディスク、磁気テープ、RS-2に対する処理速度が同程度になったと考えられる。また、このことはMUMPSでデータ処理を行う限り、RS-2を外部記憶媒体として用いても、VMSのファイルに比較してあまり遜色がないと解釈できる。

これらの転送速度を比較すると、転送するデータの容量が61.2Mbyteまでであれば、RS-2に拡張子、KANの形式でファイルを出力する方法が最も転送時間が短いので、大量データの転送に適すると言えよう。しかし、それ以上の容量を転送する必要がある場合、あるいはPC側の1ファイルに1グローバルを格納したいなどの要望のある時は、プログラムで転送を制御できるMAIL BOXを使用した手法も検討されるべきであろう。

3-2) 転送データの容量推定

今回のデータ転送に関しては、MS-DOSの扱うファイルが最大61.2Mbyteまでという制限があるため、転送データの容量は自分で管理しなくてはならない。特に、一つのグローバルを管理の都合上一つのMS-DOSファイルに記録したい時などは、グローバルの容量を事前に求めなければならぬ。この目的のためには、DSMの%GE(Global efficiency)ユーティリティを使用し、ポインタとデータが事前に使用しているディスクブロック数を求め、転送容量を概算した。ただし、DSMのポインタブロックはデータ圧縮をしているので、正確なデータ容量を求めるには、事前に転送対象とするグローバルの一部を抽出して一時的グローバルを作成した上で%GEを用い、実際に転送に必要なデータ容量とディスクブロック数の関係を求めた。

4. まとめ

近年、ネットワークの環境が整備されつつあり、同一機種の間では分散データベースが簡単に構築することが可能である。そのため、大量グローバルデータの転送は分散データベース等を使用して行なうのが望ましい。しかし、PC9801のSP-MUMPSよりVAXのDSMのグローバルデータを分散データベースとして見に行くような、MUMPSにおける異機種間データベースの開発は、まだ日が浅い。そのために、これから今回のように、パーソナルコンピュータの上に何十MBYTEもの大量のデータを転送して、データの移植をする機会は少なくないと考えられ、このような時に今回の実験データが役に立つと考えられる。

今回使用したRS-2自体の記録速度は決して高速とは言えないが、パーソナルコンピュータへのデータの移植の他に、ホスト計算機上のデータベースのバックアップをDECNETを介して遠隔地より簡便に行ないたい時などは、機械自体も小型で比較的廉価なため一考に値すると考えている。

参考文献

- 1) 大量データのバッチ処理: 田久 浩志、馬場 謙介、昭和59年11月 第11回日本MUMPS学会
- 2) 日本病理剖検輯報データベースのパソコンへの移送: 田久 浩志、馬場 謙介、昭和61年9月 第13回日本MUMPS学会

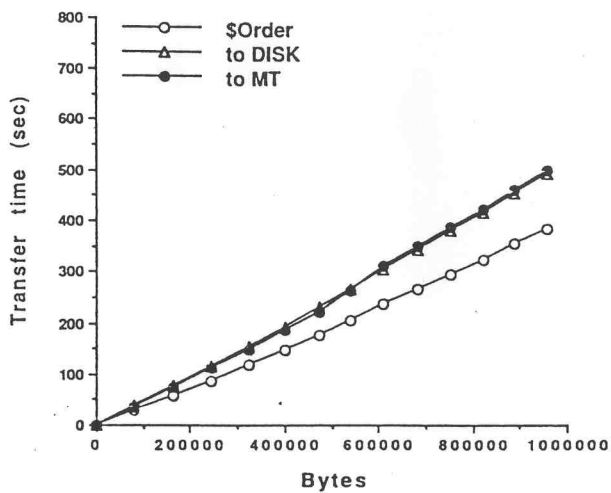


Fig. 1 Transfer time at the control procedures

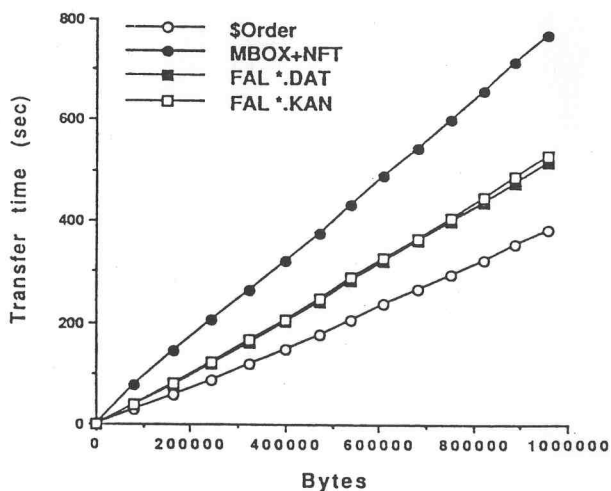


Fig. 2 Transfer time via DECNET

呼吸機能線画データベースの構築について

Development of Computerized Graphic database system
for Respiratory function

- 寺村 昌文, 古林 榮次郎, 野口 弘, 大杉 隆史
- Masafumi Teramura, Eijiro Kobayashi,
Hiroshi Noguchi, Takashi Ohosugi

大阪府立羽曳野病院
Osaka prefectural Habikino Hospital

MUMPSによる呼吸機能線画データベースの構築を行なった。
これにより、オーダリングを実施しながら検査結果を確認することができた。
線画は数値からバイナリに圧縮変換し記録した。
高解像度のパーソナルコンピュータとグラフィックドライバを用いることで
鮮明に表示することができた。今後、呼吸機能データをもとに各種、疾患の
解析が期待される。

We developed MUMPS graphic database system for respiratory function
on ordering system.
Graphic data is converted from numeric data to binary data.
This on-line graphic database is very compact.
The data are displayed on high resolution personal computer by graphic
driver.
We can easily follow and analysis patient's respiratory function data.

キーワード: 画像, パソコン, 呼吸機能, データベース

Key Words: Graphic, Personal computer, Respiratory Function, Database

1. はじめに

大阪府立羽曳野病院は全国でも有数の呼吸器・アレルギー疾患の基幹病院である。
1988年1月にCPUの入れ換えが実施され、第3世代の病院トータルシステムが
開始された。その移行に際し、従来のANK端末の他に、グラフィックが取り扱える
(解像度1000ドット以上) パーソナルコンピュータを病棟他各部所に配置をおこなった。
羽曳野病院では、従来より呼吸機能の生理学検査がさかんに行なわれている。
通常生理機能などの検査結果はその測定器が持つFDDに保存されてはいるが、有効
な利用はされていなかった。
データの有効利用と病棟での診療支援を目的に、ホストコンピュータと検査システム
とをオンライン接続し、呼吸機能のデータベースとしてMUMPSによる構築を実施
した。

〒583 大阪府羽曳野市はびきの3-7-1
Habikino 3-7-1 Habikino city, OSAKA

2. システム化の目的

- 1) 呼吸機能検査データの有効利用
全病棟でオーダを実施しながら照会ができる。(診療支援)
検査機器のみによるデータのデッドストックとならない
(FDDの保存だけに終わらない)
- 2) パソコン端末の有効利用
グラフィックにより曲線データを視覚化する
- 3) データ圧縮につとめる(数値データをバイナリデータに変換)

3. システムの構成

図1. に機器構成を示す。

生理機能検査室における精密呼吸機能検査の測定はSYSTEM55(ミナト医科学株式会社)を用い、1986年4月より実施している。(1日あたり患者4~5人)測定された結果はいったん付属するFDDに蓄積され、1日の業務終了時に病院システムに接続されている漢字端末VG720(ビクタ)を経てVAX8530(DEC社)に取り込まれる。

病棟で使用しているパーソナルコンピュータはPC98XL(NEC)、グラフィックをコントロールするドライバーはEGR98XA(CANOPUS)をMSDOS(V3.1)のもとで使用。

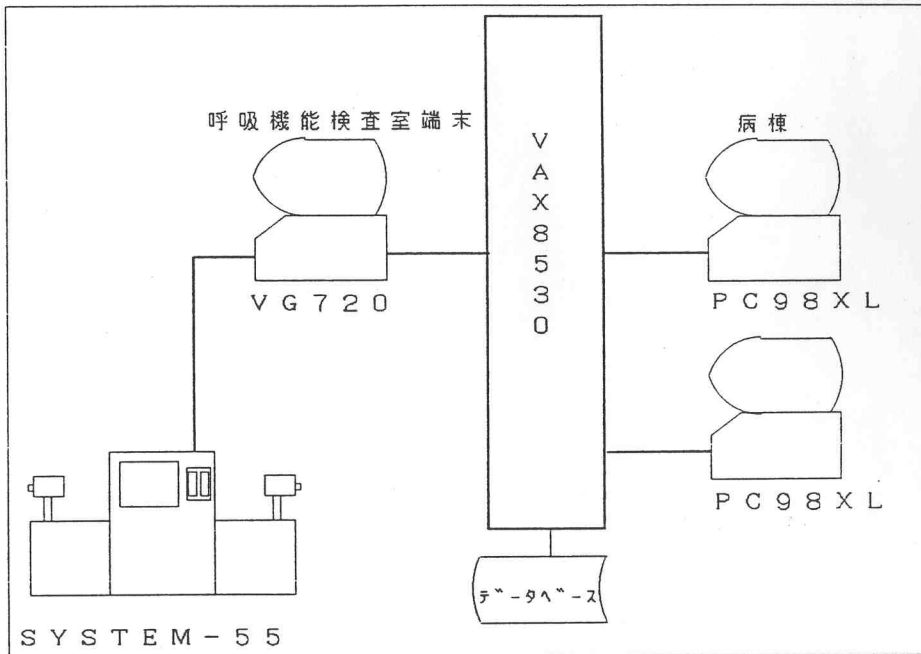


図-1 システム機器構成図

4. 呼吸機能線画データベースの種類
参照できる線画曲線は5種類

- 1) Flow-Volume 図-2 参照
- 2) Spirogram
- 3) Closing-Volume 図-3 参照
- 4) N₂ Washout
- 5) Dico-Curve

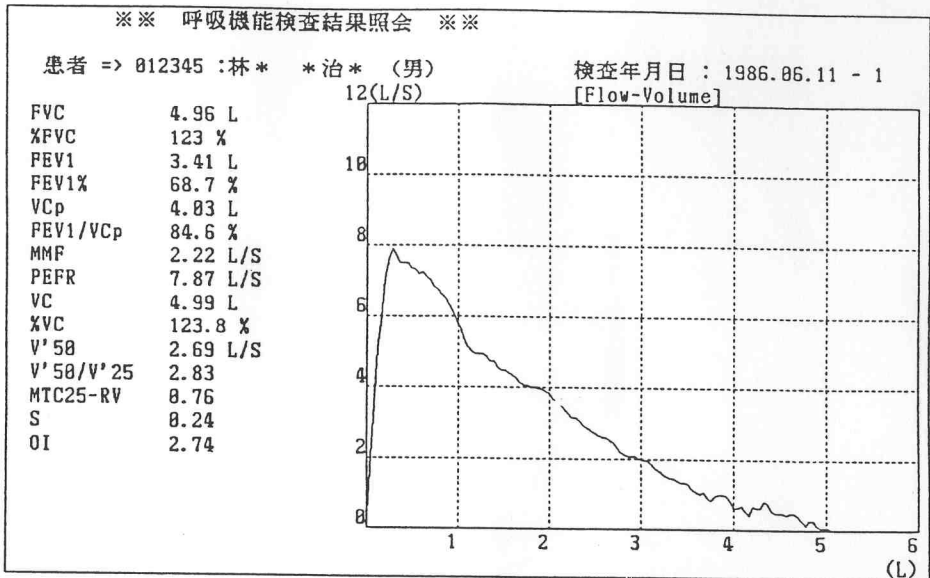


図-2 Flow-Volume

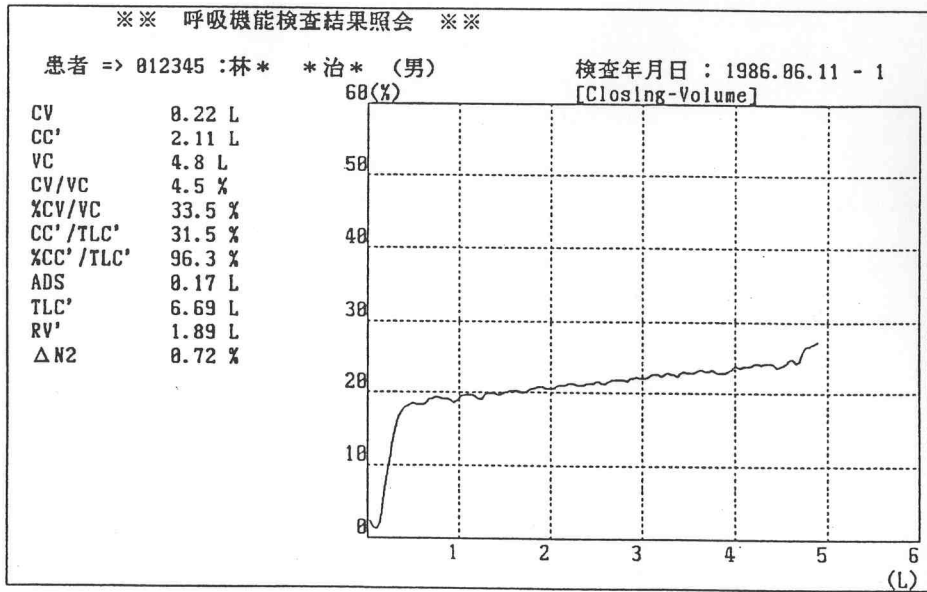


図-3 Closing-Volume

5. 曲線の数値データ（バイナリ変換前）とバイナリ変換後データの比較

1989年3月14日 現在	検査	1,921件
数値データ（バイナリ変換前）		23,350 (DSM Block)
バイナリ変換後のデータ		6,280 (DSM Block)
圧縮比		26.8%

6. MUMPSによるEGR98の使用方法

以下にMUMPSコーディングによるグラフィックドライバ使用方法を示す
MUMPSはグラフィックドライバ指定のコマンドとそのデータをパソコンに出力するだけで自由に描画できる。
また、画面のイメージをプリンターにそのまま出力することができるので用紙に記録することもできる。

ラベル	機能説明	コーディング内容
ZGINI	グラフィック初期化	W \$(27,7)_"INIT"_\$\$(13,10) Q
ZGCLS	アクティブ画面クリア	W \$(27,7)_"CLS"_\$\$(13,10) Q
ZGCIR	円・楕円・円弧描画	W \$(27,7)_"CIRCLE"_"_XMIN_"_"_YMIN_"_"_XMAX_"_"_YMAX_"_"_COL"_\$\$(13,10) Q (円描画の例)
ZGLIN	2点間の直線/四角	W \$(27,7)_"LINE"_"_XMIN_"_"_YMIN_"_"_XMAX_"_"_YMAX_"_"_COL_"_"_GF_"_"_GL_"_"_GPL"_\$\$(13,10) Q
ZGPRN	プリンタへハードコピー	W \$(27,7)_"COPY 3"_\$\$(13,10) Q
ZGPST	画面にドットをセット	W \$(27,7)_"PSET"_"_X_"_"_Y_"_"_COL"_\$\$(13,10) Q
ZGCSR	グラフィックモード設定	W \$(27,7)_"SCREEN 0,0,0,1"_\$\$(13,10) Q

注釈 XMIN : X最小値 , YMAX : Y最大値 , GL : ラインスタイル
XMAX : X最大値 , COL : ドット番号 1 , GPL : ドット番号 2
YMIN : Y最小値 , GF : 描画コード , X : Xの位置 , Y : Yの位置

7. まとめ

通常、生理機能検査データ（呼吸機能検査）はその性質がクローズなシステムであるため、データが有効に利用されずデッドストック化されてしまうことが多い。それらの検査は、各パラメータの値だけではなく、各曲線の形状、傾き、大きさに意味がある。

今回、我々はそれらをオンラインデータベース化することで、データの共有が可能となり、パーソナルコンピュータを用いることで表示がグラフ化された。

これによりオーダリングをしながら患者の病体変化を時系列に参照することができるようになった。

MUMPS言語の使用及びデータの圧縮につとめることで、資源の有効利用、コストパフォーマンス、メンテナンスビリティの向上が計られ、機能の拡張が可能になった。

仕様書自動作成ツールの開発

Development of auto documentation tool

- 野口 弘, 古林 榮次郎, 寺村 昌文
- Hiroshi Noguchi, Eijiro Kobayashi,
Masafumi Teramura

大阪府立羽曳野病院

Osaka prefectural Habikino Hospital

要約:

我々は規定に準拠したプログラム群の内部構造を解析しその仕様書を自動作成するツールを開発した。このツールは変数抽出、プログラムオーバーレイ解析などともトッキングし、仕様書として出力するときにはプログラム仕様書、HIPOの図式目次、ファイルアクセスの一览、変数表を連続して出力することが可能である。

仕様書自動作成ツールによりドキュメント作成作業の効率化、標準化が達成出来た。またその副次的効果としてルーチン作成の標準化及び規定の準拠などプログラム作成者のモラルアップなどの効果がみられた。

Abstract:

We developed auto documentation tool for MUMPS application programs conformed to our due form. Using this tool we raised the level of efficiency and standardization. And effects such as programing standardization and programing follwed to due form are obtained.

キーワード : 仕様書、標準化、保守作業の効率化

Keywords : documentation, standarzation, maintenancebility

[はじめに]

仕様書の作成はシステムの文章化には不可欠なものであるがその作業の性格上疎かにされがちである。

仕様書の作成という作業はプログラム作成に比較すると労多くして報いの少ない様に考えられているがシステムの保守を考慮にいれると重要なポイントを握る作業である。この作業をいかに軽減するかがシステム全体の効率にも関係してくる。

大阪府立羽曳野病院では従来より標準化の観点にたち羽曳野病院ソフトウェア開発規定を作成しその規定に従ってプログラム及び仕様書の作成を行ってきた。

〒583 大阪府羽曳野市はびきの3-7-1

Habikino 3-7-1 Habikino city, OSAKA

プログラムの変数抽出、プログラム間のオーバーレイ解析などはメカ提供のユーティリティにもみられ、我々も上記の様なユーティリティを作成し仕様書作成の一助としていた。しかし仕様書作成時にはこれら以上にプログラム内のモジュールの関連や各モジュールでどのファイルがどのようなアクセスのされかたをしているかが問題になってくる。また規定の準拠が進めば進むほど汎用ユーティリティの共通モジュールの使用などが増加しそれらがどこで使用されているかが重要な情報となって来る。

今回上記の規定に準拠したプログラム群の内部構造を解析しその仕様書を自動作成するツールを開発したので紹介する。

[ツールの概要]

このツールは従来から使用していた変数抽出、プログラムオーバーレイ解析などともトッピングしプログラム群の解析、プログラムのリスト、サマリーのリスト、プログラム・グローバルの使用場所の検出、不要ラベルリストの出力、変数表のメンテナンス、変数表の出力及び仕様書の出力からなっている。

プログラム群の解析を実施する場合2つの方法があり、1. 一定のプログラム群を選ぶ、2. ルートとなる最初のプログラムを選択する、の何れかを選択する。2. を選択した場合指定プログラムがCallしているすべてのプログラムを探し出しさらにそれらからCallされているサブルーチンもすべて検出し解析の対象とする。

解析の内容は、変数名の抽出、Call Overlayプログラムの抽出、各プログラム内のモジュール間の関連の解析、各モジュールでのグローバルファイルの参照の内容、(Read, Writeの区別)などである。抽出された変数名は変数表処理プログラムに転送され、共通変数の内容登録、変数表のメンテナンス、プログラム群共通変数の登録などが可能である。

*** ルーチン解析 & 仕様書自動作成ルーチン ***

1. ルーチンセットの解析(仕様書自動作成)
2. ルーチンとクロスリファレンス、シンボリックラベルの出力
3. 解析ルーチンの再出力
4. サマリーからのクロスリファレンスの再出力
5. ルーチン、グローバルの使用場所の検出
6. ルーチンセットの構造解析(Do, Gotoの解析)
7. 不要ラベルリスト
8. 共通モジュール・ルーチン テーブル メンテナンス
9. グローバル・モジュール構成メンテナンス
10. 仕様書自動出力
20. 変数表処理 (^%ZVARIA)
21. 変数ファイル転送 (^%ZFSUM -> ^%ZFVAR)

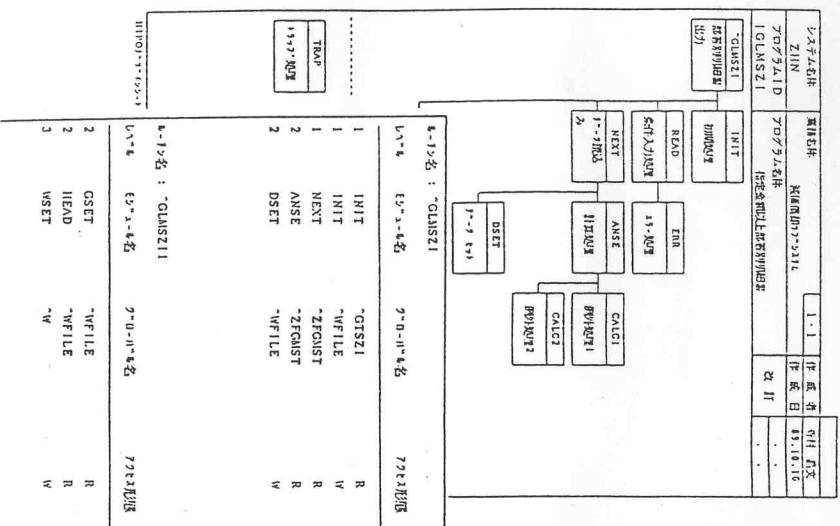
* ルーチン解析 & 仕様書作成 オプション :

図1. 仕様書自動作成ツールの概要

```

1: CLMSZ1 1 ; 科目 品文 (COPYRIGHT 大阪府立知覚野病院) : 10:53:58 9-NC
2: START 1 D INIT I %SE%-DOWN% U 0 W !!:*** _%ZSYS_% の起動を中止中
3: 4: D READ I ONE%/" G EXIT
5: 5: D NEXT
6: 6: D %CLMSZ1
7: 7: G START
8: 8: K QUIT
9: 9: 10: ; 追加処理
11: 11: INIT 1 K K (%FILE%)
12: 12: S %SE%-S %ZSYS%-QUW% %ZUCI%-VAD% D %ZDOP% I %SE%-DOWN% QUIT
13: 13: D %H01%-M11
14: 14: S YV% %ZSZ1%(M11)
15: 15: S LIN%
16: 16: S D%
17: 17: QUIT
18: 18: ; 実行
19: 19: READ 5 %LIN%
20: 20: U 0 RI I %JOB
21: 21: S %M12%
22: 22: S %M12%
23: 23: RQ QUIT
24: 24: ; 終了
25: 25: NEXT 1 I %JOB
26: 26: 27: D OSET
28: 28: ; 終了
29: 29: ANSE 1 S (%JOB)
30: 30: 31: ; 終了
31: 31: 32: QUIT
32: 32: ; 終了
33: 33: CALC1 1 S (%JOB)
34: 34: ; 終了
35: 35: ; 終了
36: 36: ; 終了
37: 37: ; 終了
38: 38: ; 終了
39: 39: CALC2 1 S (%JOB)
40: 40: ; 終了
41: 41: ; 終了
42: 42: ; 終了
43: 43: ; 終了
44: 44: ; 終了
45: 45: DSET 1 S (%JOB)
46: 46: ; 終了
47: 47: ; 終了
48: 48: ERR 1 S (%JOB)
49: 49: ; 終了
50: 50: ; 終了
51: 51: ; 終了
52: 52: ; 終了
53: 53: ZALL W %JOB
54: 54: ; 終了
55: 55: ; 終了
56: 56: ; 終了
57: 57: ; 終了

```



プログラムの属性 (実行中)		システム名	実行日時	実行場所	実行結果
ZIN	プログラムの属性	プログラムの属性	08.11.18	大野 邦雄	0.5人日
PROGRAMID	プログラムの属性	プログラムの属性	08.11.18	大野 邦雄	0.5人日
CLMSZ1	プログラムの属性	プログラムの属性	08.11.18	大野 邦雄	0.5人日

変数名	内容	備考
M11	実行日時 (YYYYMMDD)	
M12P	実行日時 (YYYY.MM.DD)	
%SE	'%ZDOP%' or 'START-ERR %ZDOP%' (%ZDOP%)	
%ZSYS	'%ZDOP%' %Z-4%	
%ZUCI	'%ZDOP%' UC1%	
COL	'%ZDOP%' %Z-4% (M11)	
D	'%ZDOP%' (%)	
D3	'%ZDOP%' (%)	
DD	実行日時	
DEV	実行日時 (M11)	
FL	'%ZDOP%'	
I	'%ZDOP%'	

図 2. ルーチンと仕様書の例

モジュール間の関連・構成図も間接指定などの使用も考慮して解析後その内容をマニュアルで修正・追加が可能である。図1にこのツールの内容一覧を示す。

主要なユーティリティ、共通変数、共通モジュールなどの内容はテーブルとして登録してあるので一切入力する必要はなく、モジュールの処理内容などはプログラム作成時のコメントを使用しているので、手作業として残されたものは、HIP0の詳細ダイアグラムの作成と各プログラムで使用している個別変数の説明の入力だけとなった。

解析の終わったプログラム群は個別変数の説明入力を終わると仕様書として出力が可能である。

出力時に入力する項目は使用作成者およびプログラム作成者のオハレタコート、仕様作成日、工数、業務名称、プログラム名称の六項目のみである。

仕様書はプログラム仕様書(プログラム間のオーバーレイ解析、ファイルのアクセス状況)、HIP0の図式目次(プログラム間の関係、プログラム内部のモジュール間の関係)、プログラム内のモジュールごとのファイルアクセスの一覧、変数表と連続して出力される。

図2に例としてプログラム ^GLMSZIの概要とそれにより作成された仕様書の例をあげる。

[結果およびまとめ]

MUMPSは修得が簡単である反面プログラミングの標準化が達成されず各個人ばらばらなプログラミングがなされ個人のシステム、趣味のシステムから脱却出来ないでいる例がよくみられる。しかしある程度以上の規模のシステムとして成功させるためには、プログラム開発・保守に関する標準化のみならずドキュメンテーションの標準化も必要不可欠である。一定期間ごとにドキュメントの存在しないプログラムは消去するぐらいの処置を実施しないと使用使途不明のプログラムが増加しシステムとして自滅することも考えられる。

ドキュメント作成の標準化もシステム全体の効率と密接に関連している。ドキュメントの標準化が達成されればシステムの保守性も向上しシステム全体の効率も向上する。

仕様書自動作成ツールを使用することによりプログラム作成者の負担を軽減しプログラム作成に、より労力をつぎこめるようになった。またプログラム作成時に入力したコメントがそのまま仕様書に生かせるためプログラム作成時のモラルアップにも繋がった。

また規定に適合しないルーチンの検証にも利用可能でシステム全体の標準化の推進にも役立っている。

以上仕様書自動作成ツールによりドキュメント作成作業の効率化、標準化が達成出来た。

機械式カード検索機からパソコンマンプスへ

—— 入院病歴管理

Peek-a-Boo to Personal MUMPS in Chart Library

馬場 謙介,* 杉崎 登,** 三宅 和夫,** 鈴木 孝始,*** 加藤 寿夫,** 石名田 洋一**
Kensuke Baba*, Noboru Sugizaki**, Kazuo Miyake**, Koshi Suzuki***, Yukio Katou** and Youichi Ishinada**

*国立埼玉病院 臨床研究部, **同 検査科, ***同 病歴室, *同 整形外科
*Department of Clinical Research, National Hospital of Saitama, **Clinical Laboratory,
***Chart Library Division and *Division of Orthopedic Surgery

ABSTRACT

Since 1974, a mechanical peek-a-boo has been used for tasks to search pastry or currently administrated patient registration number from his or her name and to manage his or her intrahospital chart transfer including chart lending and returning between chart library and doctors. Recently the tasks are become to be treated by a personal MUMPS. In the new system, charts are managed by the current location of the chart (not by transfer event; lending, returning etc). The new system works like as specially designed word processor as that clerk edit the old place (A) of chart to the new place (B). When A and B are doctor A and library, respectively, this editing corresponding to the event that doctor A return the chart to the library.

はじめに

本院では、過去約20年間、機械式カード検索機を、入院病歴の管理に応用してきた。導入当時は最先端の技術を取り入れた病院として評価されていたが、今日では、時代遅れになってしまった。今回、機械式カード検索機の代りにパソコンマンプスで入院病歴の管理を実施することを計画し、徐々に実務に組み込まれているので報告する。この報告では、開発過程における業務担当者(プログラム依頼者)と、電算化担当者(プログラマー)とのやり取りについても記録にとどめるように努力した。

Kensuke Baba, Department of Clinical Research, National Hospital of Saitama, 2-1 Suwa, Wakou, Saitama, 351-01 JAPAN (TEL: 0464-62-1101 ext 558, 316 FAX: 0464-64-1138)

パソコンマンブスに至る経緯

医事課ならびに病歴室の変革:

本院は、入院病歴の管理の目的で1970年に病歴室を設置し、4年後に専用の機械式のカード検索機(Fig. 1,2)を設置した(Table 1)。以来、15年間、この機械式のカード検索機が入院病歴の管理の主役を演じてきた。1985年に、SHIS^{1,2)}が、医事会計計算業務を対象に導入されたが、このシステムは 今日でも病歴管理業務には拡張されていない。このような状況下で、今回、パソコンマンブスを入院病歴管理業務に応用することを試み、徐々に 実務に組み込んでいる。

機械式カード検索機:

本院の機械式カード検索機(JOMICS社・大沢商会)は 検索キーボード、穿孔機、本体から成る(Fig. 1)。本体に 検索カードを入れるトレイ(Fig. 2)を乗せて検索する。トレイのカード収容能力は 最大1300枚で、トレイは 本体から脱着可能である。(患者名から患者番号を知るための)患者名カード(Fig. 3)、(病名から該当患者を知るための)病名カード(Fig. 4)、退院要約カード(Fig. 5)、病歴貸出カード(Fig. 6)等を作り、夫々の目的に使用していた。病名カード、退院要約カードは 枚数が多いので、複数のトレイに格納せざるを得なかった(Fig. 7)。

カードへの検索キー情報の刻みは カードを穿孔機に装着し、キーボードを操作して行う。検索の際も、キーボードを操作する。カードに盛り込める情報は 基本的には 数字5桁であるが、キーボードは 仮名をキーにすることを想定して工夫されている。この機械では Table 2 の如く 仮名50音を10に分類しているが、キーボードの操作時には この分類を意識する必要はない。患者名カードは、姓の第1字、姓の第3字、名の第1字をキーとし、一枚のカードの裏表合わせて108例が書けるように設計してある。

カード枚数の増加と共に 本機による検索は 他の方法に置き換えられ、今日では、患者名カードによる患者番号検索のみに 使われている(Fig. 7)。しかし、患者名カードでさえ、症例数が 数10万症例となった今日では、鈴木、田中などの頻度の高い姓の場合、4枚のカードを要するようになり、頻度の高い姓の場合、数百の氏名から該当症例を探し出さなければならなくなっている。

パソコンマンプスへの移行

仕様書の検討:

上記の機械式検索機では、もはや入院病歴の管理ができなくなったので、入院病歴の'図書'管理をパソコンマンプスで実施することを計画した。入院病歴を管理する担当官は システムの仕様を メニューの形で 提示してきた。そのエッセンスは Table 3 の如くであった。著者らは この仕様で示された機能を 下記の如く置き換えてみた。

初 回 入 院 を 一 行 (症 例) 挿 入、 諸 情 報 入 力 に、
入 院 記 帳 を 病 歴 所 在 場 所 の 修 正、 入 院 日 の 入 力 に、
退 院 記 帳 を 病 歴 所 在 場 所 の 修 正、 退 院 日 の 修 正 に、
貸 出 記 帳 を 病 歴 所 在 場 所 の 修 正、 移 動 日 (貸 出 日) の 入 力 に、
返 却 記 帳 を 病 歴 所 在 場 所 の 修 正、 移 動 日 (返 却 日) の 入 力 に、
患 者 番 号 表 示 を カ ー ソ ル の 直 接 移 動 に、
貸 出 先 表 示 を 選 択 画 面 での 要 約 表 示 と 詳 細 表 示 兼 編 集 に、
貸 出 先 別 一 覧 を 選 択 画 面 での 要 約 表 示 と 詳 細 表 示 兼 編 集 に、

夫々、置き換えた。上記の右を見ると、「病歴の所在場所」を管理することで、全ての病歴の移動に伴う記帳業務がマネージできることが判る。また、上記の右は、全て、ワープロが持つ機能である。そこで、ワープロ風のプロトタイプを依頼者に渡し、試験運転をしてもらった。依頼者に渡したプロトタイプは、項目の画面上の位置などをパラメータで、与えるように設計されていて、画面設計とそれに伴うグローバル設計は、プログラマーが指導して依頼者が行った。

プログラムの変更を伴わないプロトタイプの変更:

依頼者は、プログラム自身に変更を加えないで機能を充実することができた。その内の主な二つを説明する。

依頼者は、「病棟」を「病棟(入院中)」、「病棟(病歴要約作成中)」の二つに分け、後者に病歴所在場所コードを新設した。これによって、病歴要約が未完成のため、病棟管理下にあつて病歴室に未返却の病歴を把握できるようになった。同様に、病歴室でも、奥倉庫、'アクティブ庫'、製本中等に細分し、対象業務範囲を拡張した。

プロトタイプには、備考の項目が用意されている。項目の名称は「備考」になっているが、備考も論理検索の対象となるように設計されているので、依頼者は、これを'本データ'と同様に利用することができる。依頼者は備考に、死亡、剖検、主治医等の情報を一定書式で記入し、それらの検索に応用している。

プログラムの変更を伴うプロトタイプの変更:

プロトタイプには、(画面に表示範囲を越えた症例の)患者番号を指定して、その症例を表示する機能が備わっていなかったため、この機能を追加した。この場合、先頭行に当該症例が、以下に19症例が患者番号順に表示される。同様に、患者名を指定して、当該症例から表示する機能を追加した。

プロトタイプにも、病歴所在場所別に病歴一覧を表示する機能はあるが、不十分であったので、補助画面(後述)を新規に追加した。

システムの画面と機能

選択画面:

システムを起動すると、Fig. 8 のような'選択画面'が表示される。この選択画面は、初期メニューの機能を兼備している。選択画面の上方4行の選択領域に、業務とカルテ所在場所とが、40項目並んで表示される。カーソルキー操作で、項目を選択するのであるが、カーソルが選択領域を越えようとするとき選択領域の範囲でスクロールする。また、カーソルキー操作を1秒休むと、Fig. 8 の約下半分のように当該カルテ所在場所にあるカルテの要約(カルテ番号と仮名氏名の6字)が自動的に表示される。ここで、リターンキーを押すと、Fig. 9 のように当該カルテ所在場所にあるカルテの情報が表示される。患者を選択し(表示範囲を越えればスクロール)、必要項目を編集できる。ここで、

- 1) デリミッター文字 "." は、挿入も削除も変更も不可
- 2) 重書モードでは、文字を削除しても、デリミッター文字 "." の位置は不変
- 3) 挿入モードでは、文字を削除すると、デリミッター文字 "." の位置は左遷
- 4) 挿入モードでは、文字を挿入すると、デリミッター文字 "." の位置は右遷
- 5) デフォルトは、重書モードに設定

である。つまり、この目的に特別に設計したワープロのように動作するように設計してあるので、データの書式を崩さずに、ワープロの快適な操作環境で、編集業務ができる。この、ワープロ風の快適な操作環境は、Xecute ^%e や Do %XR 等のスモールコンセプト^{3,4)}を多用とて実現している。

病歴室画面：

病歴室画面(Fig. 10)で扱うのは、(表現は病歴室となっているが)病歴室にある病歴だけでなく全病歴が扱われる。病歴室画面の機能は、Fig. 9 では、マスクされている次の機能がアクティブになる。

- 1) 直接アクセス(ESC/=)： 患者番号または患者カナ名 (数値か文字かで自動判断)
- 2) 行挿入(^O)： 一症例の挿入
- 3) 行削除(^Y)： 一症例の削除
- 4) 行復帰(^U)： ^Y で削除した症例の復帰

データベース：

本院では、一冊の入院病歴は、患者番号・連番で、定義されている。ここで、連番は第一回目の入院であれば .01、15回目の入院であれば、.15 である。編集と同時に、データベースが更新されるように設計した。設けたインデックスグローバルは、

```
^YLR(-1,-病歴番号)=""  
^YLR(1,患者仮名名,病歴番号)=""  
^YLR(2,病歴所在場所,病歴番号)=""
```

である。

新規入院の入力作業を軽減する目的で、本院では、外来初診受付が終了する午前11時30分に、毎日、SHIS システムからフロッピーで、患者仮名、患者番号、生年月日、性を移送する準備を進めている。これによって、入力業務は軽減され、編集操作で、入院病歴ができるようになる。

考 按

一般的にプログラマーは業務の内容をよくは知らない。逆に、業務担当者は 計算機をよくは知らない。まして、業務担当者は 担当のプログラマーの得手不得手を知らない。プログラムの作成に当り、両者の間のこの溝を如何に埋めるかは、難しい命題である。著者らが 今回この点で配慮したことを記しておく。

第一に配慮したのは、依頼者がパラメーターを指定して、画面設計できるようにしたことである。(最近、画面設計をスクリーン上で実施し、プログラムをジェネレートすることが、一種の流行になっているが、少なくともこの規模の画面設計か、設計しようとする画面の数が少ない場合は、パラメーターで指定する方がはるかに現実的である)。

第二の配慮点は、項目として、論理検索の対象となる備考を常設するにしていることである。前述の如く、依頼者は早速これを利用し、システムが扱う情報を豊富にしてくれた。著者らは、データベースバウンドの入力・編集画面にはこのような備考を必ず設けるようにしている。

第三の配慮点は、業務の解析に当って分析的な方向だけでなく、全体を包括する方向で業務を眺めたことである。その結果として、複数の業務をただ一つの「病歴の所在場所での管理」に集約できた。換言すれば、入院病歴管理を、貸出業務(病歴室→医師)、返却業務(医師→病歴室)、病棟預け業務(病歴室→病棟)等々のイベントで管理せず、病歴の所在で管理することにしたのである。一般に座標が n 個あると、 n 個の座標を間を結ぶベクトルの数は、 $n(n-1)$ 個になことから、座標による管理方法が有利と判断して、それを採用しのだが、以外な効用があった。それは、プログラムもパラメータも変更しないで、病歴要約作成中、製本中、奥庫等々を病歴の移動先に追加でき、業務範囲が拡張できたことである。実際にやったことは、(新任医師を想定して作ったルーチンで)病歴所在場所のコードを新設しただけである。この経験を通じて、「流通現象を管理する場合には、流通の切口中で管理せず、物品の現在位置で管理すると、グローバルとの対応とワープロ機能との対応が単純明解になり、且つ拡張性が増す」ことを、ステートメントとして残す。

院内に電子計算機ネットワークを持つ病院がどれだけペーパーレスで入退院の事務的処理をしているだろうか。現行の事務部の機構では、入退院の事務的処理は書類と印鑑で処理されている。本院では、医事課に隣接する病歴室で、入退院の事務的処理を実施している。従って、少なくとも本院では、高価な端末を増設する積極的な理由は見あたらない(Fig. 11 参照)。本システム導入前と本システムの経費を比較すると、Fig. 11 の如くである。元々病歴室で電子計算機化できる作業は、高々全体の 35% であって、パソコン以上の電算機化は、極めて投資効率が悪いと結論する。

本システムは、ワープロ風に操作できて快適である反面、旧SHISに長年、馴染んで来た職員に歓迎されるかどうかは、今後に残された課題であるが、一画面に同時に20症例が表示され、いつでも表示されている症例に、カーソル操作で、到達できるメリットは評価に値する。

む す び

ワープロ風でデータベースバウンドに設計した入院病歴管理システムが、本院で稼働しはじめたので報告した。報告に関係した機械化の経済効率についても私見を記した。報告の中で、設計で事前に工夫したこと、依頼者とのやり取りの中から得た知見を記録にとどめた。

文 献

- 1) 日本電信電話株KK: SHIS (Shared Hospital Information System). 全5巻, 1987
- 2) 巽 憲一 他: 入力専門者を活用した総合オーダーリングシステム. 第7医療情報学連合大会 7: 97-100, 1987
- 3) Baba K: On small concept *Mumps* 14(suppl): 73-76, 1988
- 4) Baba K: Small concept. *Mumps* 14: 33-43, 1988.

Table 1: History / Bill Division & Chart Library

Bill Division	Chart Library Section
Chart Library	
1970 Library	Administration Chart
1974	Peek-a-boo
1985 SHIS system*	
1989	Personal MUMPS

*: Shard Hospital Information System, NTT Ltd.

Table 2: Grouping of Katakana Characters

Group	Katakana
1	カ セ ミ モ
2	イ ソ ニ メ リ
3	ウ タ ヘ ユ
4	カ ツ ネ ヨ ル
5	ケ ス ナ ワ
6	エ ク ホ レ ヤ
7	ウ ノ フ ム ロ
8	オ チ ヒ ラ
9	キ ヌ ハ マ
0	コ シ テ ト ン

Table 3. Summary of Specifications

Customer's Spec.*	Present System
Administration	One line insert (^O)
Edit administration date	
Discharge	Edit discharge date
Lending	Edit chart location
Return Back	Edit chart location
List/Dr	Edit chart location
END OF JOB	ESC/Q
Miscellaneous (see text)	

*: The customer (a librarian of our hospital) wrote her specification sketch by menu items.

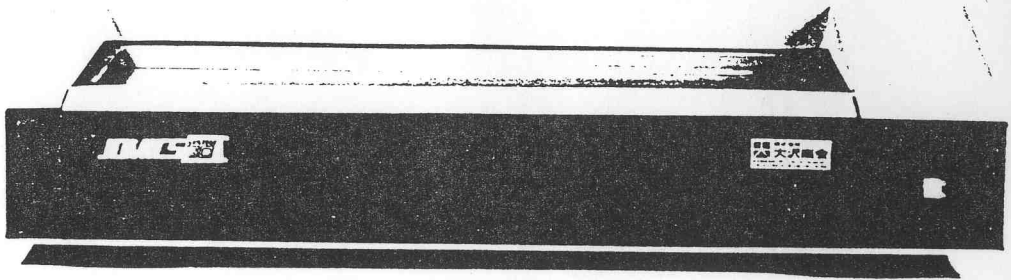


Fig. 1: Mechanical Peek-a-Boo, Used for 15 Years

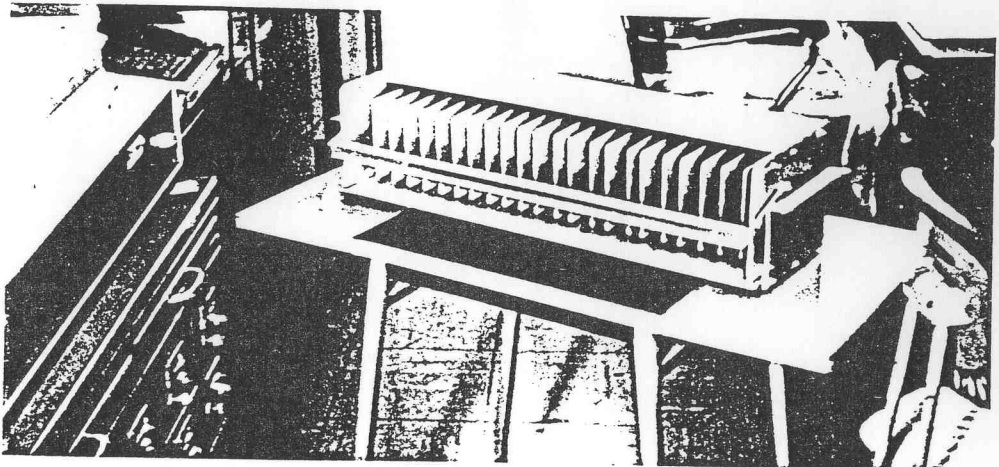


Fig. 2: A Card Tray of the Mechanical Peek-a-Boo

A tray is able to store 1300 cards at maximum.

退院者名カード

氏名	生年月日	退院番号
井上正男	28 5 75	1953
賀田真沙子	20 10 21	772129
西田茂代子	23 6 11	751658
石谷清子	20 10 17	752284
池谷金子	16 2 10	772436
園部春江	21 7 21	752531
石田八郎	1 7 27	2671
伊藤久田部	30 6 17	772286
伊藤まゆ子	1 2 77	286
伊藤久田部	30 6 17	7731
園田松次	33 1 25	78-46
伊藤久田部	30 6 17	78-70
磯部政雄	31 8 17	78-13

Fig. 3: Patient Name Card

白内障

1	374	富沢キ	1 74	1052	376
7	374	副島勝三	74	1148	376
0	374	高藤一雄	74	1171	376
2	374	安田ヨシ江	74	1258	376
4	374	佐藤由花	74	1450	376
4	374	塚本三久	74	1531	376
24	374	林忠雄	74	1575	376
202	374	鈴木才利三	74	1638	376
203	374	渡部せや	74	1632	376
656	374	須藤三と	74	1887	376
464	375	竹下工キ	74	1967	376
420	374	市川末元	74	1901	376
684	374	田中ト人	74	2044	376
854	"	沢田ヨヨ	74	2128	376
1030	"	橋本久保子	74	2640	376
4144	"	五丁茂平	74	2651	376

Fig. 4: Disease Name Card

金子 穂子 51
 32才時より D.M. あり。
 最近 glu 150mg あり
 再教育のため入
 Insulin の使用。
 朝 80 ~ 100mg / 日
 朝の尿糖 (-)
 指示インシュリン

Fig. 5: Discharge Summary Card

(医師別貸出カード) 医師名

月	日	カルテ No	月	日	カルテ
59	2.1	38091	59	12.18	67
"	"	64804	"	12.25	5
"	"	40028	60	1.5	"
"	"	36662	"	"	"
59	5.4	80-446	"	"	"
"	"	79-2493	"	"	"
"	"	80-2286	"	"	"
"	"	80-2920	"	"	"
"	"	11287	"	"	"
"	"	24239	"	"	"
59	12.10	88008	"	"	"

Fig. 6: Lend Out Card

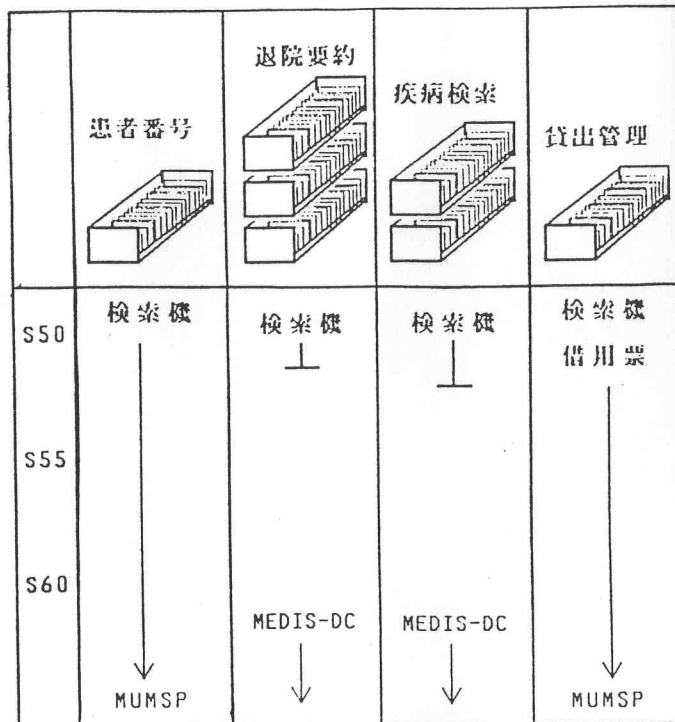


Fig. 7: History of Use of the Mechanical Peek-a-Boo

病歴	1新規	2退院	380X3	480X4	580X5	680X6	780X7	880X8
11	12東	13東二	14東四	21西一	22西二	23西三	24西四	15五病
61E1	62E2	63E3	64E4	71W1	72W2	73W3	74W4	65D5
107病棟	108古賀	109真広	110小林	111佐山	112杉	113新藤	114竹沢	201石井
202								
診察科	科名(カカ)	生年月日	入院日	退院日	病名	病名	病名	病名
36109.01	マタナ	288570.01	林ノヲ	289678.01	ウチノ	111000.01	トク	172291.01
36109.02	玉ダ	580088.01	江ノリ	754718.01	マツダ	3990648.01	ミダト	344188.01
100009.01	スダ	1068989.01	エクラ	11077722.01	イダ	110588.01	オカワ	118788.01
110000.01	アカワ	1139307.01	アリケ	1139958.01	オイ	1144788.01	カカ	1151248.01
110000.02	カタト	152050.01	オセヒ	1152058.03	カカ	1152210.01	アカ	1198888.01
110000.03	玉ダ	155509.01	ハセ	1158111.01	トカ	51158888.01	カ	1157088.01
1157088.01	カウ	1158221.01	アイ	マ				

Fig. 8: Selection Screen (1)

Before pushing the return key.

氏名	住所	年齢	性別	職業	電話番号	備考	備考	備考	備考	備考
35109.01	イサダ ナヒ	-19800219-2-020123-	-11	-	-					
283670.01	オノ ナハイ	-19800714-1-020130-	-11	-	-					
289678.01	ウツ ナツ	-19220828-2-011225-	-11	-	-					
313009.01	ヒロ イキフ	-19710111-1-011214-020108-	-11	-	-					
	カイ カ	-19821116-1-	-020126-	-11	-					
463616.01	ユダ トモ	-19830821-2-	-11	-	-					
590033.01	ルエ リウ	-19490930-2-	-020125-	-11	-					
754718.01	マダ ヲアキ	-19120503-1-	-11	-	-					
996648.01	ミホ トモ	-19780909-1-020123-020129-	-11	-	-					
1044166.01	カダ シウイチ	-19880420-1-	-11	-	-					
1060919.01	スナ ヲシロ	-19881223-1-	-020123-	-11	-					
1068939.01	ユウケル	-19150407-2-	-11	-	-					
1077722.01	イダ ケダ	-19800226-1-	-11	-	-					
1110568.01	オカケイ	-19740920-2-011225-	-11	-	-					
1118735.01	カミル	-19760507-2-020123-	-11	-	-					
1129771.01	アカケイ	-19440124-2-	-11	-	-					
1139307.01	ルリ ケイ	-19410425-2-	-020125-	-11	-					

Fig. 9: Selection Screen (2)
After pushing the return key. Editable state.

氏名	住所	年齢	性別	職業	電話番号	備考	備考	備考	備考	備考
396745.01	カハ ナヒ	-19180724-2-	-020108-	-0	-					
401728.01	スナ シウ	-19301102-2-	-020125-	-21	-					
404932.01	ユキ ヲリ	-19320311-2-	-020122-	-22	-					
436159.01	ウツ ナオ	-19110104-1-	-020120-	-15	-					
436843.01	カイ ナヒ	-19250511-2-	-020101-	-0	-					
	スナ ヲオ	-19141207-1-	-020124-	-12	-					
451433.01	スナ ヲオ	-19141207-1-	-020124-	-12	-					
463616.01	ユダ トモ	-19830821-2-	-11	-	-					
464433.01	ルエ ケイ	-19100302-1-	-020108-	-15	-					
469210.01	トシヲ ミオ	-19150123-1-	-020120-	-0	-					
474926.01	イサダ シウ	-19290217-1-	-020116-	-0	-					
481117.01	イサダ シウ	-19290217-1-	-020116-	-0	-					
490825.01	フクナ リウ	-19551009-2-	-020130-	-15	-					
496269.01	ヒノ テイチ	-19220919-1-	-020120-	-15	-					
497523.01	アノ スイ	-19241129-2-	-020112-	-0	-					
513308.01	カガマ マサ	-19150331-1-	-020125-	-13	-					
518042.01	ミサカ ナツ	-19330802-2-	-020120-	-15	-					
523572.01	ミサカ ナツ	-19330802-2-	-020120-	-15	-					
530848.01	イサダ トオ	-19180324-1-	-020113-	-15	-					
532336.01	カガマ サブ	-19220503-1-	-011216-	-0	-					

Fig. 10: Chart Room Screen
State immediately pushing the '=' key to directly access individuals.

経済性の比較

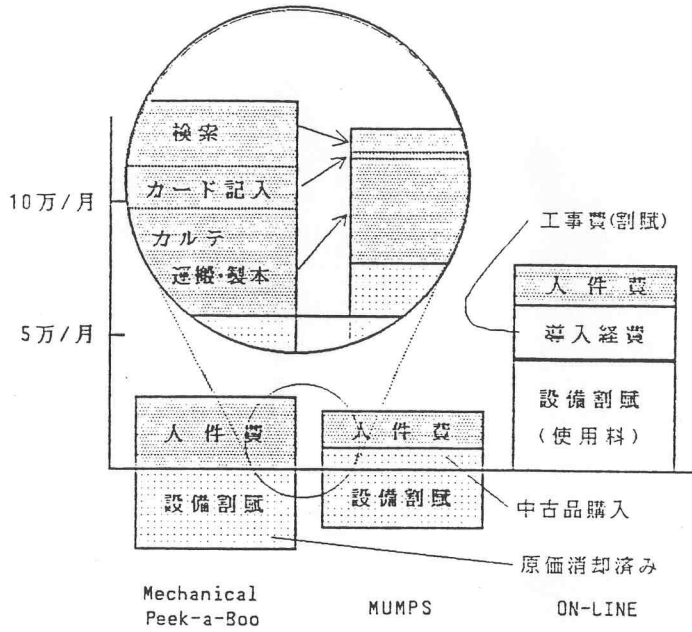


Fig. 11: Cost Comparison

小型文字をスクリーンに書くスモールコンセプトコンセプト

Small Concept to Put Small Characters on Screen

馬場 謙介**, 木村一元**, 杉崎登*

Kensuke Baba**, Kazumoto Kimura**, Noboru Sugizaki*

*国立埼玉病院 臨床研究部, **独協医科大学総合研究施設, *国立埼玉病院 臨床検査科

*Department of Clinical Research, National Hospital of Saitama,

**Institute of Medical Sciences, Dokkyo University School of Medicine and

*Laboratory of Clinical Pathology, National Hospital of Saitama

ABSTRACT

In order to use physically limited screen area more widely, the authors designed the small concept (Do ^%xf1s) to display small characters (composed of 6X8 dots) on a screen. Passing parameters for the small concept are string to be written on the screen (%), display position of the string (%Y and %X) and horizontal width of the writing area on the screen (%W). According to this specification, the authors coded the small concept using \$ZINT function provided by SP-MUMPS (Sumitomo Ltd). The authors estimated that the writing speed of a string composed of every katakana and ANSI printing characters was 0.1125 msec/character under a personal computer PC-9800/RA (NEC Ltd) connecting with a hard disk unit.

はじめに

スクリーンの面積には 限りがある。マンプスの環境下で この限られた面積を有効に利用する 方法の一つとして著者らは、%cell¹⁾ スモールコンセプトコンセプト^{2,3)}を開発し報告した¹⁾。これと、同じ目的で、今回は 小さい文字をスクリーンに表示するスモールコンセプトコンセプトを提案すると同時に、SP-MUMPS (住友電工)に実装したので報告する。

Kensuke Baba, Department of Clinical Research, National Hospital of Saitama, 2-1 Suwa, Wako, Saitama, 351-01 JAPAN (TEL: 0464-62-1101 ext 558, 361 FAX: 0464-64-1138)

当該スモールコンセプトの仕様

本スモールコンセプトは、受渡変数(%)で指定した文字列を 横方向 6/8 ANSI標準文字巾、縦方向 8/16 ANSI標準文字高の小さい文字でスクリーン装置に表示する。当該スモールコンセプトはルーチン %fls で実現する。表示位置は、%Y, %X で指定する。(%Y, %X の値の単位は character bound の整数または non-bound の小数で指定する)。表示の水平方向の巾は %W で指定する。(%W の値の単位は、半角文字単位で指定する)。表示文字の色は %C で指定する。(%C の値の単位は color code で指定する)。

グラフィックによる実装例

SP-MUMPS \$ZINT 関数で 上記スモールコンセプトを 固定ディスク内蔵の PC-9800/RA システムに実装した。その概要は次の通りである (Fig. 1 参照)。

グローバル ^%xf1 に、生成する文字のベクトルデータを持った。そのデータ形式は：

$$\text{^}\%xf1(\text{character}, n) = x_1, y_1; x_2, y_2$$

である。ここで、 n は 書き順、 x_1, y_1 は 夫々 直線始点の(相対)座標、 x_2, y_2 は 直線終点の(相対)座標である。

グローバル ^%xf1 を読んで、全ての半角文字 (\$C(32)~\$C(128)、\$C(160)~\$C(222))を screen #4 の定位置(文字セットストッカー)に書く。ここで、各文字は、screen #4 で定位置をドット単位で持っている。各文字の screen #4 の定位置(Y,X)と文字のアスキーコード(A)とは、次の簡単な関係で定義されている。

$$\begin{aligned} X &= A - 32 \# 100 * 6, \\ Y &= A - 32 \# 100 * 8 + 384 \end{aligned}$$

表示画面(普通、screen #1,2,3)に文字を表示するときは、文字セットストッカーからその文字画像を予め確保したワークメモリーに複写し、その複写した(ワークメモリーの内容)文字画像を、表示画面に再複写する。(どの文字を表示する時も、同一のワークメモリーを使用する)。このように、一字一字を 表示画面に表示する機能を subconcept にまとめてある。特に必要ならこれを直接利用することもできる。スモールコンセプト %xf1s は subconcept を使って、指定した文字列を指定した任意の位置と範囲に表示する。

固定ディスク内蔵のPC-9800/RA で、文字セットストッカーに文字を書く速度は、平均 43.75 msec/文字であった。160の全ての半角文字からなる160文字の文字列を文字セットストッカーの文字から呼び出して、表示画面に表示するのに要した時間は、0.1125 msec/文字 であった (Fig 2)。

当該スモールコンセプトには、下記の複数の入口が設けてある。システムの状態に応じてこの入口参照を選択する。

INSTALL	グローバル ^%xf1 を生成する
INIT	全半角印刷文字を表示画面に隠し書きする
OPEN	ワークメモリーを領域確保する
USE	表示画面の隠し書部分を文字セットストッカーに移す
WRITE	受渡変数の条件に従い、文字列の表示を実行する
CLOSE	ワークメモリーの領域を解除する

普通、グローバル ^%xf1 の生成は完了しているので、ラベル INIT をまず実行する。INIT を実行すると、\$ZINT(#43,"GINIT") 等が自動的に実行される。ユーザーは、いかなる \$ZINT関数も意識しないで済む。ラベル OPEN から実行を開始すると(実際には、ワークメモリーの領域確保を実行するが)、あたかも小型文字を発生する装置をオープンする感覚で動く。USE によって全ての準備が完了する。諸パラメータを指定して、DO WRITE ^%xf1s を実行すると、目的の位置と範囲に、目的の文字列が指定した色で表示される。Fig. 3 に、表示される小型文字を例示した。

考 案

文字の表示速度は、Fig. 2 に示した如く、テキスト画面への書き出しより2倍以上速い。従って、高速、大量文字の表示が同時に実現できる。通常の 400×640 ドットのスクリーンをフルに使うと、50桁×106桁の文字が表示できる勘定になる。

このスモールコンセプトを用いると、入力画面の項目見出しに費やされる面積を節約できる。一画面に245個の入力項目がある「予後解析用データベース処理システム」に応用した⁴⁾。このシステムは、Xecute ^%e を多用して、入力画面がそのまま、編集画面であり、表示画面になっている。一度にその症例の全容が把握できるので、予後解析の見当をつけるのに威力を発揮している。

イタリック、拡大文字などの表示は、マンプスの夢であった。夢を現実のものとして使えるように、このスモールコンセプトのフォントデータは書き順毎の直線の始点終点で記述しておいた。

むすび

実装例では、依存の関数(\$ZINT)を使っているが、このことは、このスモールコンセプトの提案を阻害するものではない。そもそも、スモールコンセプトは、出入口と機能を厳格に定義し、その実現方法は規定していない。スモールコンセプトの作成者は、マンプス装備者でもユーザーでもかまわない。今回の場合、マンプス装備者が(主に別の目的で用意した)\$ZINT関数をユーザーが別の目的に利用して、マンプスの機能を拡張した。他のマンプスにも実装されることを期待する。

謝 辞

実装に当り、貴重な助言と技術指導を賜った住友電工の煙山氏に深謝する。

文 献

- 1) Baba K: Linked cell. *Mumps* 14(suppl): 118, 1988 (in Japanese)
- 2) Baba K: On small concept. *Mumps* 14(suppl): 73-76, 1988
- 3) Baba K: Small concept. *Mumps* 14: 33-43, 1988.
- 4) 馬場謙介ら: 予後解析用汎用データベース処理システム. 第43回総合医学会総会, 仙台, 1989.10.26-27. *医療* 43-I: 556(24-11)

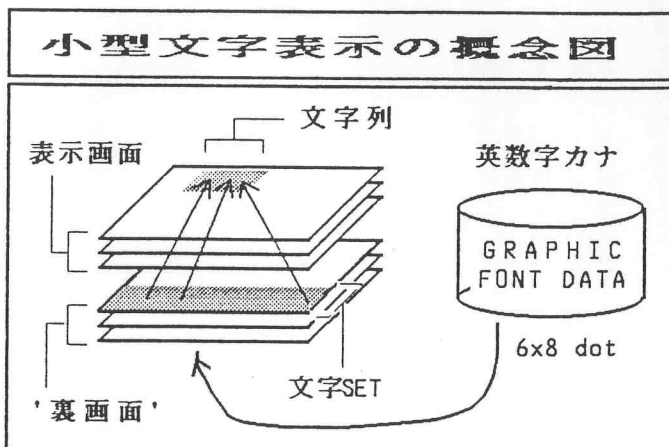


Fig. 1: Procedure Schema of the Present Study for the Small Concept

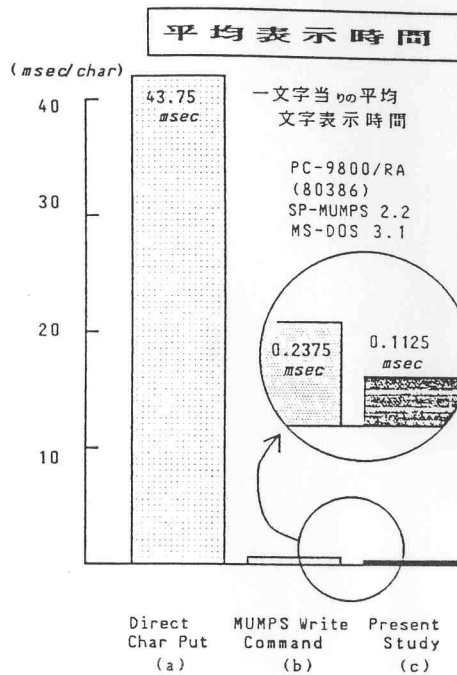


Fig. 2: Comparison of Time Duration to Write a Character on the Screen

- (a): Direct drawing (writing) of a string composed of 160 characters (TESTac).
 (b): Ordinary writing of same string by MUMPS WRITE command.
 (c): Present study: Indirect drawing (writing) of same string (see TESTac).

```

TESTac ;
W !,$H ;
D INIT ;
W $C(J) ;
W !,$H,!,"*****" ;
D OPEN ;
S %USE=0 ;
D USE ;
S %= ""
F J=32:1:128,160:1:224 S %= $C(J) ;
S %X=10 ;
S %Y=5 ;
S %W=10 ;
W !,$H ;
F JJJ=1:1:1000 D WRITE ;
W !,"/",JJJ,"/",$(STRING) ;
W !,$H ;
D CLOSE ;
Q ;

TESTb ;
W *26,!!!!!,$H ;
F I=1:1:1000 W *30 F J=32:1:128,160:1:224
W *30,!!!!!,$H ;
D CLOSE ;
Q ;

```

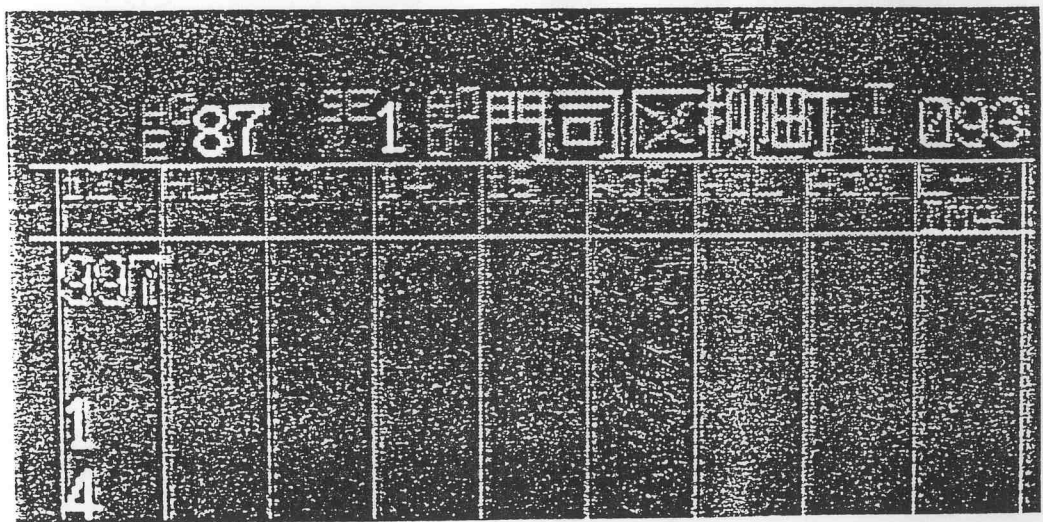


Fig. 3: Example of Displayed Small Characters
 Larger characters on the figure are ordinary ANCI characters.

地域がん登録における M U M P S の応用

An application of MUMPS in
the population-based cancer registration system

林 恭平

Kyohei Hayashi

京都府医師会・がん登録委員会、京都府立医科大学・公衆衛生
Kyoto medical association・cancer registration committee,
Kyoto prefectural university of medicine・preventive medicine

[ABSTRACT]

In Kyoto, the population-based cancer registration system for all sites of cancer has been introduced by Kyoto medical association since 1982. The average of registration per year was about 4,500 and now total registration number is above 37,000. All data for registration has been processed by a micro-computer which the system programming language was changed from BASIC to MUMPS in 1987. The present hardware is a 16-bit CPU with a 40 MB hard disk, which is standard and low cost, but the function, for instance identification of cancer patients, is sufficient for cancer registration.

[キーワード]

癌登録、MUMPS、マイクロコンピュータ
Cancer registration, MUMPS, micro-computer

[目的]

癌登録は癌の発生状況を把握する事により、癌の対策および予防を行うことを可能にする重要な作業であるが、そのためには癌登録が正確に行われなければならない。この癌登録を正確かつ効率的に行うためにマイクロコンピュータ上でのMUMPSにより癌登録システム構築し、現在約37,000件の登録を行っているが、このシステムは癌登録システムとしての基本機能は十分に果たしていると思われるのでこのシステムについて紹介報告する。

[経過]

京都府では1982年から地域癌登録事業を始め、その当初からその情報処理はマイクロコンピュータで行ってきた。登録開始当時のマイクロコンピュータは、8ビットであったが癌登録を正確に行うためには漢字を扱うことを必要条件として漢字処理が行えるSORD社の漢字BASICでシステムを構築した。しかし登録件数は年間平均4500件程であり、その件数が累積されることによりシステムで処理できる補助記憶装置の容量の限界と多量の漢字情報を入力するには8ビットでは漢字入力に負担がかかりすぎる等の問題により、5年目の1987年度から現在の16ビットのマイクロコンピュータでMS-DOSのもとでのMUMPSによるシステムに移行した。

〒602 京都市上京区河原町広小路上 京都府立医科大学 公衆衛生学教室

[システム構成]

ハードウェアの基本構成としてCPUはNEC PC9800VX21（増設メモリ付き）、補助記憶装置はフロッピーディスク（5インチ × 2）、ハードディスク（20MB × 2に分割）、及びプリンターである。ソフトウェアはMS-DOS上のでucdマイクロMUMPS V. 4. 6をプログラム言語としている。

[内容]

本癌登録システムは、MS-DOS上のMUMPSでの一つのアプリケーションとして位置づけられている。すなわち入力されたデータはMUMPSだけでなく、他のMS-DOS上のプログラムでも利用できる。

入力される基本的最小情報は、届けられた癌患者の姓名、生年月日、住所等の個人識別情報と、個々の癌患者の癌の部位、診断日、病巣のひろがり、診断方法、治療方法、転帰等の癌に関する医学疫学的な情報とである。その入力された情報から計算機に作られる登録情報としてのグローバルファイルは登録患者の個人識別情報からなる患者台帳ファイルとその届け出患者の癌登録用紙に記載された情報のデータファイルでどちらも各登録年度ごとに作成される。

現在癌登録システムのために用意されているプログラムの種類は図-1に示されている。これはMUMPSから癌登録システムに入った時の初期メニューの内容である。

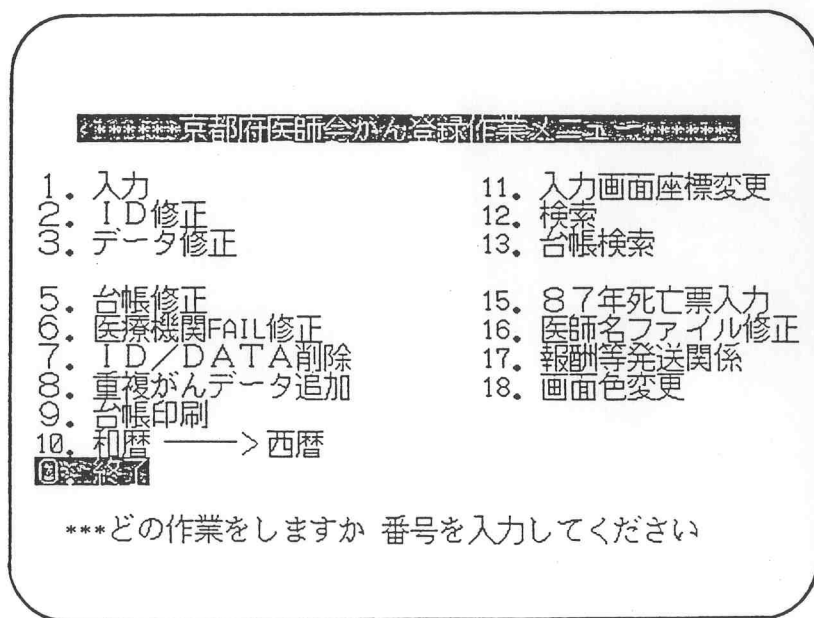


図-1 初期メニュー画面

即ち、この内容は

- A) 登録情報の初期入力と入力された情報の修正、追加
- B) 入力作業を正確かつ効率的に行うための支援情報の入力、修正項目
- C) 登録された情報から必要な情報を検索、集計する

D) その他

に分類することができる。

それぞれの代表的項目について以下に説明する（作業過程のディスプレイ画面は実際の入力中の画面であるが、表示の一部は削られている）。

1) 登録情報の入力（メニュー番号1、8、15）

かん登録1989年度分を入力します

1. 番号 8000	2. 受付年 1989	3. 月 3	4. 日 9	5. セイ ヲカ	6. メイ ミル
7. 姓 田中	8. 名 みのる	9. 生年 1933	10. 月 1	11. 日 1	12. 性 2
13. 住所 3 左京区	14. 機関 223 京都第二赤十字病/4/2				
15. 科番号 1	16. 医師姓 安	17. 名 健 朗	18. 胃届 0		
19. 診断年 1989	20. 月 20	21. 確疑	22. ICD O	23. ICD 9	
24. 組織型	25. 単発重複	26. ひろがり	27. 診断方法		
28. 手術方法	29. 治療方法	30. 転医	31. 検診		
32. 転帰	33. 死亡年	34. 月	35. 日	36. 死因	

B(A)を入力すると前の項目に戻ります

図-2 届出情報入力画面

図-2はメニュー番号1の届けられた登録情報を入力中のディスプレイの画面である。この画面の様な状態に入るためには、入力前半の個人識別情報の入力の過程で、この届けられた情報を新規に登録すべきかどうかのチェックを経なければならない。

このチェック過程が本システムの重要な部分になるが、本登録システムでは

- 1) 個人の姓名の読み方（セイ、メイ）、
- 2) 漢字による個人の姓名（姓、名）
- 3) 個人の生年月日の3種類の項目についてチェックすることになっており、それぞれの項目の検索ファイルによってその項目の情報が入力される度に既に登録されている患者かどうかチェックされる。

図-3は患者の姓名の読みだけを入れた時点で、チェックとして同じ読み方のすでに登録されている登録者の一覧が表示される様子を示したものである。同一人と思われる人物がいればそれを確認するステップに画面が変わり、該当者がなければ図-2の画面に戻り次に漢字の姓名を入れることができる。漢字の姓名を入力した時点、およびさらに生年月日を入れた時点でそれぞれこの図-3と同じような入力された情報から該当者がいるかないかチェックする画面が現れるステップがある。

未登録とされた届出患者に対しては届出用紙に記入された情報を入力していくが入力すべき情報は届出用紙に記入された数字そのものがほとんどであるが、漢字

検索、集計プログラムは現在も開発中であるが、MUMPSでのシステムであるのでその都度の条件にあった検索をおこなうプログラムをつくることも簡単である。現在用意してあるのは個人を検索する（メニュー番号12）ためのものと、登録情報を組み合わせた検索おこなう台帳検索（メニュー番号13）プログラムが用意されている。図-4は後者の検索プログラムで検索情報と検索条件を選択する画面の例である。表示画面から判るように検索条件の表現はMUMPSのプログラムで使用する表現と同じものを使っている。

さらに各種の統計をとるためのプログラムを用意してあるが、癌登録の管理者の権限として利用するために一般のメニューからは隠してある。

任意の項目指定による検索を行います

1. 番号 : 無指定	2. 受付年 : 無指定
3. 月 : 無指定	4. 日 : 無指定
5. セイ : 無指定	6. メイ : 無指定
7. 姓 : 無指定	8. 名 : 無指定
9. 生年 : 無指定	10. 日 : 無指定
11. 日 : 無指定	12. 性 : 無指定
13. 住所 : 無指定	14. 機関 : 無指定
15. 科番号 : 無指定	16. 医師姓 : 無指定
17. 科名 : 無指定	18. 胃届 : 無指定
19. ID番号 : 無指定	

1:X= 2:X> 3:X< 4:<X< 5:X[6:X'[7:ELSE 8:中止 9:無指定

1.番号:無指定 の指定内容を変更します 条件を選択してください

図-4 検索条件入力画面

5) その他

その他のメニューとしては台帳の印刷や癌登録の事務的な処理を行なうためのプログラム等も用意してあるが、特徴的なプログラムとしては画面色変更（メニュー番号18）で、ディスプレイの色を黒色、白色、青色の三種の中から入力者が一番都合の良い色を選択できるもので、入力状況（画面）を改善することによって入力速度と正確さを向上させる目的がある。

[考察]

地域癌登録は対象とした地域での癌患者を全員登録し、その届けられた患者がどうなったかを把握することが目的である。従って癌登録のシステムに求められることは届出情報が正確で、登録洩れがないことであるが、そのために多量の届出情報を収集する方法・手段を確立することが重要になり、計算機がそのシステムに対して担う役

割は多量の情報の入力を容易にし早くすること、登録の重複をなくし、正確に登録されることを支援することである。

本システムで登録の重複を避けるために利用している届出情報は既に述べたように届出患者の名字と名前、及びその生年月日であるが、実際は次のような処理を行っている。名字と名前の読み（セイ、メイ）は、漢字で書けば全く違う字でも読みが同じであったり、患者本人の読みと届出者（または登録者）の読みが異なる場合があるので「読み」の情報は同定を確定するには不完全な情報であり、直接漢字を利用すれば「読み」の情報は必要ないと考えられる。しかし例えば「さいとう」という名字は漢字で「斎藤」と「齊藤」とでは違う人物と判定されるので漢字のみでは同定出来ないことがある。または届け出患者を「五十音順」に並べるなどの検索作業には「読み」の情報を使用するので読みの入力が必要になる。

以上のことから本システムではセイ、メイを添字とするグローバルファイルと漢字を添字とするグローバルファイルとがインデックスファイルとし用意され、届出患者の同定時の検索に使われる。新たに情報が入力されればそのファイルはそれぞれ更新される。さらに生年月日でチェックが必要なのは、例えば「斎藤幸子（サイトリキコ）」を「齊藤幸子（サイトリチコ）」として読みも、漢字も異なって入力された時で、同じ生年月日のなかで候補を表示して確認する。そのために西暦の生年×10000+月×100+日を添字とするインデックスファイルが用意されている。

このように3つのファイルによって各情報が入力された時点で即座に条件にかなったグローバルファイルが検索されるが、MUMPSで行うことで簡単なステップのプログラムで現在の40、000人程度では実用に耐える速度で処理できている。

この3つの情報でまだ同一人かどうか判らない時には住所の情報を利用している。ただこの情報はこれまでは入力量が多い為に計算機には簡単なコード入力をし、必要な時は届け出用紙に戻って確認していたが、今後住所情報も市販プログラムを利用することで町名まで容易に入力出来るので、計算機上で確認出来るようになるであろう。さらに厳密な同定を必要とするなら画像情報を利用する事も考えられる。

入力作業の簡便化と迅速化の配慮の一つとして、頻繁に同じ情報を入力をしなければならぬ医療機関名や届出医師の名前などの情報のグローバルファイルを作り、入力時にそれらを参照することにより入力の省力を計っている。またその内容を修正するプログラムも用意しているが、今後さらに洗練された支援が出来るように開発をしなければならぬ余地を残している。

多量の情報を入力するためには複数の入力者で同時に入力する事が必要になってくるので、そのためにはマイクロコンピュータで稼働するマルチユーザーのMUMPSの利用を考えているが同定作業時の検索速度が、現在のシングルユーザーでの速度との比較が問題になる。

[おわりに]

このシステム開発は京都府医師会がん登録委員会の各委員および登録室の実務を担当している田中義久氏の努力によることが大なるものであることを附記しておく。

患者さん自身が外来受け付けで操作するパソコン外来予約
A System for Making Appointment in the Outpatient Clinic
which is done by the patients themselves

○林寺 忠* 西小森 隆太* 廣田 陽代* 西角 淳* 東 洋子**
Tadashi Hayashidera, MD*, Ryuta Nishikomori, MD*, Haruyo Hirota, MD*,
Atushi Nishikado, MD*, and Yohko Azuma, RN**
国立京都病院小児科* 国立京都病院看護部**
Department of Pediatrics* Department of Nursing**
Kyoto Natinal Hospital

Summary

An NEC-9801E microcomputer was used for making appointment in the outpatient clinic which was manipulated at the clinic by the patients themselves. All routines were developed utilizing SP-MUMPS and touch panel was applied for I/O interface. It was introduced successfully in the pediatric outpatient clinic, started on October 4, 1989.

Key words: パソコン、タッチパネル、SP-MUMPS、外来予約
microcomputer, touch panel, SP-MUMPS, appointment of the clinic

【はじめに】

国立京都病院の1日外来患者数は全科あわせて約1,000人である。小児科の1日外来患者数はこのうち50人である。午前中にすべての患者の外来カルテを時間内に取り出して準備することが出来ない。従って、予約制度を従来から採用し、予約の患者の外来カルテを前日の午後を用意して、外来の混雑を緩和していた。小児科では予約には予め小児科外来にそなえつけた予約簿に患者IDと氏名を記入してもらっていた。一方、予約していない患者の場合は番号札を用意して順番に診察がなされるようにしていた。この2つの作業をパソコンを使って実行しようとした。大切なことは国立京都病院では職員数が「行政改革」の錦の御旗のもとに増員されないからこの作業で1人たりともよけいな仕事をする余裕がないこと。つまり、パソコンを運用するための人員を用意したり、パソコンのために新しい診察券（例えばキャッシュカードのようなコンピュータで読み取りを容易にするインターフェース）を準備したりする人員を用意することは出来ないのである。

【使用機種とソフト】

PC-9801E、ディスプレイ（PC-KD551）、プリンター2台（PC-PR6021、PC-PR201）、
タッチパネル（PC-9873）、5インチフロッピーディスクドライブ、
ハードディスクドライブ1台、キャスターつきの移動可能な装置収納庫
プリンター切り替え装置（Switch cable SC-2P）
MS-DOS Ver.3.30、SP-MUMPS Ver.1.2、PC-PR6021プリンタードライバー（PR1002.SYS）、
タッチパネルルーチン（SPTOUCH.COM）、GDC操作グラフィックルーチン（ZGRAPHX.COM）

【作業を実用化するために解決すべき事項】

☆ソフト開発についての問題点

1. 外来患者、小児科ではその母親に入力してもらう。キーボード入力は不適當であるため、タッチパネル（タッチスクリーン）採用

〒612 京都市伏見区深草向畑町1-1 国立京都病院 小児科
Department of Pediatrics, Kyoto Natinal Hospital, Mukaihata-cho 1-1,
Fushimiku, Kyoto City 612, Japan

2. 画面の表示文字は大きい文字を描く必要がある。ZGRAPH.COM の作成。これを SP-MUMPS に組込む。これは ZGRAPH.COM の文字表示を改良したもので、JIS コードを入れるかわりに文字列そのものを \$ZINT(...) の argument に入れる。
3. タッチパネルを使うソフトの開発。SPTOUCH.COM の作成。これを SP-MUMPS に組込む。
4. ID 番号の入力。国立京都病院の ID 番号は例えば、12-3456-7 の形式である。電卓の画面を作成する。リターン（終了）を押すようにするかどうか。ID 番号のチェックはチェックディジットとする。ID 番号の”-”を入れさせるかどうかについては入れても入れなくてもよいようにする。
5. 担当医を選ばせるか、診察の曜日を選ばせるかの決定。
担当医を選ばせると、心臓外来などの特殊外来に一般患者が混入するのを防ぐ（ウイルス防止）ことが出来る利点があるが、こうすると特殊外来が無くなったりしたとき、プログラムを大幅に変更しなければならぬためこれを断念した。曜日を選ばせると、患者の母親には分かりやすくなる。
6. 日付の指定はどうするか。
カレンダーの作成をして、その日を選ばせる。日曜日、祝祭日、休診日、代診日の場合どうするか。また、急に担当医に人事異動があった場合にも直ちに対処する。当日とそれより7日後までは予約できなくする。これは病院の医事課の希望である。休日の変更があった場合どうするか。これらをいつでも必要に応じて変更可能にする。
7. カレンダーは万年カレンダーを作成するが、3ヵ月の間だけ予約可能にする。それ以外は予約が出来ぬようにする。その期間2回までの予約は出来るが、3回以上は予約出来ない。その月の15日を過ぎると4ヵ月予約可能とする。
8. 診察時間帯を3つに分け、9時、10時、11時とする。各時間帯に予約可能な患者数の制限を自由に設定できること。
9. ID 番号を入れると患者名が表示されるようにする。未登録（新しい）ID 番号を拾いだすルーチン作成。登録する時に、カタカナ氏名だけでも記録する。
10. 予約表の印刷についての工夫。
ID 番号順に打ちだすと医事課に便利である（カルテの検索が容易になる）が、これはまだ検討中である。
11. ディスクが予約でいっぱいになった時どうするか。
ある一定の期間までの予約をマンブスのグローバル領域（ディスク領域）から他のディスクにはきだす。ハードディスクの採用は、領域を広くとれてスピードアップにつながるが、バックアップしてデータの保護に配慮しなければならない。
12. プリンタードライバの作成。
PC-6001仕様のプリンター PC-PR6021 を PC-9801 で操作するプリンタードライバを作成する。

☆ハードについての問題点

1. ハードが故障した時どうするか。
パソコン、CRT、タッチパネル、プリンターを2台用意しておく。
2. データが破壊されるといった事故にはどうするか。
バックアップディスクを作成する。これもバッチファイルを利用して外来の看護婦によって毎日5時に実施してもらう。
3. 子供の手が届かないようにする。... 一定の高さにタッチパネルを引上げておく。
4. キーボードやコンピュータを外来の患者の手に触れないようにする工夫。
ロッカー型式の収納庫を用意して、CRT（タッチパネル）とプリンターのみをその上に置くようにする。この収納庫を移動出来るようにキャスターをつける。
5. 電源の準備。コンセントの新設を補給係に依頼した。
6. 設置場所を外来の入り口にし、午後もそのまま稼働させ、その後診察室に収納する。

以上の問題点はまず全て解決できたと考えている。

まだ解決できていない問題について

1. 停電の時にどうするか。．．．大混乱に陥るとされる。
特に、ハードディスク稼働時には一定時間経てばヘッドが SHIPPING ゾーンにリトラクトするようにするソフトを作成し、一部は解決した。しかし、病院内で長時間の停電は無く、絶対に停電は無いという前提で実用化する。
2. 誰もどうして動かしていか分からない時には、直ちに私が参上します。
3. 目の見えない人や、糖尿病の人、老人にこのタッチパネル使用は困難なこともある。
幸いに小児科では若い母親が絶対的に多いことからこの問題は無視する。
4. 万が一もし、システムが本当にダウンした時（動かなくなった時）どうするか。
すぐに前の予約簿記入式にもどれるかがポイントとなるが、この問題のみ解決した。

【システムの運用の概要】

まず、ディスプレイの画面（タッチパネル）に指を触れると作業が開始される。この時、画面に番号札・予約・予約取消のいずれかを選択する。予約をしていない患者さんは番号札の文字のところに触れるとプリンター PC-PR6021 から発行時刻と診察順位、担当医名が記入された番号札が発行される。予約したい患者は予約の項目を選んで画面に指を触れる。この2つの画面を下に示す。左が最初の画面で、画面にふれると右が表示される。

<p style="text-align: center; font-size: 1.2em;">画面に指で触れて下さい。</p> <p style="text-align: center; font-size: 1.2em;">予約出来ます。 外来の番号札を渡します。</p>	<p style="text-align: center;">下の項目の1つを選んで下さい。</p> <table border="1" style="margin: auto; border-collapse: collapse; text-align: center;"> <tr><td style="padding: 5px;">番号札</td></tr> <tr><td style="padding: 5px;">予約する</td></tr> <tr><td style="padding: 5px;">予約取消</td></tr> </table>	番号札	予約する	予約取消
番号札				
予約する				
予約取消				

画面に電卓形式の数字の枠が表示され、ここで自分のID番号を入力する。この時ID番号が正しくない時には再度入力をうながす。このID番号が正しくない限り次の画面には進まないが、正しいID番号が入力されると下右のように患者名を画面に表示する。このことで、自分の名前が記憶されていることを患者さんが知ることになる。

<p>診察券の番号は？</p> <p style="text-align: center; font-size: 1.2em;">12-3456-5</p> <table border="1" style="margin: auto; border-collapse: collapse; text-align: center;"> <tr><td style="padding: 5px;">7</td><td style="padding: 5px;">8</td><td style="padding: 5px;">9</td><td style="padding: 5px;">-</td></tr> <tr><td style="padding: 5px;">4</td><td style="padding: 5px;">5</td><td style="padding: 5px;">6</td><td style="padding: 5px;">?</td></tr> <tr><td style="padding: 5px;">1</td><td style="padding: 5px;">2</td><td style="padding: 5px;">3</td><td style="padding: 5px;">前</td></tr> <tr><td style="padding: 5px;">0</td><td style="padding: 5px;">消</td><td style="padding: 5px;">去</td><td style="padding: 5px;">終</td></tr> </table> <p style="font-size: 1.2em; margin-top: 10px;">中 断</p>	7	8	9	-	4	5	6	?	1	2	3	前	0	消	去	終	<p>診察券の番号は？</p> <p style="text-align: center; font-size: 1.2em;">山田 太郎 12-3456-5</p> <table border="1" style="margin: auto; border-collapse: collapse; text-align: center;"> <tr><td style="padding: 5px;">7</td><td style="padding: 5px;">8</td><td style="padding: 5px;">9</td><td style="padding: 5px;">-</td></tr> <tr><td style="padding: 5px;">4</td><td style="padding: 5px;">5</td><td style="padding: 5px;">6</td><td style="padding: 5px;">?</td></tr> <tr><td style="padding: 5px;">1</td><td style="padding: 5px;">2</td><td style="padding: 5px;">3</td><td style="padding: 5px;">前</td></tr> <tr><td style="padding: 5px;">0</td><td style="padding: 5px;">消</td><td style="padding: 5px;">去</td><td style="padding: 5px;">終</td></tr> </table> <p style="font-size: 1.2em; margin-top: 10px;">中 断</p>	7	8	9	-	4	5	6	?	1	2	3	前	0	消	去	終
7	8	9	-																														
4	5	6	?																														
1	2	3	前																														
0	消	去	終																														
7	8	9	-																														
4	5	6	?																														
1	2	3	前																														
0	消	去	終																														

ついで画面は下左のように変って、受診予定の曜日を選ぶ。受診曜日を選ぶと下右のよ
うな画面となり、受診予定の月を選ぶようにできる。この時、3ヵ月が表示されるのであ
るが、この操作をしている日が15日（例えば11月15日）を超えると、下右のように
さらに1ヵ月多く、4ヵ月を表示するように設定した。希望の枠に触れて受診予定の月を
決定する。

予約する曜日を選んで、押して下さい。
山田 太郎 12-3456-5

月曜	火曜
水曜	木曜
金曜	土曜

中 断

山田 太郎 12-3456-5

何月に予約しますか？

11	12	1	2
----	----	---	---

中 断

その後、下左に示すように選んだ月のカレンダーが表示される。前の画面で指定した曜
日のみ選択可能となる。その他の曜日は×で囲んで表示するか、赤で日曜日・祝祭日を表
示してその日を選択不可能にする。このようにして受診日を選ぶと、下右に示したように
その日の外来担当医を選択するようきいてくる。

12月 日 月 火 水 木 金 土

					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

月曜日
12-3456-5
山田 太郎

中 断

予約する日を選んで、押して下さい。

担当の医師を選んで下さい。

一 診 +	林 寺
二 診 +	廣 田

中 断

すると画面は変化して、予約時間帯をきいて来る。各時間帯には制限最大予約数をもうけてある。時間帯（9、10、11時）を選ぶと画面は下右のように変って、プリンターからこれらの項目が印刷され、患者はこの出力を持ち帰ることが出来る。

カルテ番号：12-3456-5 山田 太郎 予約年月日：12月 4日 指名の医師：廣田 予約した人数→ 5 4 0 9 10 11 時 中 断 予約する時間帯を入れて下さい。	カルテ番号：12-3456-5 山田 太郎 予約年月日：12月 4日 指名の医師：廣田 予約時間帯：9時 はい いいえ それでよろしいか？
--	---

以上が、具体的な画面の構成である。この他にここでは示さないが、予約取消しや、番号札を同じように画面に指で触れるだけで容易にその作業が出来るようにした。患者さんに直接に画面を触ってもらうことですべての作業が進行するのであるが、いずれもその場で誰か係をおいて説明するようなことは一切しなかった。従って、マニュアルがないソフトである。しかし、全くマニュアルがないかというところではなく、CRTをおいた格納庫の扉に「予約済みの方は番号札は不要です。必ず画面の正面から指で触れて下さい。間違ったり途中で止める時は中断に触れて下さい。」と紙に書いて貼りだした。これが最少のマニュアルであった。

このように患者さんが自分で入力して作業が進むのであるが、午後に患者さんがいなくなった時から職員の作業にはいる。それは、予約された患者の名簿を PC-PR201 より印刷し、医事課に外来カルテを準備してもらうようにした。また、ハードディスクのデータを毎日フロッピーディスク2枚にバックアップする作業もした。

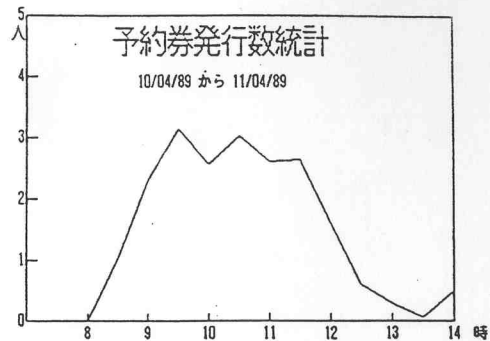
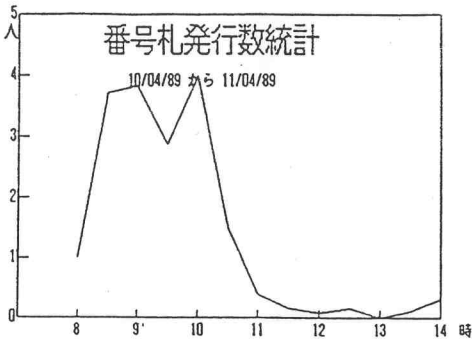
職員がするその他の作業はすべてキーボードを使ってした。①ID番号に対応するカタカナ氏名の無い・誤った患者さんにはその都度氏名を登録・修正する。②休診・診察を記憶させる。などの作業である。しかし、その仕事はいつもあるわけではないので、必要に応じて実施した。

【結果】

平成元年10月4日に実用化に踏み切った。マニュアルがあっただけとはいけませんが、すでに記したように最小限のマニュアルを使用した。扱い方が分からないと訴えて診察室に飛び込んでくる人が出ないか心配したがほとんどそのような人はなかった。たまにあったとしても「とにかく触ってください。」とつぶねることができ、思ったよりスムーズに実用化できた。なかには、「これは小児科だけでやっているのですか？」と言って他の科でも実施して欲しいような口ぶりの人もみられたことは喜ばしいことであった。しかし、「どうしても分からないのでついていて欲しい。」と訴える人もあったが、次からは予約出来そうであった。このような人は極めて稀な存在と言えよう。物珍しさも手伝ってか番号札の必要の無い人まで番号札をもって行ったりする事もあった。ID番号を入力すると患者の氏名がカタカナまたは漢字で表示される場所は「コンピュータはあなたの名前を憶えていますよ。」とのサインを使用する人に伝え、あたたかい配慮ができた。おおむね好評の内に実用化出来たと考えている。

更に、このシステムの運用中に患者さんが作業する時刻がどんな分布になっているかを調べるルーチンを作成したのが下の図である。1カ月の平均取り扱い数をグラフィックにして表示した。下の左は番号札発行件数の時刻別分布であり、右は予約券発行件数の時刻別分布である。興味があることは、外来に来院する患者さんは非常に早く来院する群と締め切り時刻（10時半）ぎりぎりに来院する群に分かれ2峰性の曲線を描いている（下左図）ことである。この曲線を手作業で描く事は人件費の制限から絶対に不可能であろう。

1カ月間に患者さんが操作する件数は番号札発行約 450件、予約券発行約 500件であった。



【考案】

なんの説明もなしにいきなり外来で開始したのであるが、ほとんど大きなトラブルもなく実施できたと考えている。そこで欠点と利点について考察する。

☆このシステムの欠点について

1. 高価である。これを補うために中古のパソコンを購入した。
CPU 7万円、CRT 3.5万円、タッチパネル 2万円、ディスクドライブ 7万円、プリンター 1万円で合計 20万5千円。
2. 電卓型式にID番号をいれさせるには問題がないか。
磁気カードでID番号を読み取らせる方法はエレガントであるが、新たにこのカードを発行しなければならず、そのための装置と人員が必要となり、これは不可能である。
3. 診察番号発行と予約は互いに関係がないから別々のシステムにする方が効率的である。診察番号の所で患者の待ち行列は発生しないか。実際混雑時にはこのことがみられたが周期的にこの待ち行列が見られる程度であった。
4. 時間外の予約は不可能になる。特に夜間に予約に来る人もある。

☆このシステムの利点について

考えて見るとそんなに大きな利点はない。

1. 外来の順番について、番号札に時刻が印刷されて、待ち時間の配慮が少し可能である。新しい患者の来院時刻の把握が出来る。
2. 番号札を返却してもらう必要が無くなった。このシステム導入前の番号札は紛失することが多く、その都度作成をしなければならなかった。
3. 予約した時、パソコンが発行した予約メモ（予約日、予約時間帯、担当医、発行年月日）を患者が持ち帰れる。
4. 番号札発行件数と予約券発行件数の経時的な統計がえられる。

私たち国立病院の「職員の増員」はアイデアがたとえよかってもなかなか実現しないため、「自分たち（職員）は働かず、待ち時間が長いと訴える患者さんを使用する方法」がこれを生み出した原動力になったと言いたい。誠に皮肉なシステムである。

ワークベンチにおけるプログラム作成理論
An application program manufacturing theory in Work Bench

藤江 昭

Akira Fujie

住友電気システムエンジニアリング(株) 応用システム事業部

Applied Systems Department

Sumitomo Electric Systems & Engineering Co., Ltd.

Work Bench is a totalized MUMPS application program developing environment for supporting coding, maintaining and packaging phase. It has following components, as a developing system: screen designer, proto-type & real-type program generator, document writer and impact analyzer, as a running system: multi window emulator and data entry engine.

An application program generated by the Work Bench is a set of software ICs and can be considered that they are arranged on a software base pattern board.

Keywords: 対話型プログラム、ソフトウェアIC、プログラムジェネレータ

Dialogue style application programs, Software IC, Program generator

ワークベンチはMUMPSアプリケーションプログラムの作成/保守/パッケージングを行う総合開発環境であり、大きく開発系と実行系に分かれている。ワークベンチは筆者の展開する対話型プログラム構造論に基づいており、システマチックな開発手順、部品化プログラムによる制御構造、プログラムフレームのジェネレーションと部品化プログラムの埋め込みによるマン・リーダブルなプログラムの生成に特徴がある。

ワークベンチで作成したアプリケーションプログラムは、「ソフトIC」の集合体であり、これらが「ベースパターン」と呼ばれる「ソフトボード(基板)」の上に配置された形になっている(図1)。これらのソフトウェアパーツは、部品番号+オプション番号によってライブラリー管理されており、同時に各病院毎の仕様を管理している「モデルマネージャー」によって、どのパーツがどの病院モデルに使用されているか管理されている。

このため、特定のソフトウェアパーツの変更が、他の既存システムに対してどの範囲まで影響を与えるかを「インパクトアナライザ」により即座に知る事ができ、アプリケーションプログラムの保守作業を効率化する事が出来る。

ワークベンチのソフトICによるプログラミングアプローチは、パッケージング技術の面からみると「オブジェクト指向プログラミング」という事ができる。MUMPSでは、言語の機能上、オブジェクトの十分なカプセル化が困難であり、またオブジェクトプログラムの特徴であるメッセージセレクト機能各オブジェクト内に設ける事は、実行時の処理時間に直接影響する為、オブジェクトの作成に当たっては慎重なトレードオフが必用となる。

これらの事より、ワークベンチでは「Objective-C」の様な完全なオブジェクトを作成する事は行わないが、ベースパターン、ソフトIC等により機能のカプセル化・再利用化を行い、ソフトウェアのオブジェクト化を図っている。

〒107 東京都港区南青山1-15-9 第45興和ビル TEL 03-5474-3910

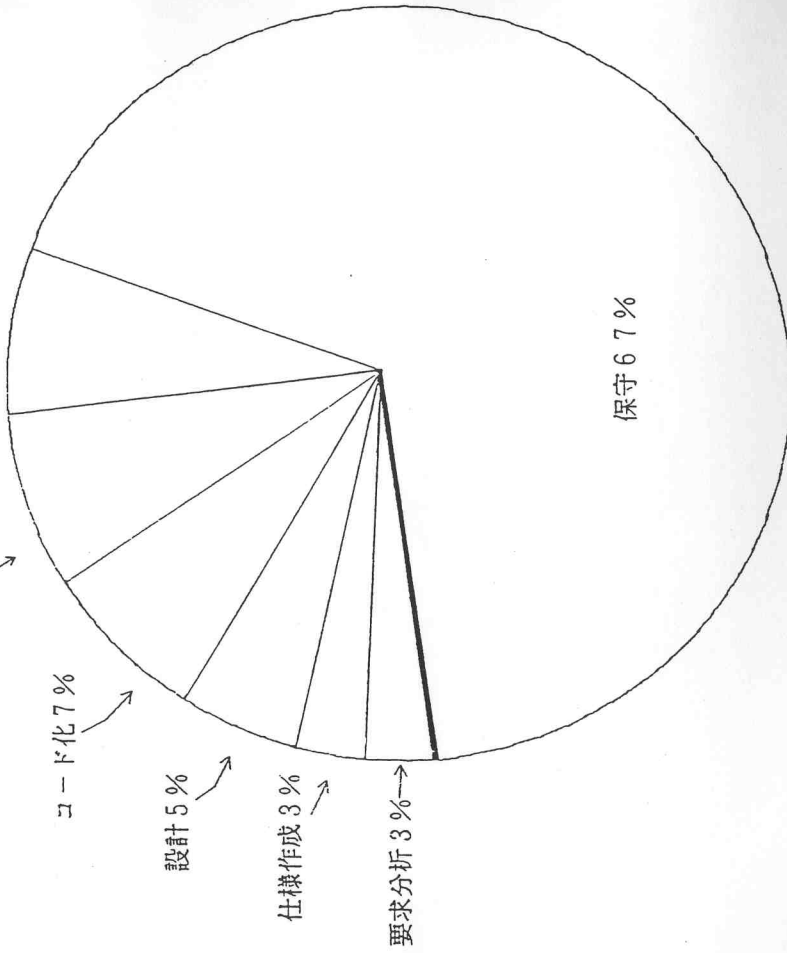
The 45th Kohwa Bldg., 15-9 Minami Aoyama 1-chome, Minato-ku, Tokyo 107

ソフトウェアの各開発段階に要する作業量

統合検査 7%

↓

単体検査 8%

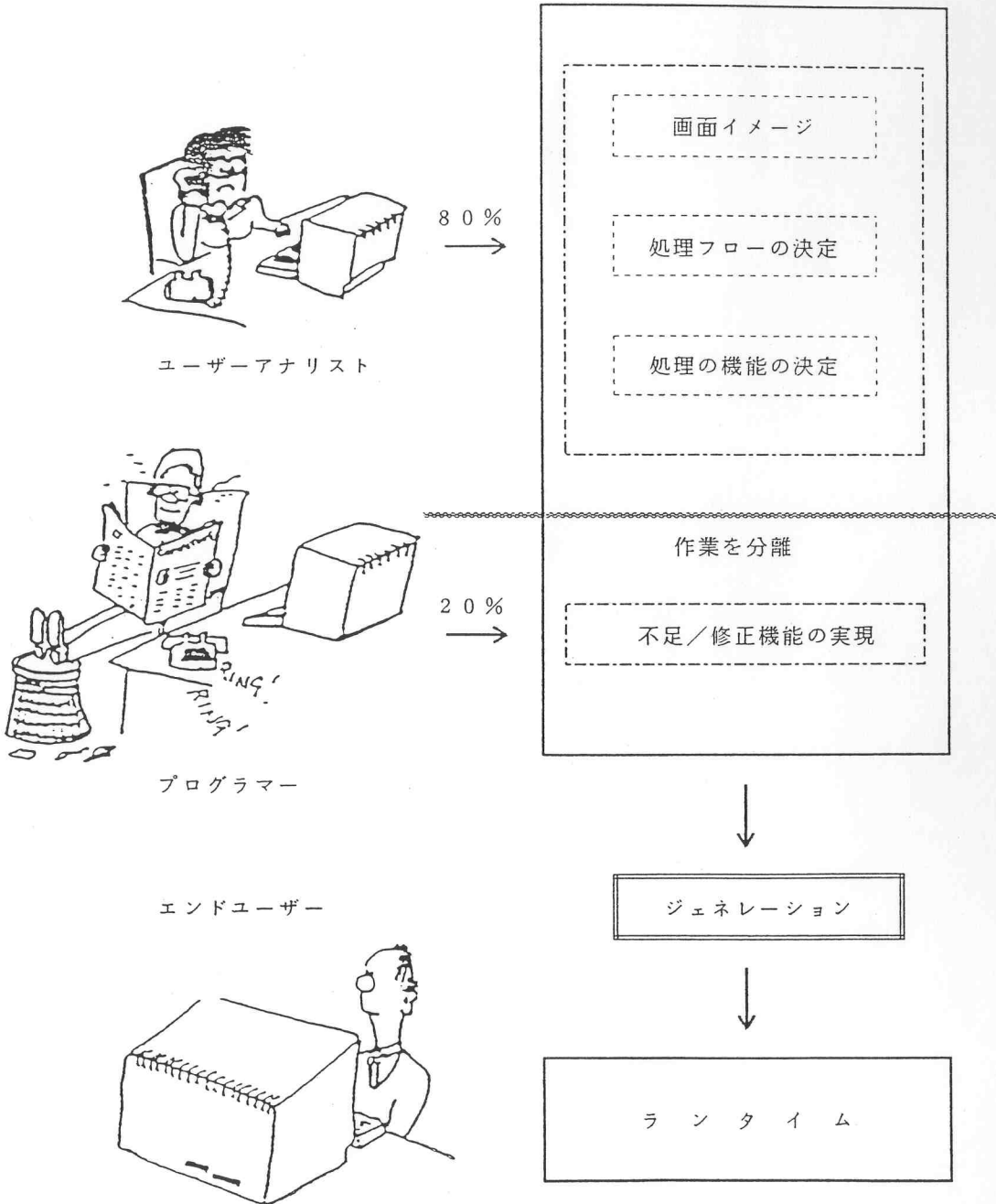


大規模ソフトウェア・システムの真の作業配分

各開発活動の所要作業量 (保守を除く)

《対話型プログラムの開発》

対話型プログラムを作る際に必要な情報量



《ワークベンチの目的》

- ① システムの「仕様決め作業」を「プログラミング作業」から分離
 - ・ S E → エンドユーザー
システムアナリスト
システムインストーラー
 - ・ 非プログラマーであるが業務知識は豊富な側に移動させ
早く正確な仕様作成を行う
- ② プログラム開発過程のシステム化
 - ・ 強制的なソフトウェアの部品化 → 再利用
 - ・ プログラマー間の技量の差の短縮化
 - ・ 製品の品質安定化
 - ・ 基本的なマンマシンのオペレーションの統一化
- ③ 既存リソースの活用
 - ・ 既存ソフトウェアのモジュールレベルでの活用
 - ・ マン・リーダブル・コードの生成によるMUMPSの長所の活用
 - ・ マンプスプログラマーの技術力の活用
- ④ 高機能マンマシンインターフェイスの標準装備
 - ・ マルチウィンドウエミュレータの制御の隠蔽化
 - ・ コーディング上の負荷の削減
- ⑤ 仕様のモデル管理
 - ・ ユーザー毎の仕様の差異をオプションとして管理
 - ・ 「標準病院モデル」及び「個別病院モデル」としてモデル管理
- ⑥ 仕様のオプション管理によるスーパーパッケージング
 - ・ オプション管理によるテーラリング
 - ・ 対象病院固有のシステムであるがパッケージシステム
- ⑦ プログラム保守のシステム化
 - ・ 個別病院モデルによるオプション管理
 - ・ インパクトアナリシスによる影響範囲の分析
 - ・ プログラムアップデートジェネレーション
- ⑧ オートドキュメンテーション
 - ・ 画面設計書
 - ・ 入出力処理／詳細内容（H I P O形式）
 - ・ 入力セルポジション表
 - ・ ベースパターン・セルルーチン表

《ワークベンチの構成》

(開発系) A 1. ハンディー・スクリーンデザイナー

- ・文字画面の定義
- ・入力領域の定義
- ・セル実行順序の定義
- ・プロトタイプジェネレータ

A 2. スクリーンデザイナー

- ・文字画面の定義
- ・マウス有感領域の定義
- ・入力領域の定義
- ・デコード領域の定義
- ・ハードボタンの定義
- ・ソフトボタンの定義

B. ソフトパーツデザイナー

- ・セルの定義
- ・セルルーチンの作成
- ・スタブの定義
- ・スタブルーチンの作成
- ・セル実行順序の定義

C. モデルマネージャー

- ・標準病院モデルオプション管理
- ・個別病院モデルオプション管理
- ・インパクトアナライザー

D. ベースパターンジェネレータ

- ・プロトタイプジェネレータ
- ・リアルタイプジェネレータ
- ・プログラムアップデートジェネレータ
- ・ドキュメントレポーター

E. クオリティーマネージャー

- ・インデックスチェッカー
- ・インデックスリビルダー

(実行系) E. データ入力エンジン

F. マルチウィンドウ・エミュレータ

《ワークベンチで作成するプログラムの特徴》

■ハードウェア基板とソフトウェアボードの対応

- ・機能毎の基板 (xxxボード…)
 - ・基板上の各IC等の部品
 - ・ジャンパー等の基板毎の特殊処置
 - ・I/Oプロセッサ
- = ベースパターン (ソフトボード)
= セルルーチン (ソフトIC)
= スタブ
= エンジン

■ソフトパーツの構成

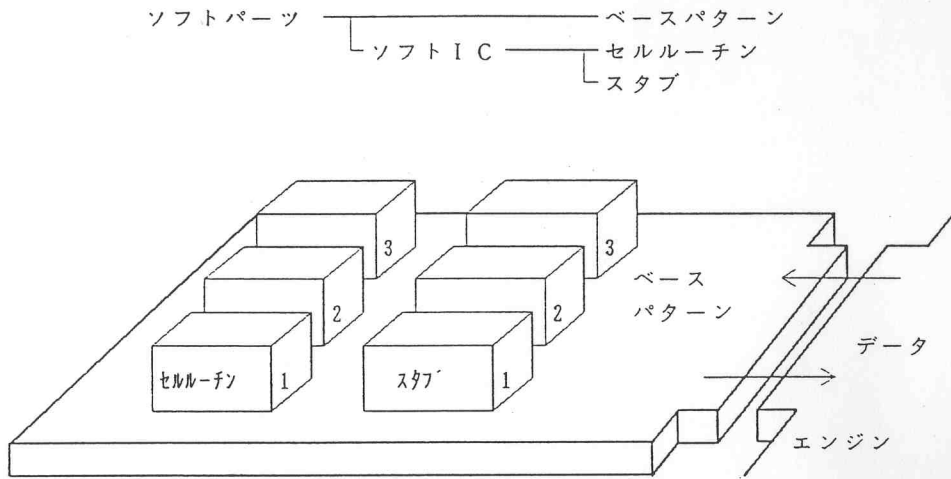


図1 ベースパターン (ソフトボード) 上に配置されたソフトIC群

■取扱い

- ・大きな機能変更 → 基板の交換 (=ベースパターンの交換)
- ・基板内の機能変更 → ICチップの交換 (=セルルーチンの交換)
- ・基板内の特殊処理 → ジャンパーの変更 (=スタブの交換)
- ・パーツの管理 → 型番とカタログで管理 (=セルコードとキーワードによる管理)

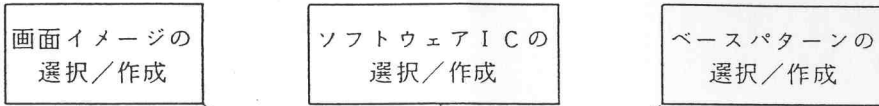
■ソフトボードの駆動系

- ・データベース → ワークベンチ実行系
- ・I/Oプロセッサ →
 - データ入力エンジン
 - マルチウィンドウ・エミュレータ

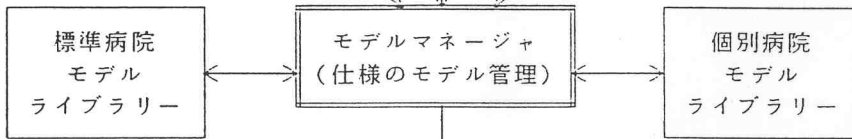
《ワークベンチ開発系での作業フロー》

■仕様定義（CAD）フェーズ

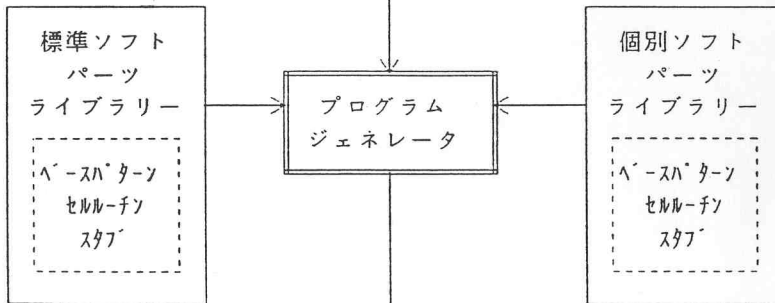
- ①文字画面の定義
- ②マウス有感領域の定義
- ③入力領域の定義
- ④デコード領域の定義
- ⑤ハードボタンの定義
- ⑥ソフトボタンの定義
- ⑦セルの定義
- ⑧セルルーチンの作成
- ⑨スタブの定義
- ⑩スタブルーチンの作成
- ⑪セル実行順序の定義



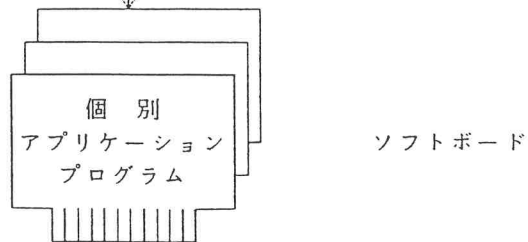
■仕様管理フェーズ



■ジェネレーションフェーズ



■アプリケーションフェーズ



《ワークベンチで作成したアプリケーションプログラムの動作》

■ オンライン対話型処理プログラムの処理の流れ

- ・プログラムの処理は、データ入力、ファンクションキーの押下、マウスのクリック等、何らかのマンマシンアクション毎に進行する
- ・マウス、カーソル移動キー等により入力順序を随時変更する事
- ・ファンクションキー、ヘルプ等、処理の流れと無関係に押下する事

■ ワークベンチで作成した A P の処理の流れ

- ・従来のプログラム → 通常順次実行（上から下へ）
- ・ワークベンチ → 各ソフト I C はデータ入力に対して全て横一列の同一レベルに並んでいる

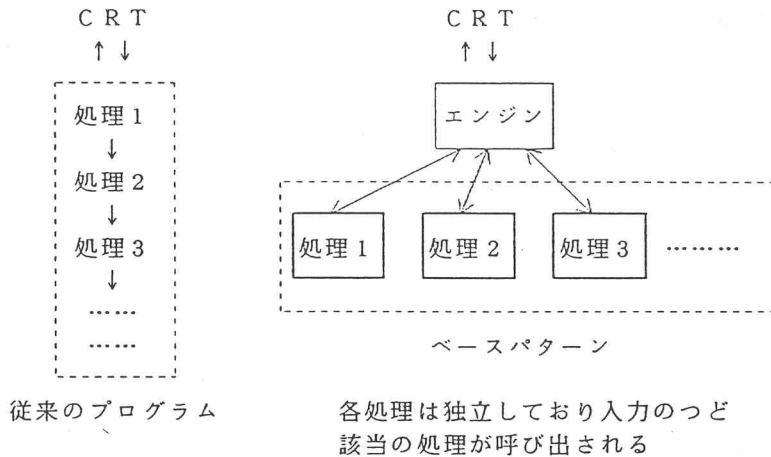


図2 ベースパターン内の各ソフト I C の処理フロー

■ ソフト I C の実行

- ・エンジンが、マンマシンアクションがあるつどイベントドリブン式に実行
- ・ソフト I C 内では、ソフト I C 間の処理フロー制御する為のコーディングは原則として禁止
- ・フロー制御は専用のスタブを用いる
- ・リターンキーによる通常入力時のセルルーチンの実行順序は、事前定義可能でベースパターンが覚えている

《エンジンとアプリケーションプログラムの関係》

■ エンジンの標準サポート機能

- ・ マルチウィンドウ機能
- ・ マウス機能
- ・ ファンクションキー・ヘルプキー等の特殊キーによる割り込み処理機能
- ・ ボタン機能
- ・ 各端末毎のシングルジョブ・マルチタスク機能
- ・ 中座機能

■ エンジンにより得られるメリット

- ・ 端末ハードウェア属性から完全に独立したソフトパーツの作成
- ・ 全てのAPで同一のマンマシン挙動
- ・ マウス、ファンクションキー、ボタン等のソフトウェア割り込み処理が可能
- ・ APプログラマーは、端末制御ではなくAP自体のコーディングに集中可能
- ・ 各パーティション内でマルチタスク処理が可能

■ 制御方法

- ① ベースパターン …… アプリケーションプログラム本体
- ② 画面定義ルーチン (xxxxxxZS)
- ③ エンジン用テーブル(ローカル)設定ルーチン (xxxxxxZC)
- ④ エンジン用テーブル(グローバル)設定ルーチン (xxxxxxZG)

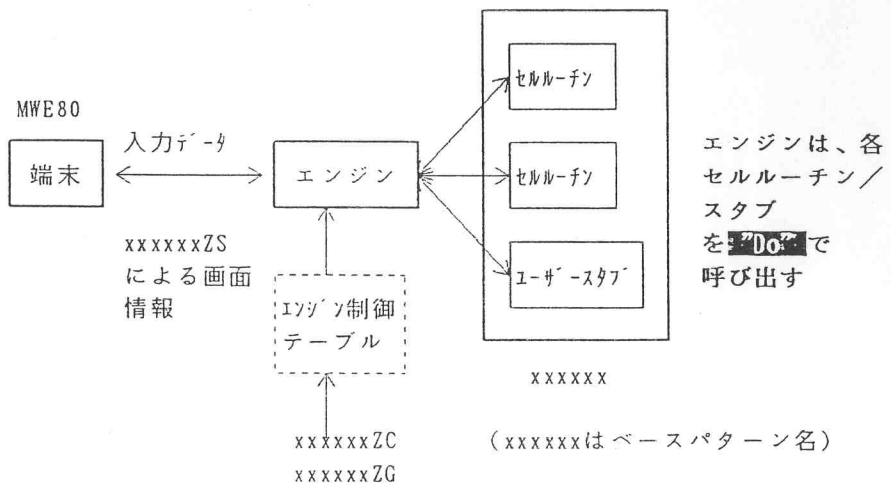
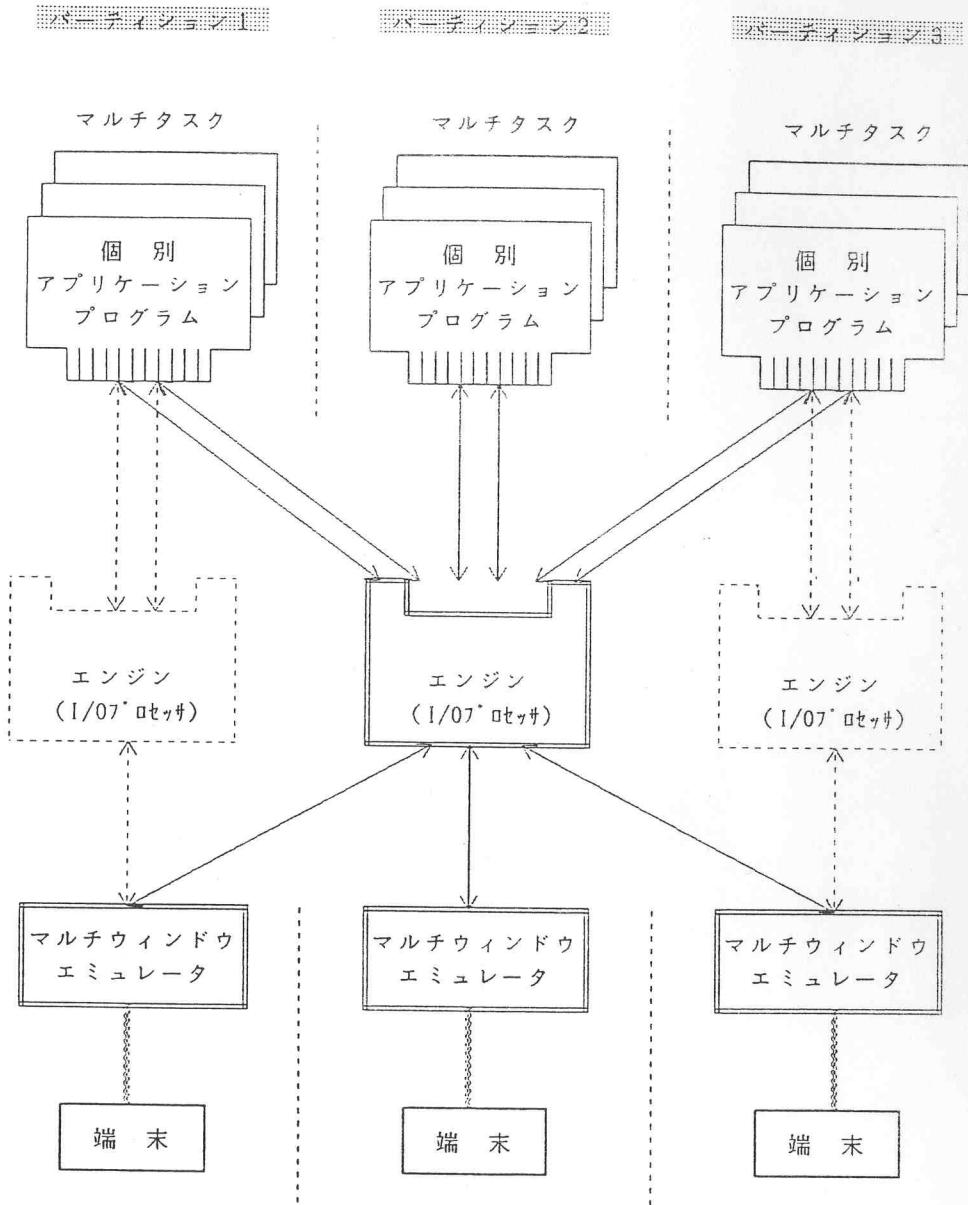


図3 ベースパターン関連プログラムの役割

《ワークベンチ実行系での制御の概略》



《ワークベンチの作業手順とモード》

■ オープニング

テンプレートとなるベースパターンコードの検索

■ モード1：画面編集

文字の定義／修正

■ モード2：領域編集

- a) マウス有感領域の定義／修正
- b) 入力領域／属性の定義／修正
- c) デコード値表示領域の定義／修正
- d) ソフトボタンの領域定義／修正

■ モード3：セル情報編集

- a) セル実行順序の定義
- b) セル情報の定義
 - o セルコードの定義
 - o セルオプション（セル変数、セルルーチン名、デコード変数）の定義
- c) ハードボタンの定義
- d) ユーザスタブオプションの定義
- e) 独立セルルーチンの定義
- f) ライブラリーへの更新

■ モード4：ルーチンの編集

- a) セルルーチンの編集
- b) ヘルプルーチンの編集
- c) ボタンルーチンの編集
- d) スタブルーチンの編集
- e) 独立セルルーチンの編集

■ モード5：ジェネレータ

- a) プロトタイプ of ジェネレーション／実行
- b) リアルタイプ of ジェネレーション
- c) 関連資料の出力

システム名	ワークベンチ・DEMO		作成者	住友電ISYS (SS024)
	臨床検査部		作成日	1989.10.24
ベースパターンコード	LSMWC1	ベースパターン名	ワークシートテーブルメンテナンス	
ウインドウ番号	0. 6307 1.	エンジンタイプ	Engin 1	

.....1.....2.....3.....4.....5.....6.....7.....+78

1 ワークシートテーブルメンテナンス

2

3 部門コード _____

4 ワークシートNO. _____ ワークシート名 _____ ワークシートグループ _____

5

6 基本様式コード _____ 属性様式コード _____

7 代表ワークシートコード _____ 印字ワークシートコード・印字開始カラム _____

8 1検体毎の改行数 _____ 何検体毎に2行改行するか _____ 改行検体数 _____

9

10 1頁に印字するワークシート列数 _____ 1ワークシートの印字間隔 _____

11 サンプルID区分 _____ カップ数/テーブル _____ 開始サンプルID _____

12

13 結果印字(再発行時) _____ 未検査検体出力 _____

14 依頼ありマーク _____ 依頼無マーク _____

15

16

17 ・印字情報メンテナンスを行いますか? (Y/N) _____

18

19 ・サンプルID情報メンテナンスを行いますか? (Y/N) _____

20

21

22

縦行登録 リスト 削除

(処理概要)

検査部管理システム
ワークシートテーブルメンテナンス

システム名	ワークベンチ・DEMO		作成者	住友電工SYS (SS024)
	臨床検査部		作成日	1989.10.24
ベースパターンコード	LSMWC1	ベースパターン名	ワークシートテーブルメンテナンス	
ウインドウ番号	0. 6307 1.	エンジンタイプ	Engin 1	

10.....20.....30.....40.....50.....60.....70.....80
1	ワークシートテーブルメンテナンス
2	
3	部門コード入力
4	ワークシートコード入力
5	
6	基本様式コード入力
7	代表ワークシートコード入力
8	印刷毎の改行数
9	
10	直接印刷する印刷列数
11	結果印刷(再発行時)
12	
13	依頼あり
14	
15	
16	
17	日字情報検索システムを行いますか?
18	
19	種別別日字情報検索システムを行いますか?
20	
21	
22	
23	
24	

縦行登録 - リスト 削除

99 実行順序 マウス 領域 デコード

順序	セルルーチン	セルルーチンコメント	マウス		領域		デコード				
			Y座標 (幅)	X座標 (幅)	Y座標 (幅)	X座標 (幅)	Y座標 (幅)	X座標 (幅)			
1	LSMWBMO1	部門コード入力	3	1 3-21	19	3	1 22-23	2	3	1 26-33	8
2	LSMWC001	ワークシートコード入力	4	1 3-21	19	4	1 22-25	4			
3	LSMWCN01	ワークシート名称入力	4	1 26-41	16	4	1 42-49	8			
4	LSMNG001	ワークシートグループコード入力	4	1 52-73	22	4	1 74	1			
5	LSMNYC01	基本様式コード入力	6	1 3-21	19	6	1 22-23	2			
6	LSMZYC01	属性様式コード入力	6	1 26-41	16	6	1 42	1			
7	LSMDWC01	代表ワークシートコード入力	7	1 3-21	19	7	1 22-23	2			
8	LSMITB01	印字ワークシートコードの印字開始カラム入力	7	1 26-60	35	7	1 53-60	8			
9	LSMITB01	印字ワークシートコードの印字開始カラム入力	7	1 62-69	8	7	1 62-69	8			
10	LSMITB01	印字ワークシートコードの印字開始カラム入力	7	1 71-78	8	7	1 71-78	8			

システム名		ワークベンチ・DEMO 臨床検査部		作成者	住友電SYS (SS024)		
ベースパターンコード		LSMNC1		作成日	1989.10.24		
ベースパターン名		ワークシートテーブルメンテナンス					
エンジンタイプ		Engin 1					
ウインドウ番号		o. 6307 1.					
セルルーチン名	セルルーチン・コメント	セル変数	デコード	ヘルプコード	ヘルプルーチン名		
LSMWBMO1	部門コード入力	BM	BMN				
LSMWC001	ワークシートコード入力	WC		HLSM73			
LSMWCN01	ワークシート名称入力	WCN		HLSM01			
LSMNG001	ワークシートグループコード入力	WG		HLSM74			
LSMNYC01	基本様式コード入力	WYC		HLSM75			
LSMZCY01	属性様式	AYC		HLSM76			
LSMDWC01	複数ワークシート	DWC		HLSM77			
LSMITB01	複数ワークシートの印字ワークシートコード	ITB		HLSM78			
LSMITB01	複数ワークシートの印字ワークシートコード	ITB		HLSM78			
LSMITB01	複数ワークシートの印字ワークシートコード	ITB		HLSM78			
ファンクションキー				スタブ (使用・未使用とオプションの記入)			
キー番号	処 理	名 称 1	名 称 2	%START	<input type="radio"/>	%FMOUTL	<input type="radio"/>
1	LSFKWC01	登録	登録処理	%EQUIT	<input type="radio"/>	%FMOUTR	<input type="radio"/>
8	^LSMNC4	リスト	リスト発行	%EABORT	<input type="radio"/>	%FMOUTU	<input type="radio"/>
10	LSFKWC02	削除	削除処理	%INIT	<input type="radio"/>	%HLP1	<input type="radio"/>
				%DEFLT	<input type="radio"/>	%HLP2	<input type="radio"/>
				%ADRS	<input type="radio"/>	%INSRTR	<input type="radio"/>
				%EHLPA	<input type="radio"/>	%INSRTC	<input type="radio"/>
				%EHLPD	<input type="radio"/>	%REMCV	<input type="radio"/>
				%ERR	<input type="radio"/>	%REMR	<input type="radio"/>
				%EXODR	<input type="radio"/>	%EXIT	<input type="radio"/>
				%FKEY1	<input type="radio"/>	備考	
				%FKEY2	<input type="radio"/>		
				%FMOUTD	<input type="radio"/>		

看護勤務表作成支援システムの開発

Development of computer assisted nurse rostering system

○本多 正幸*

Masayuki HONDA

鶴岡 藤子**

Fujiko TSURUOKA

岩尾 幸子**

Sachiko IWAO

里村 洋一*

Yoichi SATOMURA

大塚 清子**

Kiyoko OHTSUKA

* 千葉大学医学部附属病院 医療情報部

Division of Medical Informatics, Chiba University Hospital

** 千葉大学医学部附属病院 看護部

Division of Nursing, Chiba University Hospital

本システムの基本設計は、パターン化を重視した典型的な勤務表を自動的に作成し、それをベースに各フロアごとの独自の条件を入力し、補正することとした。しかしながら、本基本設計では、フロアによる状況の差異、および各個人の希望の多様化への対応が難しいと考え、いくつかの勤務条件をチェックしながら、半自動的な勤務表作成についても対処できるような設計とした。なお本システムは、MUMPS(DSM-11)で書かれている。

The nurse rostering system is developed as a part of nursing information system. The system is designed for two step procedures. At the first step, automatic scheduling is done by arranging the typical pattern. It is difficult to satisfy the variation among floor's conditions and various personal requests. The specific factors on individual ward modify the generated schedule in the second step procedure. In this step, the checking and adjustment by manual procedure is adopted. This system is written by MUMPS.

キーワード：看護勤務表，勤務パターン，病院情報システム

KEYWORDS: Nurse Rostering, Working Pattern, Hospital Information System

〒280 千葉市亥鼻 1-8-1

1-8-1 Inohana, Chiba-shi, Chiba 280, JAPAN

1. はじめに

千葉大学病院情報システムは、これまでのシステム展開を踏まえつつ、平成3年スタート予定の総合的なシステムへの移行を目指している。次期システムの構想は、情報の継続性と蓄積性に重点を置いた、現システムの基盤の上に構築されたシステムを指向している。また、この構想は病歴データを中心とした診療情報を総合的に電子化したものを想定しており、とくに外来、病棟からの発生源入力情報に基づく症例データベースおよび電子カルテシステムを特徴としたものである[1]。この構想に向けて、病院情報システム推進委員会および10の検討部会が設置され、総合的かつ概念的な検討が進められている。また、具体的な検討および作業を行うシステム推進室も設置され、仕様書作りに取り組んでいる。

看護情報のシステム化については、総合的システム化の一環としての重要な要素として位置づけ、検討部会のひとつとしての看護業務検討部会をはじめとして、看護部内においても小委員会が設置され、その検討が進められている。本報告では、コンピュータ支援による看護勤務表の作成システムを取り上げる。次期システムへのスムーズな移行の為に、現在のシステム環境での開発を通じた検討を行い、院内のコンセンサスを固めておく必要があると考え、本システムの開発に着手した。勤務表の作成業務を取り上げた理由は、この業務が看護部内の独立した業務であり、複数部門関連業務と比べ取り組みやすいことである。また将来的には、次期構想のひとつである診療要員管理システムと連携を取ることにより、看護要員の基本情報が容易に得られると期待できることも理由の一つである。

本システムの基本設計は、パターン化を重視した典型的な勤務表を自動的に作成し、各看護ユニット（本院では各フロアを東西に分けた管理体制を基本としている）の配置人数などの条件より適合する勤務表を選択し、その勤務表に対し各個人を割り振る。このステップを第1段階とし、第2段階として各個人の希望条件を組み入れ、最終的な勤務表を完成させることとした。

本報告では、これまでのシステム展開を踏まえ、次節で看護システムの現状とその環境を紹介する。第3節では、勤務表作成支援システムの概要について、作成手順およびグローバル構成について述べる。最後に、今後の展開について簡単に触れてまとめる。

2. 看護システムの経過

これまでの看護情報のシステム化については、管理業務（表1）がその主なものであった。このシステムは、患者数や看護婦の勤務状況に関する、外来、病棟からの伝票による情報を、端末より一括入力し、コンピュータによる編集および集計を行い、表1の内容の出力を行うものである。看護部管理システムは、ネットワークシステムの構成ノードの一つである業務システム(PDP11/84, DSM-11)上に構築され、情報の一部が医事課における統計業務に活用されている。次期システムでは、患者個人および個々の看護要員についての情報がシステム化される予定である。したがって、表1のほとんどの項目に関しては、派生的に得られるであろう。例えば、勤務表をベースに実施状況へ変更すれば、休暇・実働状況に関する統計が容易く得られるであろう。次期システムは現在のシステムとの情報の継続性を保証するものでなくてはならないと考える。

3. システムの概要

図1は勤務表作成に際し、業務の流れを示したものである。まず前処理として、ユニットの構成人数による典型的勤務表の作成・登録処理を行う。これは、ユーティリティの一つとして構成されているが、現在のところ、いくつかの典型例を事前に登録しておくことにより対処している。将来は、勤務表の期間（現在は4週間）、夜勤パターン（現在は「深深準準」が基本）やその他の管理上の条件より自動作成を行う予定である。

ユーティリティとして、勤務表期間の設定の他に、勤務表の縦軸と横軸の条件である、割り振り基準の設定を準備している。これは、深夜勤や準夜勤の必要人数、その内リーダーグループに属する人の最低人数、勤務時間数、深夜勤回数、準夜勤回数、休日回数、連続勤務日数、禁忌パターンなどの全てにユニットに共通する条件の設定やその変更を行うものである。つまり、勤務表作成に当たっての遵守すべき絶対条件およびユニット単位の制約条件である。この割り振り基準が、典型的勤務表や実際の勤務表の完成作業を行う際のチェック条件となるものである。

図1の基本情報処理も前処理として必要である。図2のグローバルに示した看護婦および看護ユニット基本情報の入力および変更を行う。続いて、典型的勤務表に対し、各個人の割り当て処理を第1ステップとして行う。この際、前回勤務表を参考にして割り当てを行う。連続勤務日等の条件について、前後両勤務表の境界の連続性を考慮したチェックを行う。第2ステップでは、第1ステップで作成された勤務表を参考に、個人の希望条件を入力し、同時に割り振り基準との照合を行いながら、勤務表を完成させるものである。

4. 今後の展開

今後は、あるユニットにおいて本システムを試用し、その経験を通してシステム設計の再検討を行い、柔軟なシステムとして全ユニットへその範囲を広げて行く予定である。次期のトータルシステム化へ向けて、本システムの適用は教育的な意義も含めて重要なステップであると考えている。

参考文献

- 1) 本多正幸, 他: 診療情報の総合的電子化を目指した千葉大学病院医療情報システム構想, 映像情報MEDICAL, 21, 585-590, 1989

表1 看護部管理システム

- | |
|--|
| <ol style="list-style-type: none">1. 夜間外来患者統計業務2. 看護状況管理業務<ol style="list-style-type: none">①入院看護状況（日報，月報，年報）
：看護度別患者数，付添数，休暇・実働状況②外来看護状況（日報，月報，年報）
：新来・再来別患者数，外来手術数，休暇・実働状況 |
|--|

病院に置ける内線ファクスの効用

Benefit of Intra-Hospital Fax Network

三宅 和夫^{*}, 馬場 謙介^{***}, 杉崎 登^{*}, 小玉 隆一^{***}
Kazuo Miyake^{*}, Kensuke Baba^{***}, Noboru Sugizaki^{*} and Ryuichi Kodama^{***}

^{*}国立埼玉病院 臨床検査科, ^{**}同 臨床研究部, ^{***}同 神経内科

^{*}Laboratory of Clinical Pathology, National Hospital of Saitama,

^{**}Department of Clinical Research and ^{***}Division of Internal Medicine

ABSTRACT

In 1986, intrahospital FAX network system was established in this hospital which has no networks for computer system. Large benefits of this system are reported with a little demerit, according our experience of using it for this 3 years. Functional and economic comparison among this system, computer on line network system and pure hand work are roughly estimated. It is very clear that this system is much easier than telex system in operation, and computer system is much difficult than telex system in operation. The cost is cheaper and 6th of addition connection of a work station to previously set host computer which is used for bill of this hospital.

はじめに

今日程、低コスト・最大効率の追及の努力が、国立病院に求められたことはない。本院は、省力化、低コスト化をねらって3年前に院内に内線ファックス網を設置した¹⁾。電子計算機のネットワークがない本院で、内線ファックスは、情報伝達の方法として、偉力を発揮しているのでこれを紹介する。

Kazuo Miyake, Laboratory of Clinical Pathology, National Hospital of Saitama, 2-1 Suwa, Wako, Saitama 351-01 JAPAN (TEL: 0484-62-1101 ext: 361, FAX: 0484-64-1138)

導入前後の比較

本院では、病棟を中心に院内に内線ファックス網が張り巡らされている。内線ファックスが導入される前の伝票等の流れは、図1の如くであった。病棟と検査室との連絡業務についてみると、病棟看護婦が検体を検査科に届け、病棟看護婦は手ぶらで帰り、検査の終了時間に合わせて設定した定時に、再び、病棟看護婦が検査科に来て、伝票を持ち帰っていた。

使用している機械は、図2の如く、普通のファックスと同じであるが、電算機端末のようなキーボード操作は不要で、送り先もワンタッチでダイヤルできる。操作はいたって簡単で、操作に関するトレーニングも不要であることが、このシステムならではの特徴である。

ファックスの画質特性をみると、発信前の検査結果報告書（図3a）に比べ、受信状態（図3b）が、判読上劣っているとは云えない（病棟では、正規の検査結果報告書が到着するまで、受信用紙を保管する）。

病棟とのやりとりを、対検査室に限って、通常時、緊急時に分けて、導入前後で比較するとつぎの如くである（図4）。緊急検査でない時（通常時）は、病棟看護婦が依頼箋と検体を持って、検査室に検査依頼に来る。帰りは手ぶらで帰棟する。緊急性は少ないが、整理の都合などで結果を早めに知りたい時は、検査の進行を見計らって電話で問い合わせて来たり、また、内容が込み入っていたり、量が多いときは、随時病棟から、報告をとりに来たりしていた。実際、報告書を取りにくる回数は、各個病棟あたり一日3回以上であった。ファックスを導入してからは、検査結果ができ次第、ファックスで検査結果を各病棟に送っている。実質的な報告は、この時点で完了している。検査結果報告書は、公式文書保管の意味しなくなつたので、報告書の原本（普通の報告結果が入った伝票）は、夕方1回病棟から取りに来るので充分で、時に、夜勤者がとりに来ることさえあるのが現状である。

緊急時についてみると、ファックスで返事が届けられるようになり、検査室にとっては、煩わしい電話連絡が不要となつた。病棟にとっても、緊急時の忙しいときに、部所を離れなくて済むようになった。報告書の原本は、ゆっくり、定時の検査結果伝票の回収の時などに、通常時の報告と一緒に取りに来れるようになった。緊急検査を依頼する時は、病棟では患者の処置におわれていて、この時期に部所を離れないですむことは、医療の最前線では、省力化に止らない計り知れないメリットとして評価される。また、ファックス導入によって、電話連絡による聴き違いなどによるトラブルが激減した。

ファックス導入の経済効果の資料を得る目的で、概算す情報伝達に要する時間から病棟サイドの人件費を概算してみた。その結果を、病棟側の所要発信時間を各個病棟の平均で表1にまとめた(相手側のデータは含まれていない)。医事科、薬局、栄養室、検査室との発信に、費やされる病棟の単純人件費は、一病棟一日当り、伝票運びの場合: 6300円、院内FAXの場合: 1140円であった。経費が約1/5に節約されたことになる。この表の人件費には、直接の賃金で、保健、厚生費、職員が消費する冷暖費等は含まれない。これを含めると、常識的には、二倍になると見て差し支えないから約1/10になったとみてさしつかえない。

今回の調査では、病棟から見て相手側が費やした時間は調査対象から外した。但し、臨床検査室については、ファックス導入前のファックス導入前で、導入後の情報伝達の時間を比較すると、病棟側で10分から40秒に、検査室側で(各個病棟当り、一日当り)1分から20秒に短縮された。

端末の新規追加との比較

コンピュータシステムとの経済比較を見積もってみた。本院には、昭和56年にSHISシステムが導入され、医事会計業務が行われている。このSHISシステムに、一台の端末を付加する費用と、内線ファックスを設置する場合とを比較すると、約1/9であることは、記憶にとどめ置くに値するところである。

機能の面で比較してみよう。テレックスを中間に入れて比較すると、両者の機能が明確になってくる(表2)。テレックスが院内テレックスとして、馴染まなかったのは、入力にキーボード操作によらなければ、何一つとしてできなかったことが、コストと並んで大なる原因であつと考えるのが妥当である。ファックスはこの点、キーボードの操作は、不要で、①原稿を機械に挿入し、②送り先をワンタッチキーで指定しただけで、操作は全て完了する。後は、原稿を取り出し、送受信済マークを確認すれば、それで操作は完了する。コンピュータ操作は、テレックスより複雑である。

コンピュータでは、簡単な画像でさえ、送信しようとする至難の技であること考えると、ファックスは情報伝達の手段として、いかに優れているかが判る。本院では、処方箋のサインなどの画像の電送にも威力を発揮している。画像を再現できることは、テレックスにも現状のコンピュータにもないメリットである。反面、文字情報も画像情報として、伝達するので、伝達速度を問題にする向きもあるが、秒を惜しんで伝達しなければならない情報が、病院の中にどれだけあるであろうか。著者らは、病棟、薬局、医事会計、検査室の相互で秒を競って、情報の伝達をしたことがないし、それが必要だとする正統な主張を聞いたことがない。(この観点から質問すると(今学会でもそうであつたが)、「応答時間はもっと速くしたいと思つてる」とか「別のこともやるからネットワークは必要なのだ」とかの答えが返ってきて、真面目な議論にならない。コンピュータ界は、業界も学会も、この議論をタブーにして意識的に避けているのではあるまいか)。

確かに、コンピュータには、磁気化した情報資源の利用や、情報の加工等、コンピュータならではの機能を備えているが、この機能を必要としない業務に、通信の故にコンピュータを使うのは、得策でない。逆に、ファックスには、オフラインで複写機としても使えるメリットがあつて、複写機がない病棟で、診断書、紹介状、種々の病棟業務表、連絡などの事務的書類の複写に流用され、病棟の書類作業の効率化に貢献している。

メリットとデメリットのまとめ

院内に情報伝達手段を持たない本院で、通信内線ファックスを導入した際のメリットをまとめると次の通りである。誤りのない情報が迅速に伝えられるようになった。実際、病棟看護婦が部所を離れる時間が減り、その分本来の業務に専念できるようになった。特に、緊急検査時にはファックスの効用は、測り知れない。夜間に送信しておくことができるので、医事課、薬局、栄養管理室では、出勤と同時に業務を開始できるようになった。特に、退院会計が素早くなり、午前退院・午後入院（つまり病室の有効利用）に貢献している。煩わしい、電話による応対が減少した。操作が極く簡単であること、費用がかからないことがメリットとして上げられる（表3）。

送信先を確認ができないことが、あるとすれば、最大の欠点である。小さい紙で送信しても、受信用紙は、機械の規定の幅になってしまうこと位で、さしたる問題ではない。字の大きさについても、ファックスで送信することを念頭に置いた原稿であれば全く問題はない。なお、院内の多くの部所から設置の希望が強い（表4）。

むすび

コンピュータとの比較などについて若干の考察を加えながら、病棟にコンピュータネットワークを持たない本院で、低コストで導入した内線ファックスが、院内の情報伝達の手段として、有効に機能していることを報告しました。

文 献

- 1) 宮川俊篤、小玉隆一：院内電話ファックスによる情報伝達の経験。第44回国立病院療養所総合医学会総会，総合指定シンポジウム。仙台。1989

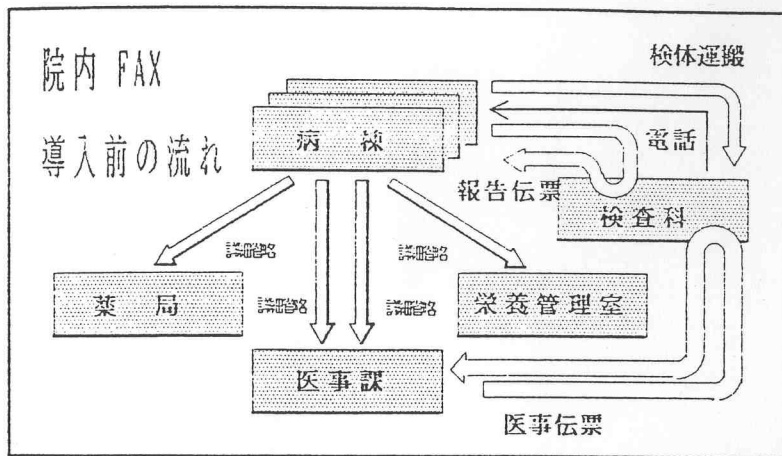


図1 検体・伝票の流れ

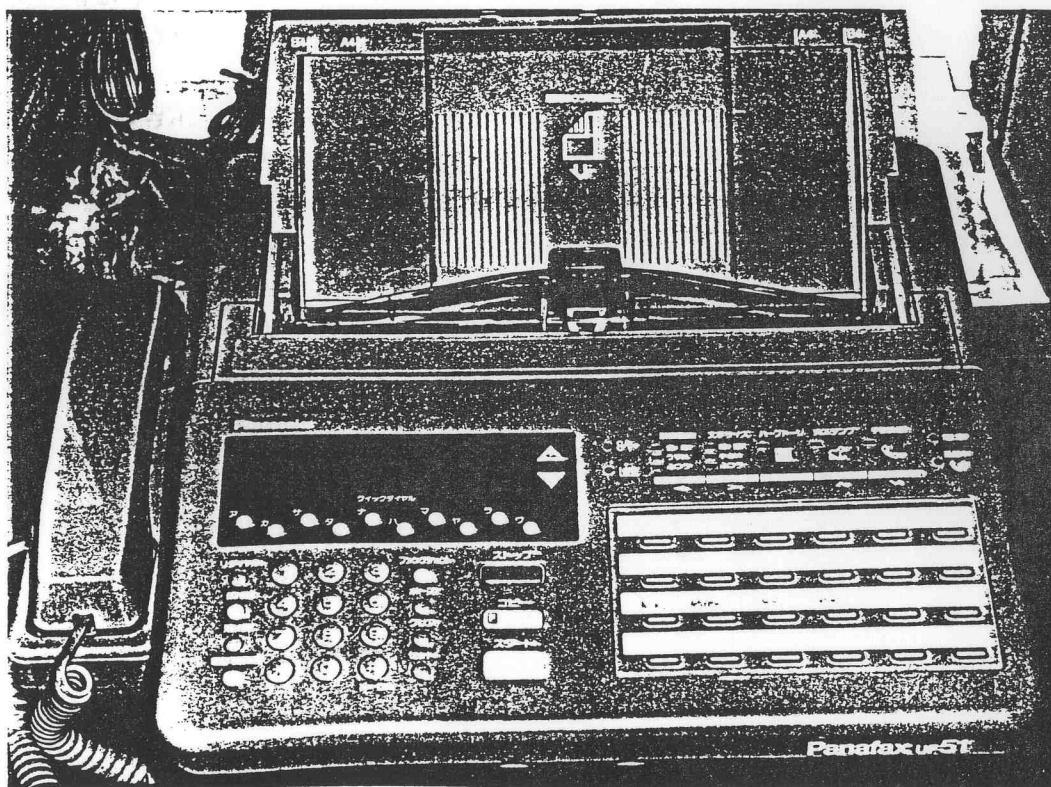


図2 内線 FAX の外観

臨床化学検査報告書		受付日 90年02月02日		
検体番号	11-4493-8	受付番号	0001	
氏名	イシマシロウ	性別	M 61	
生年月日	S04/01/29	検体採取日	90/02/02	
依頼元	E-4	検体種目		
コード	項目名	基準値	単位	成績
*1	総ビリルビン	0.1-1.0	mg/dl	H 1.5
*2	直接ビリルビン	0.0-0.2	mg/dl	H 0.3
*3	T-T-T	0-5	Ku	H 8.1
*4	Z-T-T	4-10	Ku	H 11.8
*5	GOT	8-36	Iu	18
*6	GPT	2-30	Iu	29
*7	LDH	230-430	Iu	L 222
*8	ALP	80-260	Iu	220
*9	LAP	40-70	Iu	53
*10	γ-GTP	7-40	Iu	H 69
*11	コリンエステラーゼ	2700-7500	Iu	4573
*12	CPK	22-230	Iu	58
*18	アミラーゼ	21-100	Iu	38
*21	総蛋白	6-8	g/dl	7.1
*23	A/G比	1.2-1.9		1.45
*24	総コレステロール	128-273	mg/dl	159
*25	トリグリセライド(TG)	50-150	mg/dl	104
*26	リン脂質(PL)	140-260	mg/dl	
*31	尿酸窒素(BUN)			

臨床化学検査報告書		受付日 90年02月02日		
検体番号	11-4493-8	受付番号	0001	
氏名	イシマシロウ	性別	M 61	
生年月日	S04/01/29	検体採取日	90/02/02	
依頼元	E-4	検体種目		
コード	項目名	基準値	単位	成績
*1	総ビリルビン	0.1-1.0	mg/dl	H 1.5
*2	直接ビリルビン	0.0-0.2	mg/dl	H 0.3
*3	T-T-T	0-5	Ku	H 8.1
*4	Z-T-T	4-10	Ku	H 11.8
*5	GOT	8-36	Iu	18
*6	GPT	2-30	Iu	29
*7	LDH	230-430	Iu	L 222
*8	ALP	80-260	Iu	220
*9	LAP	40-70	Iu	53
*10	γ-GTP	7-40	Iu	H 69
*11	コリンエステラーゼ	2700-7500	Iu	4573
*12	CPK	22-230	Iu	58
*18	アミラーゼ	21-100	Iu	38
*21	総蛋白	6-8	g/dl	7.1
*23	A/G比	1.2-1.9		1.45
*24	総コレステロール	128-273	mg/dl	159
*25	トリグリセライド(TG)	50-150	mg/dl	104
*26	リン脂質(PL)	140-260	mg/dl	
*31	尿酸窒素(BUN)			

(a) 発信原稿

(b) 受信状態

図3 画質の比較

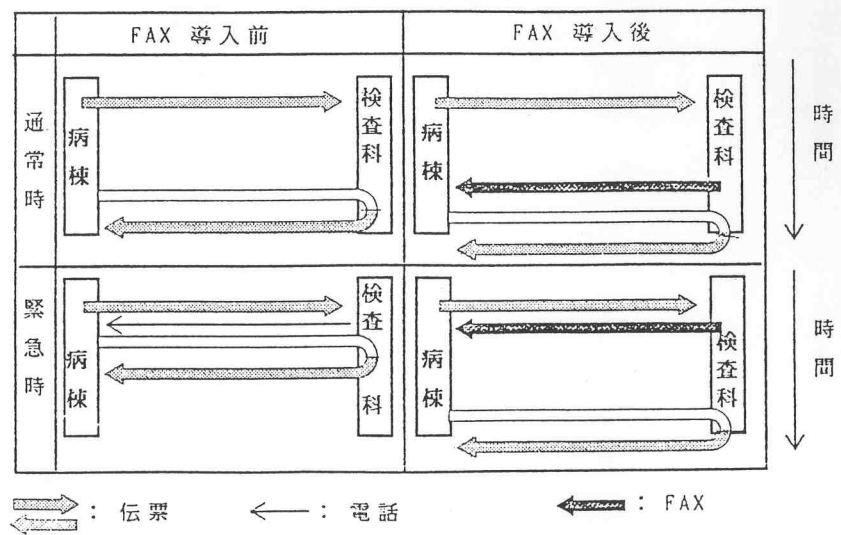


図4 通常時/緊急時別の伝票・情報の流れ

表1 病棟側の経費の概算

交信相手先	伝票運び			院内 FAX		
	往来頻度/day	所要時間/day	人件費円/day	交信頻度/day	所要時間/day	人件費円/day
医事課	10	50分	1500円	22.3	11分	330円
薬局	12	60分	1800円	26.5	14分	420円
米菱室	4	20分	600円	12.3	6分	180円
検査科	8	80分	2400円	14.5	7分	210円
合計	34	3hr	6300円	約77	38分	1140円

注) 交信相手先の経費は、含まれていない。

表2 三者の機能の比較

	FAX	TELEX	電算機
発信(入力)	紙上画像	キー操作	キー操作
発信(出力)	白黒画像	文字印字	文字の表示印字
情報処理	画像の再現	文字の再現	加工可記憶可
その他	複写機電話	? 及び電話	資源利用演算可等

表3 電話FAXの利点

- 情報が誤りなく迅速に伝えられる。
- 看護婦の病棟から離れる時間が減る。
- 夜間送信が可能なので、勤務開始と共に業務が開始できる。
- 電話による応待が減少する。
- 特別な機械操作の訓練がいらない。
- 費用がかからない。

表4 問題点と将来の展望

- 送信したものが目的の場所で受信できたかが判らない。
- 用紙・字の大きさ等に問題がある。
- 外来診療科・放射線・病歴室等からも設置の希望が出ている。

MUMPSの国際化への展望

若井一朗

マンブスシステム研究所

Views on MUMPS Internationalization

Ichiro Wakai

MUMPS System Laboratory

(要約) 標準 MUMPS は当初からオープンシステムを目指していたが一時スタンドアロンを標榜したため言語として孤立の危険があった。今は国際標準化がすすむ SSSP を十分に生かした 1990 年代のコンピュータ環境の主流に向かう時期である。MUMPS の国際化で得られる最大のメリットは国際間移送性とコンパチビリティにある。ANSI が FASTTRACK 法による ISO 各国の投票を提案し実施される時期に際し、真の国際化に成功するためには X11 と協調して国際的に最大限 generic な標準仕様と、漢字もふくめ各国 National Portability Requirements を最小限に完成する国際作業が必要となる。

(Abstract) Since the first standardization of MUMPS in 1975 it has aimed to be an open system, for some time being self-contained as stand-alone system which could have endangered itself as a solitary language. Current X11 activities have taken up various international movements toward common tools and platforms of computing environment to put MUMPS in the main stream of the 1990's by establishing external relations. As ANSI Standard MUMPS is to be processed for ISO FASTTRACK method among national bodies, endowment of maximum generic specifications as International Standard should be pursued leaving minimum room for National Portability Requirement issues on coded character set handling, cultural conventions, input/output support, multi-lingual implementation, etc. to attain maximum international transferability and compatibility.

キーワード： MUMPS 国際化 ANSI ISO JIS SSSP SC22

Key Words: MUMPS Internationalization ANSI ISO JIS SSSP SC22

オープンシステムとしての MUMPS

1975年に編集を終えたばかりの標準 MUMPS の Type-A 草案を日本に送付した時、NBS の J. T. O'Neill 氏は、「日本に送るまでに3ヶ月熟考し、これは天使の言語であると確信するに至った。日本にとってもメリットのある言語であることを祈る。」と筆者に書いてきた。MUMPS の天使性とは何であったかと言えば、表面的なユーザ指向性や対話型といったキャッチワードではなくて、本質的にオープンシステムであろうとする MUMPS の拠って立つ基盤を言ったのであった。医療情報システム開発センターの1975年度の重点計画である「医用データベース言語の開発」作業の一部として詳細吟味した調査報告書では、5種のデータベース言語を取り上げて移送可能な唯一の言語として MUMPS を推した [1]。

MUMPS は OS であると同時に通信系であるという理解は、それが他のシステムの支援を必要としないスタンドアロンであるという思想にも通じ、OS や他のシステムのツールを必要としない世界

461 名古屋市東区代官町 39-15 39-15 Daikan-cho, Higashi-ku, Nagoya 461

であるという信仰さえも生んだ。退役軍人省傘下の基幹メディカルセンター(VAMC)の自称アンダグループが完成した VA Kernel と数十の現場応用ソフトは 169 の VA 病院と 55 のサテライトクリニック、1 つの独立的外来病院、1 つのナーシングホームで重複開発労力をかけないで横につなげることを成し遂げた。1500 万人の人口の健康を管理するこの統合的な MUMPS システムが着手される以前(1982年) 172 の VA 病院で病院自動情報処理システムをもっていた所はただ 1 ヶ所に過ぎず、総ての試みは失敗に終わっていた。VA での驚くべき成功の結果、MUMPS が連邦情報処理標準言語(Federal Information Processing Standard -FIPS-)の制定をうけたのが1986年であった

オープンシステムとしての MUMPS の第 2 の証明は昨年 3 月に米国防総省がその陸海空軍病院の 754 ヶ所の情報処理を今後 8 年間にわたり VA Kernel と同種の MUMPS によるインテグレーションで行なうことを決定したことであった。病院群の 30% を海外にもつ米国防総省が 18 ヶ月にわたるテストの結果 MUMPS をセキュリティをもつオープンシステムであると判断し、マクドネル・ダグラス社の Cobol によるシステムの 2 倍の価値を認めた結果である。この前年 IBM は MUMPS が Cobol よりも優れていることをはっきりと認め、MUMPS/VM のリリースを発表し MUMPS プロジェクトのオフィスを設けた。

オープンシステムとしての MUMPS の第 3 の証明は本年の 9 月に VA 省がスポンサーとなって、病院画像処理の統括的な高速通信システムの標準化に乗りだし、これが FileMan を中心に展開されつつあることと、National Institute of Standards and Technology (NIST) がその計画に参画したことである。

オープンシステムでない MUMPS

以上の例を可能にしているのは機種/装備依存性のない標準言語仕様で書かれたアプリケーションのなせる技であった。しかしそれまでスタンドアロン言語としての性格をもっていた MUMPS も、Graphics や Networking のみを例にとっても MUMPS で標準化されているものはなかった。DEC の MUMPS と IBM の MUMPS がオープンに情報を交換するかという問題は痛切なものとなる。これは MUMPS のみの問題ではなくて他の高級言語にとっても同様ではあるが、オープンシステムの神話に頼り 20 年間にも及ぶ遺産をもっている MUMPS ユーザにとっては特に大きな問題となってくる。また今後の MUMPS ユーザーが MUMPS マシンのみに頼ってすべての情報処理を行うとは考えられないし、装備者は競ってひろく普及している事実上の標準 OS である MS-DOS やシェア率の高い機種に備わる機能をフルに利用しようとする。こうして徐々に MUMPS が移送性を失いそうになったのに対して、第 10 回ヨーロッパ MUMPS ユーザーズ大会のユーザー側から "MUMPS の環境結合を標準化せよ、これらの機能を既成の国際標準 (GKS, OSI, ISO) にインターフェースする方が全て便利でありオープンシステムを保つ道である (MUMPS は卵を生み乳をだす豚ではない!)" という声があがった [2]。

MUMPS 国際化への道

殆どのプログラミング言語や関連機器の標準は CBEMA (Computer and Business Equipment Manufacturers Association) が監視する X3 の各言語委員会によって開発されているが、MUMPS は当初から X11 という孤立的な言語委員会が開発されてきた。言語制定は "ANSI/MDC X11.1-年度" という形で行われる。X11 委員会 (MDC) は最近になって X3 を監視する CEBEMA と連絡を保つようになった。それは、他言語の開発と横の連絡をとりながら、正規なルートで国際化 (IS 化) を計る必要が生じたことが動機であり、他の言語環境が力強く国際標準化を歩んでいる現状に気がついたからである。

米国の標準化団体は迷路の様に絡みあい、一概には言えないが、国際的標準化面で活躍する日本の有識者で MUMPS が 20 年前からオープンシステムであったことを知って、それは X3 グループに始めから加入するべきであったという人もある。それは X3 が代表するコンピュータ言語には見当たらず、他の言語に影響する事もされる事もなく孤独に進化することは難しいという事と、さらに重要なのはコンピュータ産業が X3 に属しない言語をコンピュータ言語と認めないという通弊もあるからである。X3 グループは始めから商業的であり、委員会の会員は年間に数万-20万ドルという会費が必要

であり、その為にそれ以上の収益を上げるような団体でなければ所属する意味がない。X11は、極言すれば、ユーザーが仕様を作ってメーカーに吞ませるといふ所から出発したコンピュータ言語標準化委員会であって、言語の生い立ちの性格がX3と大きく違っている。

IBMがMUMPSをCobolよりも優れていると明言してその市場化にのりだしたインパクトは非常に大きく、比喩で言えば、嫡子のなかった将軍がそれまで一顧も与えなかった庶子を嫡子として発表したので似ている。別な比喩で言うと、十数年前にはX11は縦割りの幕藩体制を打破しようと脱藩して横の情報化を計ろうとした幕末の志士であり、池田屋騒動の様な事件さえあったが、参勤交代でお互いの権益を守りあっていた藩主らの中央政権はすでに交通や通信の国際自由化を進めている時世をむかえ、X11は天下の大勢に伍して国際化の道を歩まねば却って鎖国体制に取り残されることを痛感している。X11シンデレラが王子に見初められクイーンのチャンスに邂逅したと比喩する人もいる。

X11委員会には1978年から日本のMUGと沖電気、東芝とが参加し、翌年からヨーロッパ代表が多数参加した。X11委員会はそれを国際的な委員会として位置づけ、それと協調する国際的な委員会として、MUMPS Development Coordinating Committees (MDCC-Europe, MDCC-Japan, MDCC-Brazil)を規定している。沖、東芝の後からも多くの日本の企業も参加している。必要に応じて各国のNational Standard Bodyとの連絡を計るリエゾンが登録されており、日本は日本語マンブス標準化委員会委員長の東海大学大橋氏がその任に当たる。MDCC-Europeは数ヶ国間で頻繁に定期会議をもち、MDC規約通りに運営され、現在MUMPS開発の拠点が一移った感を与える程の実績を上げている。1984年度ANSI標準の文字列添え字がヨーロッパの提案で成立したことを見ても分かるように、MUMPS進化へのヨーロッパの寄与は大きく、始め日本もそのTGの1員であったレーザープリンタ、磁気テープ、など外部装置ハンドリングを始め、GKS, SQL, X-window, シーケンシャルファイル、TCP/IPなどを含むBindingを標準化する委員会(X11/SC9)はヨーロッパ勢によって進められている。MUMPS国際化の道として、X11に対して各国がその言語開発と協調し作業分担して結果を持ち寄ることが何よりも重要である。日本でも正規な会議を開き恒常的な委員をX11に送り込み、相応な分担を果たして行くことが今ほど強く望まれる時はない。後にのべるISO化の活動の一環として1990年の10月にはX11の全体会議がアムステルダムで行なわれる。

X11には9の委員会がSubcommitteeとして実際の起草を行なうが、それらの委員長を国別に示すと表1. のようになるが、殆どのSCにヨーロッパ委員が属していることも注目される。

X11 (MDC) 委員会	委員長	(米国)
X11/SC1 構文仕様		(米国)
X11/SC2 検定	欠	(日本 ->NIST ->NTIS)
X11/SC3 文書化		(米国)
X11/SC4 他団体リエゾン		(米国)
X11/SC5 安全保証		(米国)
X11/SC6 論理プログラミング		(日本)
X11/SC7 ネットワーキング		(米国)
X11/SC8 ライブラリー		(米国)
X11/SC9 結合		(欧州)

表1. X11委員会の構成

世界のMUMPSユーザー

国際的なMUMPSユーザ数の見積りはIBMが行なっているが少なめに見積られている。それに手直しを加えたものが表2.に示してある。ブラジルは独特の位置をもち、IBMの調査では国中の情報

処理の半分が MUMPS 機器を用いており、ブラジル独自の装備会社が存在する。ソビエトには2種の装備があり、C/UNIX 版の装備者にボリス・ブルシュテイン氏がいる。世界のプログラマーの数をみても、MUMPS の国際的普及は進んでおり、それと知らずに MUMPS を利用している人口はこれらの何倍にもものぼることを考えあわせれば、MUMPS が今後国際的な共通言語としてコンピュータの主流の中に入って行くことが予想される。それはスタンドアローンのANSI 標準 MUMPS が国際的に普及 (Spread) した時代を超えて、一方ではこれから国際標準化されるコンピュータ環境のうえでの共通性を持つ言語になり、他方では MUMPS が世界を自然言語を支え、移送性/コンパチビリティを保證する国際標準言語としての国際化 (Internationalization) を迎えようと言うのが国際 MUMPS 共同体の共通目標になっている。

アメリカ大陸		ヨーロッパ大陸		アジア-オーストラリア	
アメリカ	30000	英国	3300	日本	4000
カナダ	1300	オランダ	2400	オーストラリア	400
ブラジル	10000	ベルギー	300	シンガポール	700
		イタリア	390	中国	150
		フィンランド	600	韓国	50
		スイス	1400	マレーシア	30
		オーストリア	1800	タイ	30
		西ドイツ	3900	インドネシア	20
		フランス	1200		
		スペイン	2500		
		チェコ	300		
		ポーランド	400		
		ソビエト	4000		

表2. 世界の MUMPS ユーザ数

国際標準化された環境での MUMPS (Standard Software Support Platform)

現代のユーザーは言語が装備されたコンピュータシステムに依存する事なく、OS の国際的標準化などを通じて移送性を追求するし、ユーザーインタフェース (ウインドウ、グラフィックス、画像、スクリーン、アイコン) を MUMPS レベルで作ることをせず、MUMPS がそれ自体の中で全ての必要を満足させるのでなく、もっとユーザー指向型の標準的ユーザーインタフェースとユーザーレスポンスを求めている。これが達成されると MUMPS プログラムは軽量化されて今まで MUMPS で書かれた複雑な機能も MUMPS 過程を容れている環境へと飛び出してそこにある標準機能を使うことになる。X11 では国際標準化された OS と、その提供する標準的インタフェースプロトコールを利用することが MUMPS をコンピュータの国際的主流の中に持ち込む唯一の道であると強く意識している。

Standard Software Support Platform (SSSP) の動き

MUMPS 国際化の展望で、始めに考慮すべきものは国際化が進むソフトウェア環境である。それは全ての応用ソフトウェアが共通して用いる事のできる標準的支持台を作る事であり、これは Standard Software Support Platform (SSSP) と総称される。

SSSP は (1) OS, (2) Data Base System, (3) User Interface, (4) Net Work Communication, (5) Data Transfer を標準的支持台として応用ソフトウェアとの間に標準的なインタフェース仕様をもたせる動きであり、種々の団体が固有の名称をもつものを仕様化している。ヨーロッパ X/Open 共同体の Common

Application Environment (CAE), 米国 Open Software Foundation (OSF) の Application Environment Specification (AES), National Institute of Standards and Technology (NIST) の Application Portability Profile (APP), 日本 TAG (JTAG) の System Software Interface (SSI) が代表的なものといえる。このような標準的支持台となる基本ソフトウェアは幾種類かのコンピュータに移送できて、それ自体の環境で多国語をサポートし、国語を異にするために移送できないというものではないようにする必要が説かれている。応用ソフトウェアも数カ国語をサポートするど様な基本ソフトウェアの上にも移送できる必要がある。POSIX の国際化を進めている慶応大学の斎藤氏らはこのことを強調し、Internationalization と Local (National) Adaptation を弁別的に定義し、local (national) な形で容易に適応できるような、システムデザインによって Internationalization を達成する必要を説き (POSIX 国際化 WG/ITSCJ), 日本語環境については JTC1/SC22 N460 として ISO に登録されていることを述べている [3]。

開放型システム間相互接続 (Open Systems Interconnection -OSI) を現実的な標準プロトコールにしようとして活動している米国の組織は Corporation for Open Systems (COS) マークを作り、ユーザーも含めて OSI 通信 7 層モデルを実質的に推進しようとしている。種々の権威団体は夫々の SSSP が提供する機能にどの様に種々の高級言語がインタフェースするかを解説している。

NIST APP (米) Ada C Cobol Fortran Pascal
5 種

OSF AES (米) Ada C Cobol Fortran Pascal BASIC LISP
7 種

JTAG SSI (日) Ada C Cobol Fortran Pascal BASIC LISP APL MODULA-2 PL/1 SIMULA
11 種

この様にこれらの権威団体が仕様化している言語群の中には MUMPS は見当らない。MUMPS が将来国際的なオープンシステムとして標準化されるであろうこれらプラットホームのうえに、移送性、コンパチビリティ、そして相互操作性をもつ言語として搭載されるのを望むのであれば、MUG も X11 もこれらの表の中に MUMPS が含まれるように活動しなければならない事になる。日本の提案する SSI は最も優れたものであることは疑いないとされながらも政治的な駆け引きでは不利な立場にたつ事が多い。MUMPS の国際化が基本ソフトウェアの国際標準化とともに前進するならば、ヨーロッパ文字や 2 バイト系文字セットを考慮している SSI と日本語 MUMPS との親和性が最もよい点で、SSI の国際標準化を通じ MUMPS の国際化を推すのも日本の MUMPS の 1 つの立場であるかもしれない。

X11 では先ず ANSI 標準の MUMPS を ISO のレベルに上げ、そして国際標準言語の仲間入りをさせれば、SSSP らがこれを搭載可能な言語として国際標準化のために取上げると考えている。ユーザー団体 (MUG) もこれを強く支持し、その成功のために協力し運動している。ANSI の ISO 提案はこのような背景からなされた。この運動を日本をはじめ国際的なユーザー組織に十分に及ぼさなければ米英 2 国からの推薦のみで MUMPS の国際標準化が果して成るかという点も疑問がないとは言えない。これについては以下に触れたい。

国際標準言語としての MUMPS の動き (ISO/IEC JTC1/SC22)

前節でのべたように MUMPS の国際化をはかるにはまず ISO 化を急ぎ、国際標準化する SSSP に必然的に取り上げさせようとする働きかけが X11 から ANSI に、次いで ANSI から ISO に対して 1988 年 12 月 15 日に行われた (ISO/IEC JTC1/SC22 N590)。国際標準化機構は国際電機標準会議 (IEC) と

の合同技術委員会 (Joint Technical Committee) ISO/IEC/JTC1 をもつが、これは ANSI と協力関係にある。ANSI から新作業項目として提案された "Standardization of MUMPS" は3ヵ月の期限で各国の標準機構を回りそれらの投票結果が1989年4月に発表された (ISO/IEC JTC1/SC22 N636)。後者の内容をみると、逐語的に次のようになる。

ISO/IEC JTC1/SC22 N636 文書

投票要約：

提案賛成：13 カ国

1. FASTTRACK法による決定を支持：8 カ国

オーストリア、カナダ、フィンランド、フランス、日本、オランダ、英国、米国

2. 決定方法の指定のない支持：5 カ国

ベルギー、西ドイツ、ハンガリー、イタリー、ソ連

提案を支持し積極参加をする国：5 カ国

フランス、オランダ、英国、米国、ソ連

提案を支持するが積極参加しない国：4 カ国

オーストリア、ベルギー、イタリー、日本

提案反対国：0

回答保留国：0

投票棄権国：7 カ国

ブルガリア、中国、チェコ、デンマーク、イラン、ニュージーランド、スウェーデン

秘書官の決定： 提案に対する以上の結果と受理したコメントにより、SC22 秘書官は合衆国標準協会に対し ANSI MUMPS 標準を ISO 中央秘書局に提出すべく招聘しそれを FASTTRACK 法による処理に付するものとする。

付帯コメント1) 米国 (CT22/89-012, 1989 年2月24日)

合衆国は SC22 参加国に十分な MUMPS IS 化への興味があれば FASTTRACK 法による採決を強く推す。この推薦は次の理由による。

- * MUMPS 開発委員会は合衆国標準を開発する責任をもつが、殆ど世界各国の装備者とユーザを含んでいる。この委員会は合衆国はもとよりヨーロッパと日本のベンダーとユーザが参加している。もし SC22 WG が発足するならば、その参加者はほぼ同じ構成となろう。これが目的を果す為他に委員会に参加するために相当な付加費用が必要であるとは認めがたいし、また必要でもない。
- * 現在別々にヨーロッパと日本に MUMPS 開発協調委員会があり、言語開発問題の討議のために、夫々合衆国の MUMPS 開発委員会に代表を送っている。
- * MUMPS は国際的な要求に対して注意をはらってきた。例えば言語拡張の面では中国文字セット GB2312-80、日本文字セット X0208、および ISO 8859/1 を包含することを可能にした。MUMPS はロシア語アルファベットにも適応されており他の文字セットの装備も進んでいる。
- * MUMPS は1977年以来合衆国標準であり、言語追加改定は1984年に採用された。次期改定は現

在第2回レビューが行われており好ましい反応が予想される。ANSIに認可され次第、このFASTTRACK法による国際標準採択が適当であろう。

付帯コメント2) 英国(BSI 1989年3月16日)

英国はSC22 N590に対し以下のコメントをおこなう。

1. ANSI X1.1-1984をFASTTRACK処理に付すべきである。
2. 上記記載の標準に関する改定はISOにおける正規のワーキンググループ処理法に委ねられるべきである。

ISO FASTTRACK Method に関する展望

日本語MUMPS標準化委員会委員長の得た回答によれば、日本の標準団体としては今のところMUMPSのJIS化は考えていないという。それは日本で標準MUMPSの装備をして商品化している会社がないという単純明解な理由からであり、日本規格協会がCBEMAと同様な立場にあると考えれば理解できる。しかし、日本のNational BodyがANSI標準MUMPSのFASTTRACK法による国際投票に賛成していることには2つの重要な因子が考えられる。FASTTRACK法ではISOは直接最終的投票段階にはいり、各国代表による単純なYESかNOによる1回限りの投票でIS化は決定され、その前に国際環境を考慮した技術的な修正は全く許されない。ここではANSIの提出する仕様書がそのままISになるか否かを只一度で決める際どいものがあり、FASTTRACK法で1ヵ国が反対投票をすれば、IS化は失敗し再考慮の余地はない。国際標準となる言語には国際的なコンセンサスが成立しなければならない。各国の標準団体を代表する委員(ここではSC22)に1票も反対がないという背景が保証がない限りFASTTRACK法は大きな賭けということになる。

そのため日本を含む各国には、提出されるために改定中であるANSI草案の内容に対してその改定期間中に、各国にとっては何が削除されるべきであり、何が加えられるべきであるかと発言することで1種の影響を与える事ができる。

日本が1983年に2-バイト系文字セットへの拡張案を提出して以来、X11はそれをB-型草案には許しているが、A-型草案まで上げていない。X11がANSのための下請け委員会であるという前提は覆せない点と、またこれらの「外字」を扱うためには広くインドヨーロッパ語の文字セットや、外部装置を考慮した統一案も必要であるためである。それと同時に日本でのコンセンサスを経た結論から提案された「イ草案」にある\$ZWと\$ZPが装備依存的な構文をもち、Z-構文の中ではすでに装備者相互に衝突が生じつつある現状からみて、ANS以外の国家的な移送規準仕様のために「Y-構文を外字問題にかんしてリザーブする」提案が日本から米国装備者をスポンサーとして提出されたが、外国問題を討議するX11の適・不適論のために、これもB-草案のままになっている。1989 Type-A草案では文字数は日本案をいれてバイト数から文字数に変わり、言外に十進コード127の制限をはずしてあるが、第1部の1.2.1項に「MUMPSルティンとデータに許される文字は7ビットASCIIである」という定義があり、移送規準においても1.7節で「文字列に許される文字はASCII文字とする」と定められている。日本語は勿論ASCII以外を認めない国際標準が今後もあってよいのであろうか。

日本語の問題でヨーロッパと日本が協調する面はないが、共通の問題として取り上げることができるのは、ただ一つ「ASCIIに対する外字」としての国家標準文字セットである。日本が国際標準化に拒否権を用いる可能性として、1989年度ANS MUMPSの草案ではMUMPSプログラムで許される文字を印字可能な95個のASCII文字セットに限定し、そのまま国際標準として制定されると、少なくとも日本は非国際的JIS標準拡張で国内のみの移送性を計ることになり、さらには外国で日本製のMUMPS機器を入手できない障壁の遠因も考慮されるであろう。日本が拒否できないようなANSI草

案を作らねばならない。そしてこの10月に ANSI 提出を控えた最終 X11 会議がボストンで開かれた。

合理的な MUMPS IS 化の道は、MUMPS データを 7 bit, 8bit, 2 byte, 4 byte 文字などに拘泥しない最大限 generic な標準として制定し、どうしても国単位で拡張したりそれに伴う移送規準が必要ならば ISO でそれを付録に追加して行く事であろう。ANSI に拘泥してきた X11 は文字制限を変更するまいと考えられ、日本からは「このままでは日本は国際標準に違反する MUMPS を使う事になるので拒否の可能性がある。草案が ISO に提出される前に至急日本語の標準的拡張と移送規準、できればその他の国々の文字セットや移送規準を米国のそれと並べて移送基準の部に挿入する必要がある」と提案した。日本を除いて、この国内移送基準を提出できる所はない訳で、これは無理な提案なのである。結果としては「MUMPS 標準仕様での印字可能な文字セットは 7 ビット ASCII 文字に限る」という条項が削られ、移送規準では「装備は最低規準として ASCII 文字を含まねばならぬ」という条項に改められた。これは ANSI 標準とその移送性基準を曖昧化したかもしれないが、これは英国 BSI の一旦 1984 年度 ANSI を FASTTRACK 法で採択して、正規の ISO/WG で IS 改定をせよと言うコメントが採用されない以上これ以外に MUMPS 国際化に妥当な方法はなかったであろう。

日本から同じ提案を各 MUG, MDCC に送付したが、MDCC-Europe は日本の提案は尤もで重要であり、National Character Sets の前進を保証するに必要な段階を踏みたいと日本に提案した。ヨーロッパ MDCC は文字セットに先に凝って MUMPS の際移送性の損なわれことを恐れ、「1990年10月アムステルダムでの X11 会議を期限に、複数フォントサイズ、プロポーショナルスペーシング、グラフィックス、などを含むデータの外部表現その他の問題を解決する提案書を日本と共同で作成したい。自然語の操作はこの作業に含まれるが、X11 が今回 ASCII 文字への執着を放棄した以上特定文字セットとして持ち込むのは難しい。文字の内部文字処理の問題と切り離し外部表現の面で TG 活動ができよう。自然言語操作上、同じ原理で処理できる所は MUMPS の標準として最大限国際移送性を与えたい」とした。MDCC-E の方策は次のようである。(1) 特種符号の扱い：どんな符号もデータとしては文字列であり、それが 7 bit でも 4 byte 文字でも内部問題であり、ISO Draft Proposal (DP#10646) が 32 bit 格納で各国の文字セットをもつスーパーセットであるから、それを対象に多国語タスクグループは外部表現問題を統括する。(2) テキスト文字の Collating Order は MUMPS 以外の問題としても ISO/TC1/SC2/WG3/N37(1988) を参考に出来る。(3) Pattern Match の方法は \$TRANSLATE 関数が解決すると考えるが、もし Collating Order と Pattern Match を優雅に解決できれば全ての自然言語が同時に解決され、国別に解決される必要がなくなる。(4) 出力媒体上の表現も国定文字セットに依存しない文字間翻訳の標準化を早期解決したい。(5) 出力文字ディメンション：\$Width の問題は日本だけではない。文字の高さも国際標準で解決されるべきである。

X11 のヨーロッパ開催は ISO 化時期に照準を合わせた行事である。国際標準化はこうして行なわれ、X11 会長は十分に日本 MUG との意見調整をしたいので日本 MDCC の会期に合わせて訪日する。日本語標準 MUMPS イ草案は ISO に吸収されたり廃棄される部分もありそうである。上位コンパチを失わない間に専門家を多くもつ MDCC-Japan の開催を促したい。

MUMPS の国際化は国際標準化で達成される。それは X11 構成員である MDCC s が相互に寄りした上での最大限の歩み寄りで行なわれる。ISO 投票が行なわれるに際し MDCC-J の働きは成功への鍵の 1 つであると思われる。

参考文献

- [1] 医療情報システム開発センター：医用データ・ベース言語，海外の実情調査，March 1976.
- [2] Giere, W.: The tenth anniversary of a challenge - Software development in MUMPS, Proceedings of the Tenth MUG-Europe Meeting, Travemuende, 1985.
- [3] Saito, N. & Nakao, S.: Internationalization on POSIX, POSIX WG/ITSCJ Japan, April 1989.

電子化カルテの開発

○山下 芳範*、遠藤 晃*、河野 清司*、笹山 哲**

岡田 好一*、桜井 恒太郎*、高橋 隆*

* 京都大学医学部附属病院医療情報部

** 京都大学医学部附属病院放射線科

—はじめに—

京大病院では、新しい病院情報システム（KING）の構築を進めている。

このシステムでは、誰でも、どこでも、いつでも必要な情報を得ることができ、総合的な情報環境をめざしている。

このなかでも、診療に必要な情報としてのカルテを電子化し、記載内容の参照、検索が容易に行えることを計画している。

本研究は、このシステムの一環として、現MUMPSシステムをベースにした電子化カルテシステムである。

カルテには、患者の貴重な診療データが記載されているために、本院においても半永久的な取扱いで保存されており、膨大な量のカルテのために収容スペースやカルテ庫からの入出庫管理などの問題が表面化してきた。

しかし、カルテを光ディスクなどへの記録媒体の変更による物理的な体積の縮小では、記載されている診療情報を情報として取り扱うことができない。

本研究では、カルテに記載されている所見、病名等の文字情報、検査値等の数値情報、患者の状況を書いたイラスト等の図形情報をMUMPSの上で統合的に取り扱っている。

このため、カルテの検索や伝送も容易におこなえ、複数の場所での同時参照も可能である。

また、実際の診療に必要な各種オーダーや診療支援情報もこのカルテシステムから利用することが可能である。現在、このシステムのデータベースは、現京大病院の病歴検査データベース・オーダー用データベースを仮定して作成している。

—システムの概要—

ほんシステムは、VAX-DASMとVMS上のDEC-WINDOWS（X-

WINDOWの上位互換)とを結合することにより実現している。(図1)

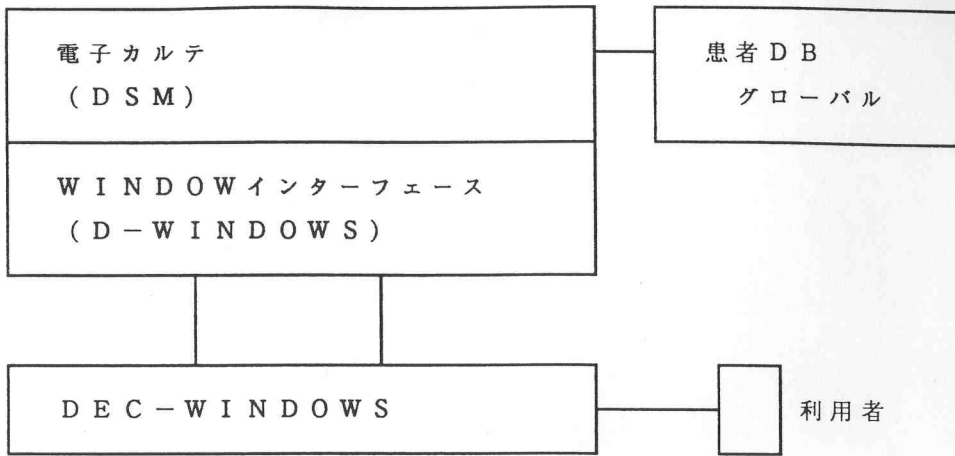


図1. 電子カルテシステムの概要

患者基本情報のデータベースおよび検査結果データベースは、現在の京大病院での利用しているグローバルと同等である。

入力された、テキストは分類タグを付けた形でグローバルに格納する。

イラスト等の情報は、プリミティブとしてパラメータ化し、WINDOW上での表現のままグローバルに格納している。

このため、ノードデータとしてこれらのパラメータをデータ属性として採用しているため、\$O関数により容易に検索ができる。

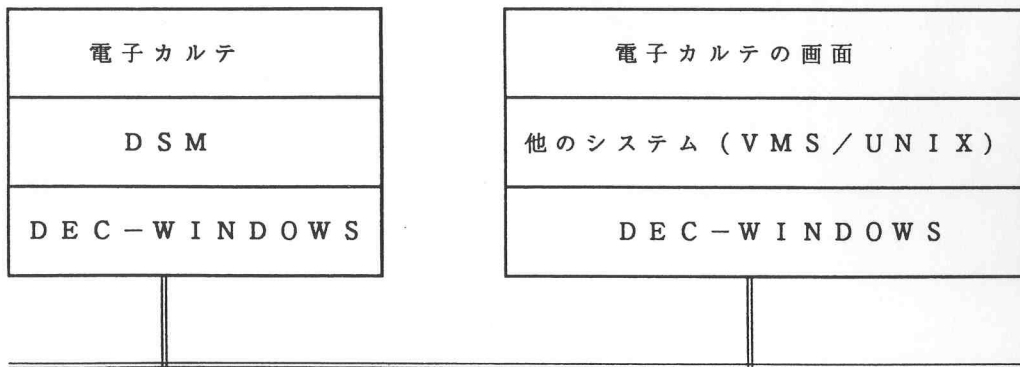


図2. ネットワーク上での活用

また、WINDOWSのサーバー・クライアントの関係を利用する事により、画面だけを他のシステム(異OSも可)でも実行することができる。

今後は、MUMPSをベースとした、マルチメディアデータベースへと発展させていく予定である。

D A S L の 試 用

○山下 芳範、遠藤 晃、高橋 隆

京都大学医学部附属病院医療情報部

京大病院では、第5次病院情報システムにおいて給食情報システムの構築を行ってきた。

このシステムにおいて、試験的にD A S Lを用いて開発を行なったので、実際の開発での利用を報告する。

この給食システムは、2つのサブシステムから成り立っている。

1つは、栄養管理サブシステムで病態栄養部における献立作成および病棟向けの支援情報提供・食事申し込みを行なうのである。

もう1つは、食数管理サブシステムで医事課における食数の管理・会計関連の処理を行なう部分である。

今回、前者の栄養管理システムにおいて、実際に病棟で運用するにあたりいくつかの不具合が発生した。

これらの問題を解決するために、内部のロジックおよびマン・マシンインターフェイス部分の修正が必要になった。

この作業は、情報部よりM U M P S経験者2名が4日間、約20時間の講習を受けたのち、修正作業を行なった。

内部ロジックおよびマン・マシンインターフェイスの変更にあたり約1人月、メインメニューの変更・他の支援情報の追加および電子メールの追加に約1人月要した。

内部を熟知していない者が、これだけ短期間で変更を行なった理由にD A S Lが、グローバルの構造を管理しており、また、プログラムもモジュールとして管理し、これが画面に対応している点があげられる。

D A S Lは、M U M P Sのソースをジェネレートするものであり、ジェネレートされたソースは人が容易に解析できるものではないが、D A S Lだけを見ればM U M P Sで書かれたルーチンを利用する事も可能であり（本システムでも従来のルーチンを利用している）、開発の効率を考えれば、非常に優れたツールと言える。

ユーザー側からみたアクセルワークベンチ

ACCEL WORKBENCH FOR USER-ORIENTED DEVELOPMENT

○山本 和子、須藤 正克

Kazuko YAMAMOTO and Masakatsu SUDO

福井医科大学医学情報センター

Center of Medical Informatics, Fukui Medical School

In 1989, two VAX 8350 systems were introduced as the 3rd generation systems in Fukui Medical School Hospital. In order to enable system development by school staffs, the system uses the easy-to use MUMPS language for development and the ACCEL workbench for screen design and programing. This report outlines the ACCEL workbench and describes merit/demerit of the ACCEL workbench for user-oriented development.

Keywords: workbench, program development support, MUMPS

1. はじめに

ACCELは住友電気工業株式会社がプログラム開発ツールとマンマシンインターフェースのドライバーとして開発した製品で、ACCELにはマルチウインドウとグラフィックのためのエミュレータ機能、データ入出力エンジン、画面設計、データ辞書、プログラム開発サポート（ワークベンチ）等の機能があり、パソコンを通常の端末やグラフィック端末にしたり、その他、多目的のワークステーションとして利用することができる。

今回、本学の病棟オーダリングシステムの開発に際してACCELワークベンチを使用した〔1. 2〕ので、利用者側の立場から、使用経験者6名の意見をもとに、アクセルワークベンチによるプログラムのジェネレーションの速度、使い勝手、長所、短所等について報告する。

2. ワークベンチを用いたプログラム作成例とその速度

産婦人科基本指示入力（一画面）のアプリケーションプログラムの作成を例にしてワークベンチのジェネレーションの速度を測定した。入力または出力の項目は5.2項目である。ワークベンチでは次の5段階を経てプログラムが作成される。そこで、各

福井県吉田郡松岡町下合月23

23 Shimoaizuki, Matsuoka-cho, Yoshida-gun, Fukui 910-11

段階毎に作業に要した時間を測定した。但し手作業の時間も入っている。

第1段階：画面設計

基本指示入力画面を設計し、直接ワークベンチの画面に項目名を設計書どりの位置にワープロ入力し、次に項目名の色を指定する。この間約5分を要した。

第2段階：データの入力領域指定

項目名の横にマウスの入力領域とデータの表示領域を指定する。この間3分。

第3段階：実行順序の定義

カーソルの移動順を指定するのに5分、変数指定に17分を要した。

第4段階：更新

上記内容をグローバルファイルに登録するのに1分。これで準備が完了。

第5段階：リアル（プログラムのジェネレーション）

ここから、プログラムのジェネレーションが始まる。その様子が図1のウィンドウの中で見る事が出来る。プログラムが出来上がるまでに8分かかりプログラムのセーブに2分かかった。

以上、約41分で4本のプログラムが作成された。出来上がった画面が図2で、指定した順にカーソルが動く。

3. ACCELワークベンチ使用経験者の意見調査

3-1. 方法

本学の病棟オーダリングシステムの開発にACCELワークベンチを使用した経験者6名を対象に、質問紙法によりワークベンチの使い勝手、使用してみた感じ、長所・短所、改良点等について調査した。

3-2. 成績並びに考察

1) MUMPS経験月数

調査対象者のMUMPS経験月数は図3のように分布し平均45か月、年令とは関係がない。

2) ワークベンチを用いて開発したプログラム本数

ワークベンチを用いて開発したプログラム本数は一人平均33本で、図4に示したように、年令、MUMPS経験月数とは関係がない。

3) ワークベンチは楽かどうか？

ワークベンチはMUMPSでプログラミングするよりも楽かどうかの質問には、図5に示したように、意見が別れている。ワークベンチ使用経験プログラム本数との関係で見ると、図6に示したように、経験本数が少ない者が楽でないと答えており、慣れれば楽になるように思われる。MUMPS経験月数との関係で見ると、図7に示したように、経験月数が多い程楽でないと答えており、使い慣れた言語を使用したいという一般的なプログラマーの心理が出ていると思われる。

4) ワークベンチを使えるか検討し使える場合はまずワークベンチを使用するか？

図1. プログラムのジェネレーション

```

7778 base=FBKHS5 col (M= A= D= R= F=)
回経妊 回経産 入院日: ( 日) 基準日: ( 日)(術後)
身長: ( cm)( ) HBs:
体重: ( kg)( ) TPHA:
血压: ( )
検温: ( ) 畜尿:
TPR記載: ( ) 検尿:
安静度: オーダー状況

W %EC_"[>5h"
S ZEWDWVER(%WDW)=22,ZEWDWVEC(%WDW)=%WXWLG
::
SELED X "F F="_%TAGCEL_" D %PAINTD+F"
::
SELEC X "F F="_%TAGCEL_" D %PAINTC+F"
::
W %EC_"[m"
    
```

Mode 27 24 37
22:20 PM

図2. 基本指示入力画面

```

7778 産科大科基本指示入力 22:20 PM
回経妊 回経産 入院日: ( 日) 基準日: ( 日)(術後)
身長: ( cm)( ) HBs:
体重: ( kg)( ) TPHA:
血压: ( )
検温: ( ) 畜尿:
TPR記載: ( ) 検尿:
安静度: オーダー状況
保 清: 薬剤 内服: 検査 一般:
排 泄: 注射: 生理:
障 害: 食事: 放部:
定期処置: 残尿測定: 膀胱洗浄:
NST: 時間注射: 腎洗:
児心音: 時間採血: 尿路変更:
妊婦管理: 時間採尿: 人工肛門:
術婦管理: ECG 監視: バルーン:
持続点滴: G W:
精密持続: ドレーン:
    
```

縦行

図3. MUMPS経験月数

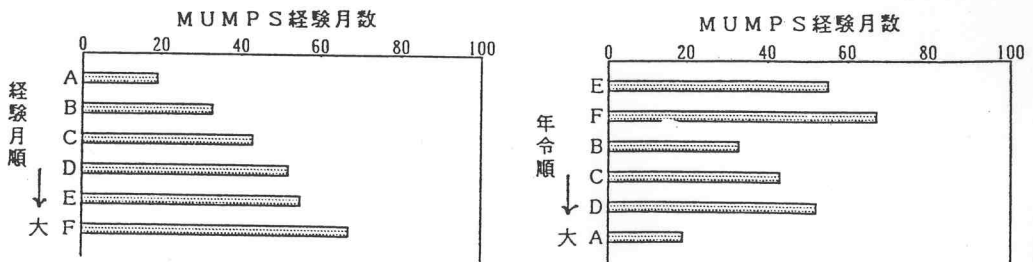


図4. ワークベンチを用いて開発したプログラム本数

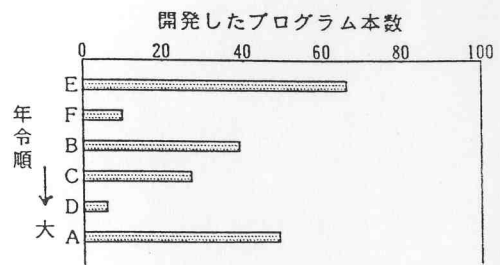
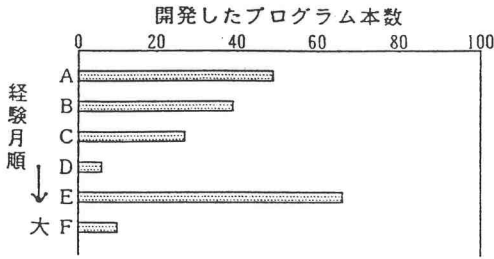


図5. ワークベンチは普通にプログラミングするよりも

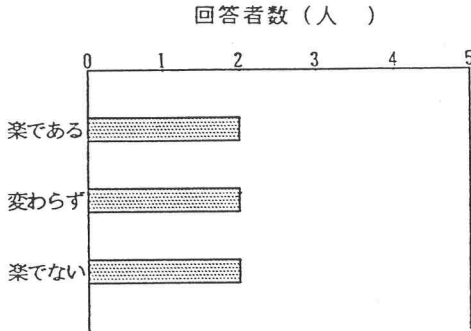


図8. ワークベンチを使えるか検討し
使える場合は先ずワークベンチを

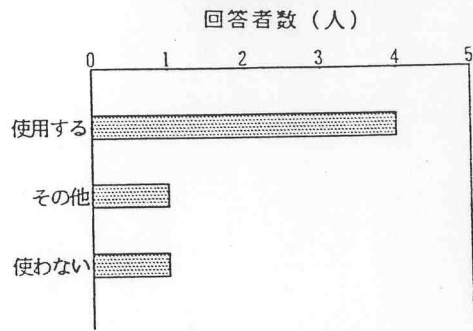


図6. ワークベンチは普通にプログラミングするよりも

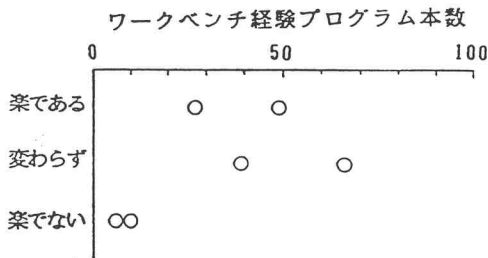


図9. ワークベンチを使えるか検討し
使える場合は先ずワークベンチを

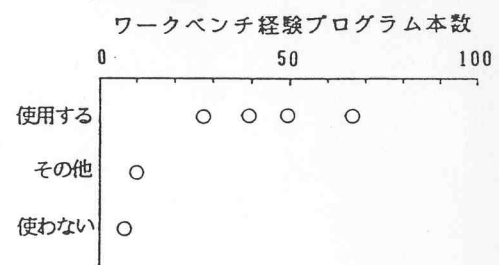


図7. ワークベンチは普通にプログラミングするよりも

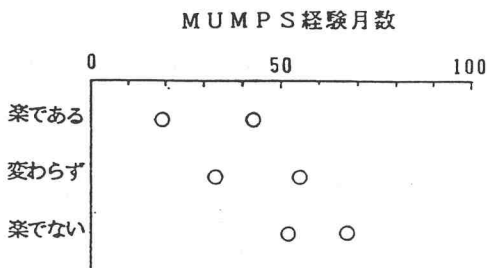
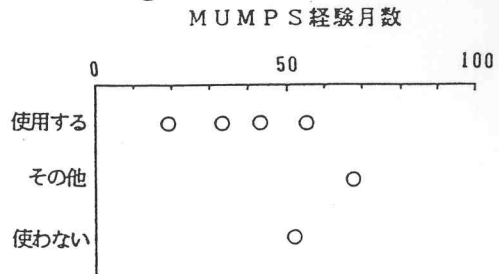


図10. ワークベンチを使えるか検討し
使える場合は先ずワークベンチを



ワークベンチは楽かどうかで意見が別れているにもかかわらず、ワークベンチを使うか検討し使える場合はまずワークベンチを使用するかどうかの質問には、図8に示したように、使用すると答えた者が多かった。楽でなくても使用したほうが便利な場合があるからであろう。ここで「その他」とあるのは「時と場合による」という回答である。ワークベンチ使用経験プログラム本数との関係で見ると、図9に示したように経験本数が多い程ワークベンチを使用すると答えており、経験を積んで慣れてくると使いやすくなるように思われる。MUMPS経験月数との関係を見ると、図10に示したように、どちらかと言うと、MUMPS経験月数の少ない方がワークベンチを使用すると答えているように見受けられ、慣れた言語を使用したい気持ちが窺える。

5) ワークベンチに対する感想

ワークベンチに対する感想としてプラス意見は、「レイアウト設計は通常に作る時よりも楽」「パソコンの機能をうまく引き出している」「ウインドウの中での表示で見やすい」全体として「なかなか素晴らしい」等で、マイナス意見は「決りが非常に多い」「内部共通変数が多いので使いにくい」「1つのプログラムを作るのに時間がかかる」等であり、「ワークベンチを使用すると負荷がかかるのでは」「数年後のメンテナンスは？」等の不安もあった。概してワークベンチ使用経験の多い者にプラス意見が強かった。

6) ワークベンチの長所・短所

ワークベンチの長所・短所としてあげられた内容を表1に示す。

表1. ワークベンチの長所・短所

長 所	短 所
画面設計が容易	プログラム作成に決りが多い
ウインドウの重ね表示が容易	プログラム修正に時間がかかる
カラーの表示が容易	処理速度が遅い
ファンクションキーの表示	データ量に制限がある
矢印でカーソルの移動が容易	
エラー情報の表示が容易	
ヘルプ情報の表示が容易	

長所としては画面設計が容易である点を全員が指摘していた。その他、これまでのMUMPSが持っていなかったパソコンの機能を利用したプログラムを作成する時に便利であるとの意見が多い。短所としては変数等の決りの難かしさ、入力項目の追加等のプログラム修正もワークベンチをかけなおさなければならない面倒さ等があげられている。

7) ワークベンチの改良

ワークベンチをどのように改良すれば良いかの質問に対しては、「マニュアルの作成」「処理速度の向上」等があげられ、積極的な改善意見は出ていない。

4. ACCELワークベンチの評価

以上、ワークベンチ使用経験者6名の意見だけでは、例数が少なすぎて普遍性があるかどうか疑わしい。しかし、ウインドウやカラーやマウス等のMUMPS以外の機能を利用しようとする時に、ワークベンチは大いに威力を発揮するものと言えよう。また、これらの機能を利用して新しい画面設計が出来るようになり、用途がさらに拡大される。

ワークベンチ使用経験者6名の意見をベースにして、独断的ではあるが、ワークベンチを用いた場合のメリットをまとめると

- a. 開発が容易で生産性の向上がはかれる
- b. パソコンの機能を利用した面白いシステムが組める
- c. ジェネレートされたプログラムはMUMPSであるため分かり易く、後で機能を追加できて便利である
- d. プログラムの質が保証される
- e. 統合性に優れる
- f. 自動ドキュメンテーションが可能

等があげられよう。

ドキュメンテーションに関しては、プログラムのドキュメントだけでなくオペレーション時のエラーが記録されていてバグの発見やオペレーションミスによるエラー時のデータの修正等にも便利であった。使い勝手に関しては、調査結果からみて、初めは抵抗あるものの多くのプログラミング言語と同様、慣れれば楽になるように思われる。処理速度に関しては、遅いという意見が見られるが、画面設計からプログラム完了まで40分程で完成できれば十分だと思われる。ただ現在のワークベンチは完成品でないので、ワークベンチでジェネレーションされたプログラムに追加しなければならない部分がある。完全なワークベンチの開発が今後の課題であろう。

5. おわりに

病棟オーダリングの開発に際して、プログラム開発ツールとしてACCELワークベンチを使用してみたところ、マルチウインドウやマウスを使用する画面のプログラム作成に大いに威力を発揮することが分かった。今後、製品として完成されることを期待したい。

文 献

- [1] 山本和子、須藤正克、他：福井医科大学総合医療情報システムの開発、第8回医療情報学連合大会論文集、285-288、1988。
- [2] Yamamoto, K. et al.: Network and Integration in Fukui Medical School Information system, MEDINFO 89, 330-333, 1989.

日本デジタルイクイップメント株式会社

西日本第一ソフトウェア部 山本広蔵

日本語DASLとは

日本語DASL (DSM Application Software Library) は、日本語VAX DSM PLUS上で稼動する会話型アプリケーションを効率よく開発し、その実行環境を生成するツールでMUMPSの4GLとも呼ばれるプロダクトです。日本語DASLは、種々多様なデータベースを基本とする高度に洗練されたアプリケーションの作成からシステム保守までを一貫して支援します。

日本語DASLを使用することによって、アプリケーション開発者は、

- ★ 短期間に高品質なアプリケーションが作成できます。
- ★ アプリケーションのコーディング、定義の標準化がなされるため、保守、拡張が容易になります。
- ★ 日本語DASLが、ドキュメントを自動生成するため、ドキュメント作成の時間が短縮できます。
- ★ アプリケーションの作成、試行、修正を即座に繰り返し行えるプロトタイピングアプローチができます。

また、エンドユーザにとっても、

- ★ メニュー形式の操作によって目的の処理を実行することができます。
- ★ ユーザインターフェイスが統一されるため、操作が簡単です。
- ★ オンラインヘルプ機能により、タイムリーにヘルプ情報を参照でき、マニュアルがなくても操作できるようになります。

日本語DASLによるアプリケーション開発

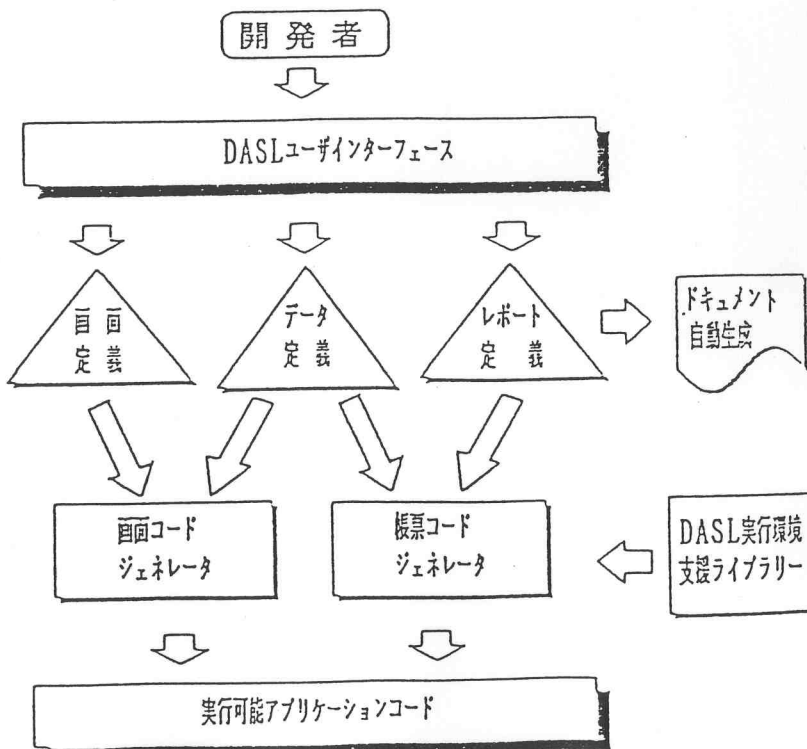
日本語DASLによるアプリケーション開発は、非常に簡単です。

アプリケーションで使うデータベースと入出力画面、出力レポートの定義を行うだけです。あとは日本語DASL自身が、非常に高品質で効率の良い日本語VAX DSMのプログラムを自動生成します。

また、ほとんどの作業はメニュー形式のインターフェイスを通じて対話的に行うことができるため操作が簡単です。

さらに、日本語DASLでは日本語VAX DSM PLUS上で既に作成された各種の集計処理やバッチ的なデータ更新処理などの個別処理ルーチンを組み込んだり、呼び出すことができます。このようにして、現在稼動しているアプリケーションを無駄にすることなく、日本語DASLを核としたトータルなアプリケーションシステムを構築することができます。

日本語DASL アーキテクチャ



日本語DASLモジュール

1. データベース定義

日本語DASLアプリケーションは、データベース（グローバル）を”データネーム”と呼ばれる様々な属性を持つアイテムを通じてアクセスします。これにより、画面やレポートを作成する時、データベースのアクセスについてグローバル構造や、データの性質を意識する必要はまったくありません。また、”データネーム”の定義は、”データ辞書”と呼ばれるデータベースに格納され、一元管理されます。このモジュールでは、”データネーム”の作成、管理を行います。

データネームには、以下のような定義が含まれます：

- ・実際にデータが格納される格納領域（グローバルロケーション）
- ・長さ、タイプなどのデータ属性
- ・データ入力時のプロンプトおよびヘルプ情報
- ・ユーザ入力時のデータ整合性チェック仕様
- ・高度なデータベースアクセスのためのクロスリファレンスインデックス指定

2. 画面定義

データベース定義をもとにユーザとアプリケーションをインターフェイスするための各種入出力画面の作成とプログラムの生成（コンパイル）を行います。

日本語DASLでは、以下の3タイプのスクリーンを提供します：

- ・データ入出力用画面
- ・オプション選択（メニュー）用画面
- ・ヘルプ用画面

ファンクションキーを使った画面のハンドリングや、スクロール領域の設定を含んだような複雑な画面の制御もその定義をするだけで日本語DASLがプログラムを生成します。データの入出力画面の設計においては、”ディスプレイデザイナー”と呼ばれる高度な画面設計用フルスクリーンエディタを用いてイメージどりの画面を設計することができます。さらに、オンラインヘルプの画面の作成もVAX/VMSが提供するフルスクリーンエディタを用いて作成することができます。

3. レポート定義

出力帳票の定義、編集とプログラムの生成（コンパイル）を行います。帳票そのものの属性（ページ長、ページ幅など）と出力されるデータの設定（データベースおよびデータの抽出条件）、ソートキー、出力位置などの指定をすれば、あとは日本語DASLが帳票出力用のプログラムを自動生成します。また、見出しや改ページなどもデータに合わせて自動制御し、帳票出力時に便利な各種統計関数も用意されています。さらに、今までに作成したレポートを保管管理しておき、あとから再出力することのできるレポートディレクトリ機能も備えています。

4. 開発環境の設定

日本語DASLでは、日付や時間のフォーマット、ファンクションキーや変換キーの配置など、アプリケーションの実行時に変化する設定をパラメータを変更するだけで即座にアプリケーションに反映させることができます。また、アプリケーションを機能別にグループとして分割し、グループごとにアプリケーションを管理することができます。さらに、日本語VAX DSMのルーチンエディタや、スクリーンおよびレポートのテンプレートを作成するユーティリティなど、アプリケーション開発を支援する様々なメニューが用意されています。

5. アプリケーション環境の設定

日本語DASLでは、作成されたアプリケーションの実行環境までも一元管理します。パスワードや特権コードを用いてアプリケーション内のメニュー単位でのアクセス制御などができますので、データの機密保護も万全です。また、アプリケーションの稼動状況や、各ユーザのアカウント情報なども自動収集されます。さらに、エラーロギング機能が備わっていますのでアプリケーションの異常に対しても即座に対応することができます。

VAX/DSMとDECwindowsの結合について

太井 寿幸

日本デジタルイクイップメント株式会社

VAX/DSMとDECwindowsとのインターフェイスにWindow Tool (仮称 D-windows) を用意することによって、DSM からウィンドウのハンドリングやイベントドリブンな処理にも対応できるようにした。DSMからは\$ZCALL関数を使用することによってD-windowsを呼び出す。

はじめに

コンピュータの処理能力はここ数年著しく進歩しているが、人間がコンピュータと接する部分であるユーザーインターフェイスについてはそれほど進歩していないように思える。その原因の1つとしてキャラクター端末の機能の限界があげられるが、ワークステーションが世の中に登場したことにより、マルチウィンドウやマウス、グラフィックの機能を用いて、より人間が使いやすいようなユーザーインターフェイスに改善していく必要がある。

ここで紹介するD-windowsを使用することによってマルチウィンドウ環境に対応したDSMのアプリケーションを容易に作成することが可能になる。

D-Windowsとは

通常、マルチウィンドウ対応で非同期入力処理や、グラフィック表示をおこなうアプリケーションプログラムを作成する場合、プログラムの大半はウィンドウ管理や、イベントの処理、表示をするための処理等に費やされ、実際に行ないたい処理は全体のなかのごくわずかになる。

また、プログラミングする前にウィンドウ関係のマニュアルも読まなければならない。そのようなことは非常に面倒なことであり、できあがったプログラムも複雑になることからバグのないプログラムを作成することは時間のかかる大変な作業になってしまう。複雑な処理はアプリケーション以外のところでハンドリングしてくれるようなWindow Toolがあれば、アプリケーションは大幅にすっきりするようになる。

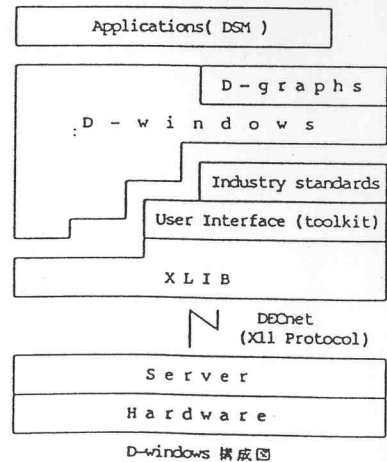
D-windowsはDECwindows上でマルチウィンドウ・アプリケーションを作成するとき、先に述べたような複雑な処理をハンドリングするためのWindow Toolである。

D-windowsの構成は右の通り。

D-windowsを使用することによって、DECwindowsで用意されているPopUP MenuやToggle Button, Scroll Bar, Iconといったウィジェットを簡単に利用することができる。

また、D-windowsにはグラフを簡単に作成、表示するためのGraph Tool(仮称 D-graphs)も用意されている。D-graphsには円、棒、折れ線、相関図の4種類のグラフがあり、DSMから表示させたいデータを与えることにより自動的に目盛りの計算をおこないグラフの表示をすることができる。

表示の機能として大量のデータのためのスクロールや、広範囲にデータが拡散しているときのための表示変更、ある目盛りに対してラインを設定する機能等も用意されている。



D-Windowsの使用方法

プログラミング

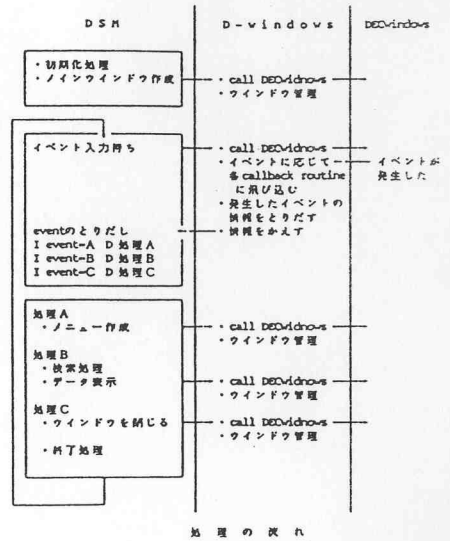
DSM, D-windows, DECwindowsの処理の流れを右に示す。

まず最初にDSMからD-windowsに対する初期化処理をおこなう。

ここで、DECwindowsのUIL (User Interface Language, Widget) を用いたユーザーインターフェイスを記述するための言語) で定義されたUIDファイルの指定をする。

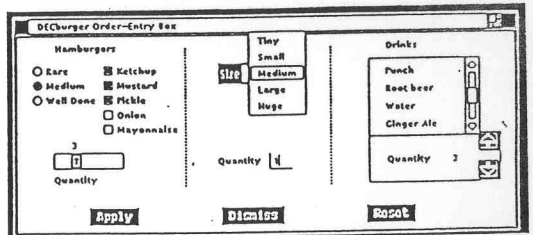
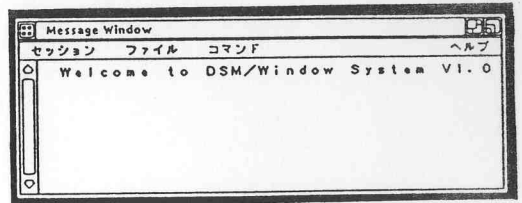
つぎに、非同期入力に対するイベントの入力待ち状態にする。

あとは入力されたイベントに対して、次の処理をおこなうためのルーチンの起動をする。



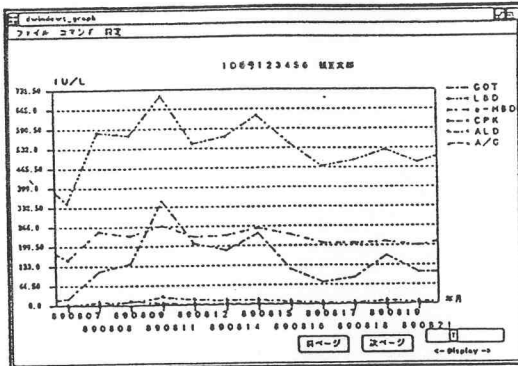
プログラム例

```
SAMPLE ; D-windows サンプルプログラム
;
;
; 初期化処理とメインウィンドウ作成
;
S STAT=$ZC(UINITIAL,"demo.uid","Message Window",10,10,TOP)
S KANJI=$ZC(KJCDSTS)
S STAT=$ZC(UKANJI,KANJI)
S STAT=$ZC(UMESSAGE," Welcome to DSM/Window System V1.0")
S ENDMODE=0
F QQ=0:0 Q:ENDMODE=1 D MAINEVNT
Q
;
MAINEVNT ; メインルーチンに対するイベントハンドリング
:
S STAT=$ZC(UEVENT,KIND,TYP,BT,CL,X,Y,W,H,SIZE,TEXT,TAG,V)
; 発生したイベントに従って各処理を行なう
I KIND=SYSEVENT D Q
.I TAG=1 D S ENDMODE=1 ;
..S STAT=$ZC(XUNNCLD,wnawin) ; D-windowsの終了
..S STAT=$ZC(UTERM) ;
.I TAG=3 D CMD0003(.TOPLEV3) ; コマンド番号3を実行する
.I TAG=4 D CMD0004(.TOPLEV4) ; コマンド番号4を実行する
.I TAG=5 D CMD0005(.TOPLEV5) ; コマンド番号5を実行する
:
Q
CMD0003(TOPLEV3) ; グラフの表示処理をする
S STAT=$ZC(XAPPRSH,"D-graphs","D-graphs",0,TOPLEV3)
S STAT=$ZC(DFETOWID,"D-graph",TOPLEV3)
S STAT=$ZC(XMANAQLD,wdwingr)
:
F QQ=0:0 Q:EDMODE=1 D CWEV0003
Q
CWEV0003 ; コマンド番号3に対するイベントハンドリング
:
S STAT=$ZC(UEVENT,KIND,TYP,BT,CL,X,Y,W,H,SIZE,TEXT,TAG,V)
I KIND=SYSEVENT D Q
.I TYPE=ACTINPT D Q
..I TAG=wgrscoci D
...D CIRCLE ; 円グラフの表示処理をする
:
Q
```

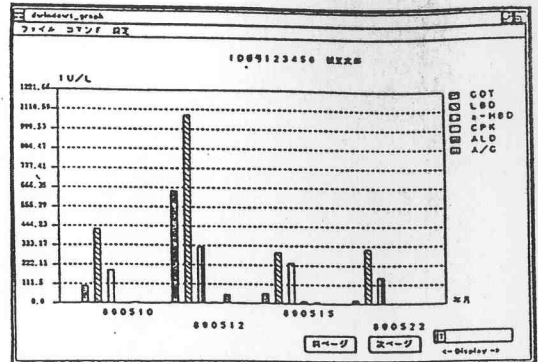


画面例

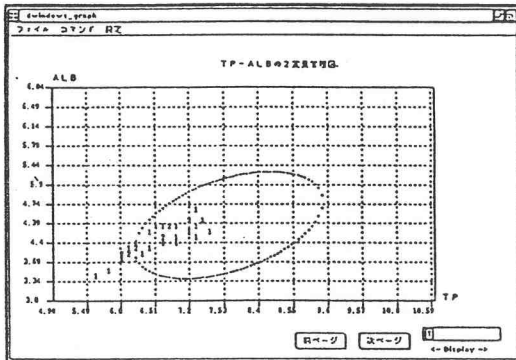
D-Graphsの表示例



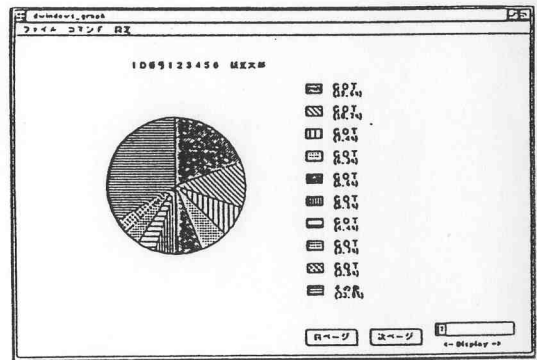
折れ線グラフ



棒グラフ



相関グラフ



円グラフ

おわりに

D-windows では、DECwindowsの機能以外の簡易システムとしてD-graphsのほかにも D-matrix (表計算用のマトリクステーブルシステム) や、D-image (イメージデータの表示システム) などを開発中である。

特に、イメージデータシステムに関しては光ファイルとの接続や動画処理などについてもおこなう予定である。

D-windows は、利用者にとって「最もなじみやすいユーザーインタフェイス」を提供するための「プログラムを作成するツール」として発展させていきたい。

