

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。

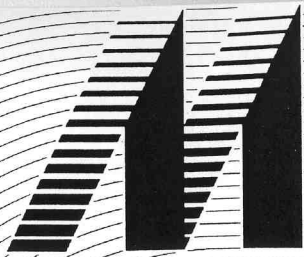
Copy right : M Technology Association - Japan

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](mailto:youichi@keyaki.cc.u-tokai.ac.jp)



*Technology  
Association  
Japan*

# Mumps

Vol.21, 1998

Journal of MTA-Japan



目 次

頁

■巻頭言

Mはどの道を選ぶのか.....笹川 紀夫 1

■論 文

The Superiority of M-Technology for the Hospital Information System..... 3

Hee Sun Jeon

Yun Sik Kwak

Hune Cho

Hyung Soo Kim

Soo Yong Jang

WAN環境でのDTMによる業務システムの構築事例.....広瀬 清司 9

DSM/CGI プログラミング.....今井 敏雄 15

DSM/DASL テスト・ターミナル・エミレータ.....今井 敏雄 23

インターネット技術とDSM/WWW接続ツール.....城崎 礼子 29

杉山 閑照

日本語 DSM V6.4 と Visual M の紹介.....佐藤比呂志 35

データ・ウェアハウスとデータマイニング.....今泉 幸雄 39

■日本MUG事務局からのお知らせ

「日本Mテクノロジー学会」ご入会のご案内.....47

「日本Mテクノロジー学会」規約.....51

■資 料

投稿規定.....57

編集後記.....60

表紙装丁.....岡田 好一

## Mはどの道を選ぶのか

島根医科大学医学部医療情報学講座 笹川 紀夫

出雲で開かれた第20回エム・テクノロジー学会大会の玉造温泉の懇親会において、旅館の1室で若井先生を中心にベンダー、ユーザーが垣根を越えてMへの熱い思いを語り合っていたことが昨日のこのように思い出される。

あれから5年、コンピュータを取り巻く環境は我々の想像を絶するスピードで正に進化している。

パソコンのCPUはDX2、66MHzからPentium II 333MHzに、ハードディスクはせいぜい250MBであったのが4GB以上に、メモリーは最大で384MB以上まで拡張されてきており、Windowsは未完成ながら従来のワークステーションの機能を持ち、ハード的にもソフト的にもはやパソコンとワークステーションの垣根は無くなったと言って良い時代になった。

GUI(Graphical User Interface)は、コンピュータの操作環境及び概念を一新し、従来とは全く違った思考回路が要求されている。

Windowsの普及は、Visual BasicやVisual C、SQLといった開発及びデータベースの共通言語を標準化させ、ソフトウェアはこれらに対応した製品でないと生き残れないのが現状であり、このような環境の中、Mも確実に時代のニーズに合わせて進化してきている。

GUIへの対応やVBとの親和性によって従来とは違った操作環境になって、Mのユーザーなら誰でも知っている、他のRDBと比してコンパクトかつ柔軟及び強力な検索機能を備えたデータベース、少ないコマンドで構築できるプログラム。これらはこれからのデータベースの求められる機能を備えており、またインターネットの急速な普及はMにとっては追い風である。

しかしながら、コンピュータ雑誌等でMが取り上げられた記憶は皆無で、一般はおろか開発者ですらへはほとんど浸透していないのが現状である。

この点に関しては、学会や雑誌Mumps等で議論されていることなので省略するが、実際1ユーザーの立場で考えた場合、業務や研究でデータベースを使う場合AccessとMとどちらを選択するかと言えば、やはりAccessを選んでしまう。データが何万件もあり大容量になるならば考え直すが。

その理由としては情報量の圧倒的な違いである。もしAccessのことで分からないことがあれば、一般の書店に行って解説本を読んだり、インターネット上検索するなどする事によって、自分が行っている仕事の参考にする事が容易であるが、Mの場合はどうだろう。

## 2 Mはどの道を選ぶのか

またデータベースだけ考えても、確かに少ない容量、早い検索、柔軟な設計が可能であるが、コンピュータのハードの急速な進歩は、ある程度の容量のデータ(とは言っても数年前なら考えられない容量であるが)では体感的に感じられず、Mの特長が何も見えてはこないのが実際である。

今後Mはより一層パーソナルなデータベースを目指して進むのか、あるいは現在の隠れたデータベースのベストセラーを目指して進むのかの岐路に立っていると思う。

趣味の話で申し訳ないが、自動車のスバルは、最初に軽乗用車を開発し、一般に受け入れられた。しかしその後時代の流れと共にその方向を見失い、低迷し熱狂的なユーザーにしか受け入れられない時期があったが、優れた4WDシステムとワゴンという上物を乗せることによって1つのステータスを築き、その後ハイパワーなエンジンによってラリーというフィールドで世界に名を知らしめている。

Mはその特長をスポイルしてまで大衆受けする必要は無いと思う。むしろその特長を生かしつつ何か確固たるステータスを作り、それを基にして新たなステータスを作り上げる方向に向かうべきであろうし、そのためにどうしたら良いか、ベンダー、ユーザーの密接な関係をより一層密として議論する必要がある。玉造温泉での1室の光景が幻ではなくそれが大きな輪になることを願って止まない。

# The Superiority of M-Technology for the Hospital Information System

Hee Sun Jeon, Yun Sik Kwak, Hune Cho, Hyung Soo Kim\* and Soo Yong Jang\*

*Medical Informatics Research Center, Ajou University Medical Center, Suwon, Korea  
and Daewoo Information Systems, Inc\*, Seoul, Korea*

## Abstract

Ajou University Hospital is a 900-bed tertiary care teaching hospital with approximately 160 medical staff, 200 house staff and 500 registered nurses. They treat on an average 850 inpatients and over 1,800 outpatients daily. There is also a 36-bed Emergency Care Unit which is the major trauma referral center in the region. To support the above clinical activities, ATOM-1 (Ajou Total Medical Information System-1), open distributed system, has been developed since November, 1990 and was implemented in June, 1994 when the hospital was newly opened. The hardware of the system consists of 23 Unix Servers (U6000/65X19 and U6000/85X4), 300 PCs, 170 Workstations, and 240 Printers. The network is with FDDI backbone, and also run by TCP/IP, Ingres/Net and 5 bridge routers. The Client is SUN (Sparc Classic). The client user interface is on SUN Motif (OSF/Motif). The software was written in Windows/4GL and C for client and Vision, C, COBOL (MF) for servers. The DBMS is Ingres v6.4 and the operating system is Unix SVR4. The applications include Order Communication System for all physician orders, Admission, Discharge, Transfer, Scheduling, Patients Registration, Pathology, Laboratory, Blood Bank, Pharmacy, Dietetics, X-ray, Nuclear Medicine, Physiological Function Tests, Account Receivables, Billing, Medical Record Tracking, Physical Therapy, Material Management and Cost Analysis. The total investments so far are as follows : (1) hardware cost was \$6 million ; and (2) software development cost was \$4 million.

One and a half years after implementing the system, user satisfaction has been high.

However, there are serious drawbacks of the system, such as (1) RDBMS has very high overhead cost for the servers ; (2) data processing has been relatively slow as compare to that by MUMPS DBMS and server shadowing is not possible; and consequently the hardware cost has been very high.

Therefore, it's been decided in early 1995 to convert the ATOM-1 to Open M based down sized system from Ingres based without change of system architecture in order to lower the hardware cost and to improve system performance. The X-ray, Medical Record Tracking, Dietetics, Physiological Function Tests, Material Management and Cost Analysis had already been migrated to Open M. The conversion of all other programs is underway.

In conclusion, a Client/Server based open distributed hospital information system with Open M has resolved all of the above RDBMS based system's difficulties. The M based system is on an average 50 times faster and hardware requirement is approximately 30% of that of RDBMS.

## 1. Introduction

Ajou University Hospital is a 900-bed tertiary care teaching hospital which has been established to support patient care, medical research and training of medical, nursing and medical technology students, interns, residents and fellows. The hospital was opened in June, 1994. There are approximately 160 medical staff, 200 house staff, 500 registered nurses working for 850 inpatients and over 1,800 outpatients daily. This is the major referral medical center in Kyunggi Province, immediately outside of Seoul (nation's capital) and supporting about 5 million population.

The ATOM-1 (Ajou Total Medical Information System-1) development project was conceived in November, 1990 and began to be implemented in June, 1994.

The database management system used at which time was Ingres v6.4, 23 Unisys Servers, 170 SUN(Sparc Classic) Workstation-Client-, 300 IBM Compatible PC and 240 printers were used as the hardware. The network was with FDDI backbone, 5 bridge routers, and run by TCP/IP protocol and Ingres Net.

The Order Entry, Admission, Discharge, Transfer, Scheduling, Patient Registration, Pathology, Laboratory, Blood Bank, Pharmacy, X-ray, Nuclear Medicine, Dietetics, Special Function Laboratories, Account Receivables, Billing, Medical Record Tracking and Physical Therapy.

However, the Ingres based System showed very high overhead cost for servers. The processing time was

#### 4 The Superiority of M-Technology for the Hospital Information System

relatively slow ; and the downtime was observed due to maintenance.

Therefore, we decided to convert the current system run by Ingres to Open M(InterSystem, Inc., Cambridge, MA, USA) based system. This conversion project was begun in July 1995. We successfully migrated the software programs supporting Diagnostic Radiology, Dietetics, Medical Record Tracking, Special Function Laboratories activities in February, 1996. The patients demographic database was still resided in the Ingres database(DB). The data conversation from Ingres DB to M DB was accomplished by using M SQL and communications between servers occurred through M Net.

This paper briefly describes the background of developing the ATOM-1 together with that of M conversion of aforementioned 4 departments. We also had a unique opportunity to compare the performance of the data processing systems between Ingres and M technology based.

In those 2 systems, the hardware and network backbone were the same except for the DB, programming languages and communication tool namely Ingres Net and M Net.

#### 2. Philosophy of Ingres Based System Development

While construction of hospital was on it's way, the Hospital Information System Development Team was formed in November, 1990.

The philosophy of system development by the medical center's top management was as follows :

- Implement OCS for all physician and nursing orders ;
- Implement an open-distributed hospital information system ; and
- Develop an infrastructure for future total hospital information system.

On the basis of these, system development objectives had been formulated.

- For the Technical Aspect : A distributed open client/server hospital information system should be developed ; an OCS for all orders should be developed ; and interface for connecting equipments should be developed.

- For the business aspect : Hospital services provided to patients should be improved ; hospital employee's productivity should be increased ; and quality of health care should be improved.

- For the OCS : Physicians should use the system for ordering ; GUI based OCS should be developed to make ordering easier ; group order system should be developed ; test results and order status reviews should be available ; system response time should be faster ; and nursing staff should be liberated from clerical tasks.

#### 3. System Development Costs of Ingres Based System

The total personnel cost was \$4 million with 1,300 man months. The total hardware cost was \$6 million which includes servers, network and peripheral devices.

#### 4. Ingres Based System Platform and System Configuration

##### 4.1. System Platform (Table 1)

Table 1. System Platform

	Client	Server
User Interface	SUN Motif (OSF/motif)	
Language & Tool	Windows/4GL, C	vision, C, COBOL (MF)
Application	All OCS entry Review reports	All ancillary support Order status review Patient registration
Database		Ingres v6.4
Network		TCP/IP, Ingres/Net
O/S		Unix (SVR 4)
Hardware		SUN (Sparc Classic) Unisys (U6000/65, U6000/85)



#### 4.2. System Configuration (Fig.1)

Servers and peripheral devices are connected through FDDI backbone with 5 bridge routers. The system currently has 23 Unix servers (U6000/85 x 4, U6000/65 x 19), 300 PCs, 170 workstations, and 240 printers.

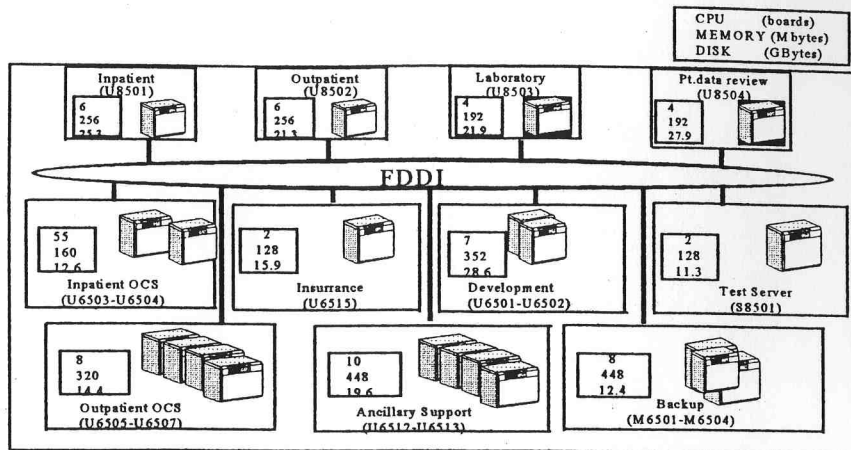


Fig 1. Ingres System Configuration

### 5. Strong Points of Ingres Based System

#### 5.1. Technical Aspect

- Open-distributed system can easily be expanded and future technology can be used without changing current system[1,2].
- Client/server system increases system's efficiency.
- Store and Forward (SF) is used to communicate servers for transaction.
- Automated devices are interfaced and barcoded labels are effectively utilized.
- GUI provides advantages to physician ordering mechanism.

#### 5.2. Business Aspect

- Physicians and nurses can order directly through OCS.
- Nursing package automatically communicated with ordering information (DB)
- OCS information communicates with patients administration package, so that patients waiting time is minimized.
- Reimbursement request is more accurate and faster.

### 6. Weak Points of Ingres Based System

#### 6.1 Technical Aspect

- Cost of system is too high due to relatively slow transaction and response time of the system. There have been frequent dead lock of the system.
  - Server and disk shadowing and mirroring are not easily accomplished.
- Therefore there is approximately 1.5% of downtime for system maintenance.

### 7. M-Technology Based System Development

M-Technology has been chosen to improve the system performance and eliminate the weaknesses of the Ingres based system. The existing hardware, network and peripheral devices should not be altered and open distributed Client/Server architecture should be supported by the new DBMS. Therefore, the M-technology was selected.

6 The Superiority of M-Technology for the Hospital Information System

The migration of Ingres system to M system should be accomplished with minimum work disruption. The migration was carefully planned and executed gradually. This M system development started in July, 1995 and the software packages for the Diagnostic Radiology, Special Function Laboratories, Dietetics, Medical Record Tracking and Acquisition of Materials and Inventory Management were implemented in February, 1996.

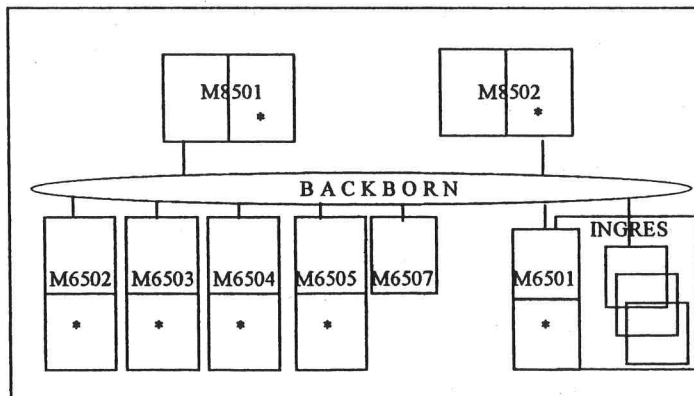
The details of migration plan(Table 2), development tools(Table 3) and the system architectures(Fig.2) are as follows.

**Table 2. System with M-technology Migration Plan**

DBMS	:	DTM
Program	:	ISM, MSQ, Visual M
Network	:	M Net
Hardware	:	Same
Network backbone	:	Same
Entire system development and Implementation	:	1995.7 - 1996.11
Cost	DB	\$ 400,000
	Personnel	\$ 2,353,000
	Total	\$ 2,753,000

**Table 3. Development of System with ISM**

Program development tools	
Pure M	: System programming
BLBX	: Data access routine for integrity
McGen	: M code generator
DD	: Data dictionary
MQ	: Data query tool
M Net	: Communication
Back-up and replication system development	
- System and database back-up	
- Transaction unit replication	
- Simplification of data recovery procedure using journal file	



\* Back-up Servers

**Fig 2. M-technology System Configuration**

### 8. Comparison of Response Time and Hardware Resource Requirement Between Ingres and M-Technology Based Systems.

The system response time of simple bulk test was on an average 105 times faster with M-technology and the application test resulted in on an average 2.6 times faster with M-technology. The M technology necessitated only 1/3 of hardware capacity that were required by Ingres system (Table 4 and Table 5).

**Table 4. Comparison of response time between Ingres and M-technology based systems.**  
(Unit/sec)

	Benchmark	Ingres	M	Times faster in M	Average
<b>Simple bulk test</b>					
Count	Query				
	1,000	417.5	3.5	119.2	
		524.7	5.2	99.9	
10,000	Delete	394.0	5.2	75.7	
	Insert	3,444.5	27.2	126.6	
	Update	4,201.5	33.2	126.5	
	Delete	3,223.9	38.5	83.7	<b>105.2</b>
<b>Application test</b>					
	Order entry	3.5	1.2	2.8	
	Radiology accession	3.7	1.0	3.7	
	Code master entry	1.7	1.5	1.1	
	Deletion of Lab test	2.0	1.5	1.3	
	Radiology test list	28.7	4.0	7.1	<b>2.9</b>
					<b>53.9</b>

**Table 5. Comparison of hardware resource between Ingres and M-technology based systems for Radiology, Medical record, Dietetics and Special function Labs.**

		Ingres	M	Rate (%)
<b>System</b>				
Resource	CPU Number	3	1	
	Memory (MB)	384	128	33.33
	Disk (GB)	27.3	10.5	38.46
Utilization	CPU IDLE (%)	33.35	37.60	112.74
	Free Memory (MB)	2.51	3.61	143.82
	Disk Util (%)	51.16	55.75	108.97
No of users	Log-In	107	102	95.33
	DB session	141	12	8.51
<b>Database</b>				
Disk usage (MB)	Engine	710	92	12.96
	Data	675	614	90.96
		x3 capacity necessary		

### 9. Conclusions

The opportunity existed to compare performance of a hospital information system using DBMS between Ingres and M-technology (DTM as DBMS and ISM as programming language). The hardware environment used in both systems was identical.

The following conclusions are made :

1. M-technology required only 1/3 of server capacity to handle approximately same amount of transactions as compared to that of Ingres DBMS.

## 8 The Superiority of M-Technology for the Hospital Information System

2. M-technology showed much better transaction and data processing speed such as 5 times faster in case of GUI and 5 to 100 times faster in case of CUI depending the nature of applications.
3. M-technology minimized system downtime because of less hardware requirement and enabling back-up systems.

### References

- [1] J.R.Scherrer, C.Louis and F.Borst, DIOGENE2, a distributed hospital information system with an emphasis on its medical information content, Yearbook of Med Info. Schattauer, Stuttgart, 1995 p86-95.
- [2] S.G.Tolchin, Overview of an architectural approach to the development of the Johns Hopkins Hospital distributed clinical information system, J Med Syst 10 (1986) 321-328.

## WAN環境でのDTMによる業務システムの構築事例

広瀬清司

Seiji Hirose

サン電子株式会社 経営戦略サービス室

Sun Corporation

はじめに

メインフレームあるいはオフコンからパソコンシステムへの移行を進める事例が、最近になって紹介され始めている。残念なことは、多くの事例が「情報系システム」であるということである。弊社においては、「基幹系システム」の「すべての機能をパソコンで対応するシステム」の構築をめざし、現在その第一歩として「販売管理システム」の構築を実施している。「すべて機能をパソコンで」という新しいコンセプトのコンピュータシステムに対しては、様々な問題が発生するが、特に大きな問題は「データベース」と「ネットワーク」であると認識している。

本稿では、弊社において実施している販売管理システムの「ネットワーク環境の構築」に主眼をおいて報告しているものである。

概要

現在、サン電子株式会社ではオフコンによる基幹業務システム運用を実施している。「ダウンサイジング」と「オープン化」の流れの中で、パソコンでの基幹業務システム構築において、データベースシステムとしてDTMを選択している。ネットワーク環境をカバーするシステムとして米国Banyan社のVINESを選択している。このVINESでは、NetBiosエミュレーション機能を使用してDTMのクライアント/サーバーシステムを成立させている。VINESは、価格面や知名度の点でNetWareに劣点があるものの、特に大規模ネットワークシステムにおける「管理の容易さ」「拡張性」「他のネットワークシステムとの親和性」等、他の製品にはないメリットも多数存在することが確認されており、DTMに多くのアドバンテージを与えるものであるとすることができる。

### 1. ダウンサイジングシステムへの背景

#### a. ピラミッド型オフコンシステムの限界

弊社で現在使用しているオフコンシステムは、データベース管理、ネットワーク管理、アプリケーション開発支援、等のすべての機能に対して1台のオフコンで対応しており、十分なパフォーマンスを得られない状況になっている。この状況の問題点は、(1)アプリケーション開発言語(COBOL)の生産性の低さ、(2)データベースシステムの変化(変更)に対する硬直性、そして(3)独自技術によるネットワークシステムの提供、が

考えられる。このいずれもが現在のような変化の激しい状況に対応するには、効力を発揮できないばかりか、システム全体のパフォーマンスを低下させる要因となってしまうている。

#### b. ネットワーク型パソコンシステムへの羨望

上記のようなオフコンシステムの問題点を解決するための手段としては、(1) 高生産性プログラム言語 (M) の採用、(2) 柔軟なデータベースシステム (DTM) の採用、そして、(3) オープンなネットワークシステムの採用、がキーファクターであると判断している。特に、パソコンネットワークシステム (LAN) においては、ファイル共有、プリンタ共有による転送データ量の増加、また電子メールシステムの活用による転送データ量の増加を十分考慮する必要がある。さらに弊社では営業所が各地に散在することから、各営業所間を接続するための LAN 間接続 (WAN 接続) 機能も十分整備され、パフォーマンスを低下させない工夫と高いメンテナンス性が確保されている必要がある。

このような様々なシステム要件をカバーするためのネットワークシステムとして米国 Banyan 社の VINES を選択している。

## 2. ネットワーク OS の検討内容

弊社では、システム全体 (ネットワーク部分、データベース部分、ユーザインターフェース部分) をパソコンのみで実現することを前提としていたので、検討対象として (1)NOVELL 社 NetWare、(2)Banyan 社 VINES の 2 つに絞り検討を行った。

(表 1) ネットワーク OS 検討結果

検討項目	NetWare Ver3.12	VINES Ver 5
価格	安価	高価
シェア	フロア LAN...高い WAN ...不明	フロア LAN...不明 WAN ...高い
処理速度 (ファイルサーバ)	高速	NetWare には劣る
WAN 接続	ルータが必須	ルータは不要
システム管理	サーバー数に比例して困難になる	容易(サーバー数が増加しても変化なし)
知名度(ユーザー数)	圧倒的多数	少数
他システムとの 接続性	各種ツールを選択し組み合わせて 実現する	VINES が統合環境を提供

VINES は、大規模ネットワークシステムにおいて大きなシェアを持つと報告されている。(1サーバー当たり 50 ユーザー以上のネットワークシステムでのシェア: 50% 1992年 Dataquest 社) また、実際に VINES を利用している代表的なユーザー

として、New York Telephone、U. S. Marine Corps、Intelがあげられることから、VINES実力は高いレベルにあるものであると判断している。なお、ネットワークシステムにおいて重要な項目として「セキュリティ」があげられるが、VINESは「C2」レベルをクリアしている。

### 3. DTMのVINESでの動作環境

上記で確認したように非常に高い機能・性能をもつVINESであるが、DTMがどのように動作し、実際にどのようなパフォーマンスを発揮できるのか。あるいは、メリット・デメリットとしてどのような事があるのかを下記に検討している。

DTMは、VINESのNet Biosエミュレーション機能を利用して、ネットワーク上にDTMのクライアント/サーバーシステムを構築することができる。VINESのリダイレクタ・ソフト（19K Byte）、クライアント制御ソフト（50K Byte）は大きく、DOSのコンベンション・メモリを圧迫するが、Net Biosエミュレーション・ソフト（32K Byte）をコンベンション・メモリ上にロードすることにより更にコンベンション・メモリ圧迫する。このため、パソコンの機種によっては、CD-ROM等の周辺装置の接続ができない場合が発生する。弊社では、コンベンション・メモリを少しでも大きく開ける目的でQEMM（メモリ・マネージャ）を使用しているが、QEMM導入により動作が不安定になるケースも報告されており、目的に応じた機種・周辺装置の選択が必要になる。

また、通常Net Biosは各クライアントごとに254セッションまで設定できるようにされているが、VINESの場合は100セッションまでという制限も付けられている。つまり、1DTMサーバは、100クライアントまでしかサポートできないということである。

ただし、通常のNet Biosはすべてのマシンに対して必ずブロードキャストを送信してしまうために、特にWAN環境でのレスポンスを低下させてしまうが、VINESのNet Biosエミュレーション機能では、WAN側のマシンに対してはVINESのネットワーク管理機能での判断が働くために、不必要なブロードキャストを流さない工夫がされているので、WANでのデータ転送効率が高い。

(表2) WAN環境での速度について

使用回線	NTT ISDN 64 (64K Bit/秒)
転送データ件数	1965件
転送データ容量	93.8K Byte
転送時間 (平均値/5回)	36秒
実転送レート	20K Bit/秒

上記のような転送レートに対して、弊社では、DTMのデータサイズを1K Byte、または、

## 12 WAN環境でのDTMによる業務システムの構築事例

2 K Byte としていることからデータ転送に対する負荷を非常に小さくしている。このような転送レートを確保できるにしても、転送対象とするのはデータのみとし、DTMのシステム部分、プログラム部分は各端末に近いサーバに置きWANでの転送対象としない工夫をしなければ十分なシステム性能が発揮できない事を理解する必要がある。

### 4. 考察

以上のような内容から、ネットワークOSとしてVINESを採用する場合のメリットとして下記の内容があげられる。

(表3) VINESのメリット

項目 (内容)	理由	メリット
ルータが不要	VINESサーバがルーティング制御を実施するため	ルータのコストが不要
WAN接続でのルーティングの設計が不要	VINESサーバ同士が相互に認識しあうため	ネットワーク担当者に対する負荷の軽減
WANでの高速性	VINESサーバでのデータ圧縮機能、RIP/SAP コマンドの送出制御機能による	DTM処理のレスポンスの向上
電子メールとの連動が可能	VINES自身がメールエンジンを持っているため	(1)電子メールシステムを別個に用意しなくてよい (2)DTMからのメッセージを電子メールに乗せることが可能である
ネットワーク上のすべての資源活用ができる	VINESのディレクトリサービス機能による	どのクライアントマシンからでも同じ環境を利用できる。

弊社では、「VINESを使用することによりDTMの業務プログラム開発に専念できる」という評価をしている。つまり、VINESとDTMは相互に強調して動作するが相互に依存しないシステムであり、この事によって、開発担当者側も責任分担ができ（ネットワーク責任者、アプリケーション責任者）、各担当部分に専念できるということである。また、弊社ではVINESを導入して1年程度経過しているが、VINESサーバはハードウェア障害以外で停止したことはなく、この事からもDTMで作成した業務処理を実行させるインフラとしての要件を十分に満足していると言うことができる。

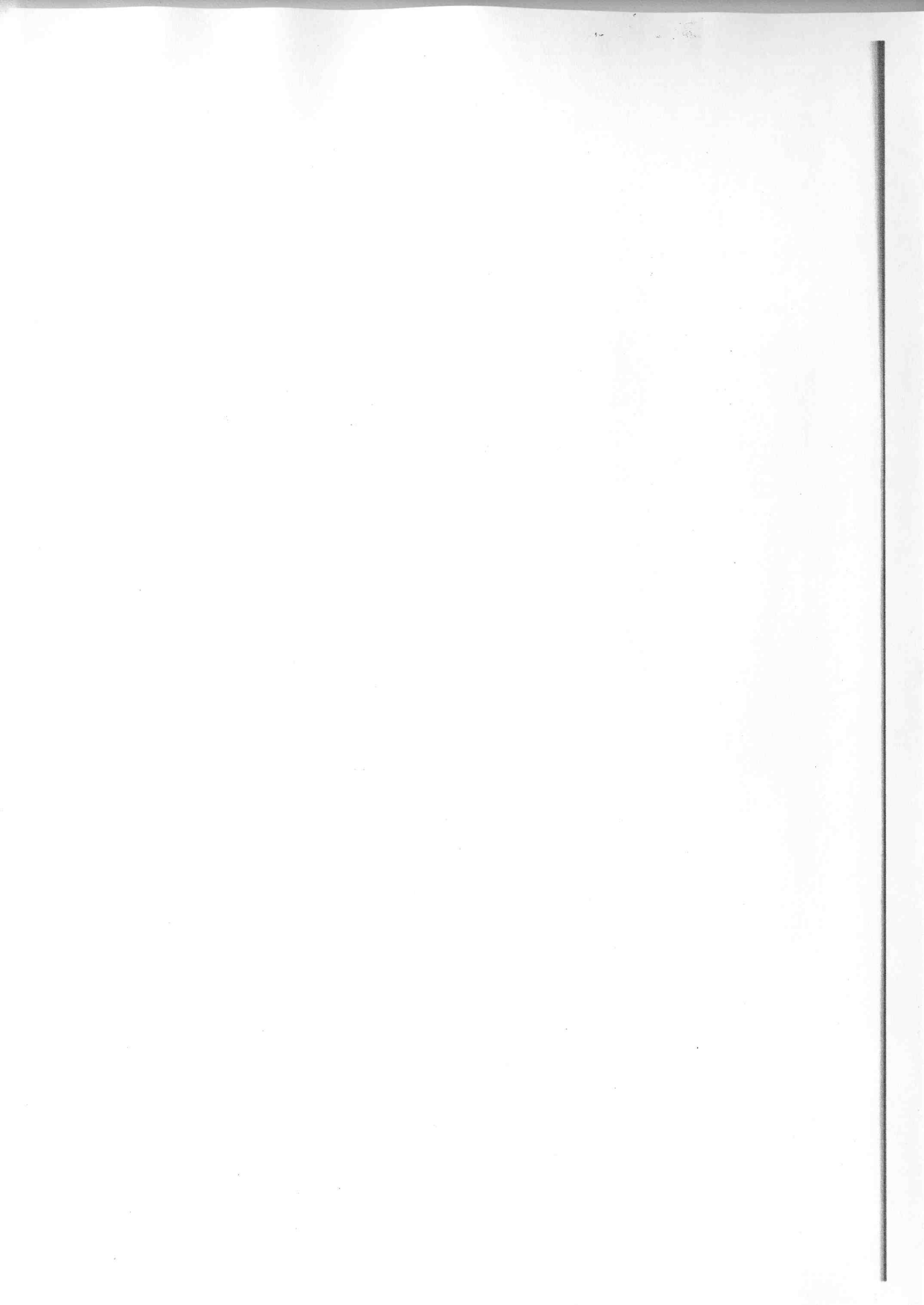
また、将来的にDBサーバーをUNIXマシンに変更する必要性が生じた場合でも、VINESのオプション機能によって容易にUNIXマシンを接続・運用することが可能である。また、NetWare上の資源活用、メッセージ交換、ネットワークの統合、等もVINESのオプション機能を導入することにより可能になるので非常に柔軟性の高いシステム



構築が可能になる。

#### 5. 将来構想

現在進めている販売管理システムの再構築を完了した後、順次生産管理システム、人事管理システムの開発へと進めていく計画である。また、VINESの持つ電子メールエンジンとオプションのSMTP機能を接続しInternetでのメッセージ交換による「発注業務」「納品通知業務」「顧客からの問い合わせ業務」を構築する計画である。このようなメッセージ交換システムを構築することによって、現時点では表面化していないDTMの潜在的な能力を引き出せるのではないかと考えられる。



# DSM/CGI プログラミング

## DSM/CGI programming

今井敏雄<sup>1</sup>  
Toshio Imai

This paper reports a CGI programming sample written in DSM in order to show that M applications can communicate to a Web server relatively easily instead of Perl, Tcl and Shell script generally used for CGI. The sample includes some code to read CGI environment variables and FORM data into M local variables and directly write back HTML text using the M Write command. I believe this exploding Internet/Intranet should be recognized as an opportunity for M technology to shed renewed light on virtues of M application/database by connecting M legacy applications to Web servers. Hopefully Web browser will play a miraculous role for M to complement M's imperfection on GUI/image/AV handling.

(Keywords : MUMPS, M, Internet, WWW, CGI, DSM)

## 1 はじめに

クライアント/サーバ(C/S)環境の構築に際し、クライアントとサーバのアプリケーション分割、PCソフトのバージョン管理と費用、ミドルウェアの選択、ネットワークの性能/信頼性、漢字コード変換等々の問題で苦勞を重ねているおり、世界中の情報をいとも簡単にアクセスし、しかも、OS/プラットフォームに独立で、かつ廉価な Web ブラウザは、新鮮な何かをもって我々に訴えてくる。今後、インターネットは、社内の情報インフラとしてのイントラネットとして、その利用はますます増加すると予想される [1]。そこでは、Web ブラウザは、既存業務アプリケーション(legacy application)の C/S の Fat Client の問題を解決する、或る意味でメンテナンスフリーの統合クライアント端末として、利用されると思われる。

M にとって、インターネット/イントラネットは、M が何が出来るかを外に知らせうるチャンスとして捉える必要がある。各所で、Legacy application のイントラネット統合による社内/外への付加価値サービスが試みられようとしている中 [2]、M も、それが簡単に可能なのだという事を実証し、M アプリケーション/M データベースの価値を、インターネット上でアピールしなければならない。

以下、Web サーバと外部プログラムとの汎用インタフェースである CGI(Common Gateway Interface) [3] を用いて、DSM プログラムで、直接、Web サーバに、HTML を出力するサンプルプログラムを試みたので報告する。目的は、CGI で通常使う Perl、Tcl やシェルスクリプトなどを使わなくても、比較的簡単に M だけで、CGI プログラミングが出来ることを示す事です。M だけで、CGI プログラムが書けるという事は、M ユーザにとっては、自分たちの土俵内で出来ることであり、朗報だと思います。Web サーバと M の統合をうまく行なえば、Web ブラウザが、クライアント端末側の面倒を見てくれる為、Web は案外、M 言語/M Technology の GUI やイメージ/AV ハンドリングの貧弱さをカバーしてくれる救世主となるかも知れません。

## 2 Web サーバ環境

試行 Web サーバ環境は、

Open VMS V5.5-2  
DEC TCP/IP Service for OpenVMS V3.2J  
DSM for OpenVMS V6.2J

Web サーバ(HTTPD)はフリーウェアの Ohio State University (David I. Jones 氏)

<sup>1</sup> 日本デジタルイクイップメント (株)  
西日本第一統合システム部  
〒530 大阪市北区中之島 2 丁目 2 番 2 号  
ニチメンビル  
TEL(06)222-9211 FAX(06)222-9408

Digital Equipment Corporation Japan  
Network and Systems Integration Services  
Nichimen Building, 2-2-2, Nakanoshima, Kita-ku,  
Osaka 530, Japan  
TEL(06)222-9211 FAX(06)222-9408

DECthreads HTTP server for VMS V1.9 [4]

を使用しました(以下、OSU Web サーバと呼ぶ)。

この Web サーバの構造を 図1 に示す。

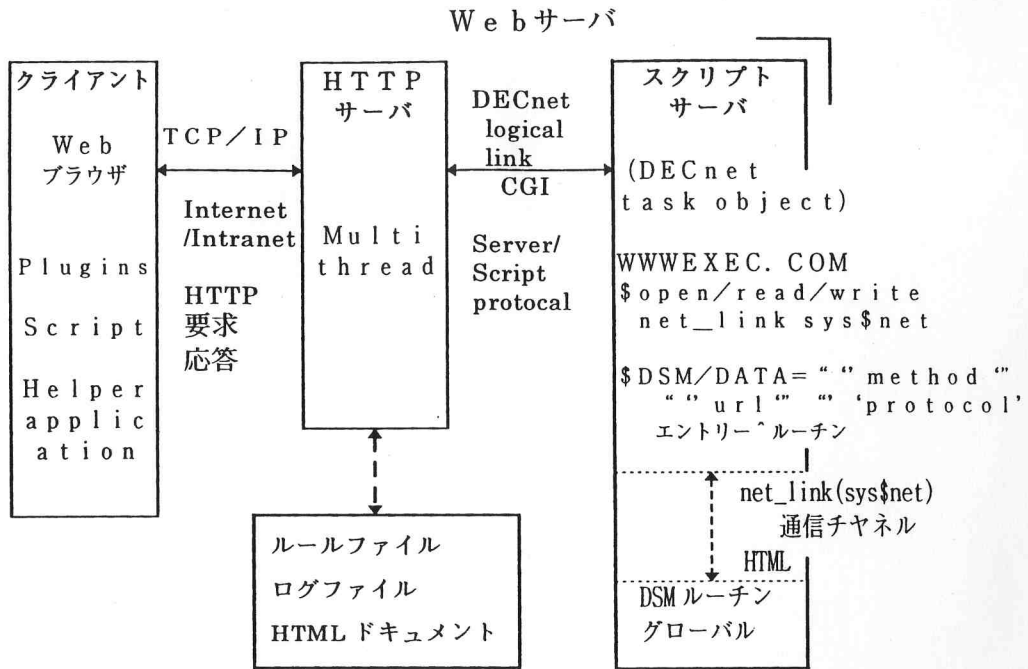


図1 OSU Web サーバの構成

Unix 環境では通常 Fork で CGI 実行プロセス環境を生成するが、VMS ではプロセス起動が比較的重たい為、この Web サーバの場合、あらかじめ指定したプロセス数のスクリプトサーバを起動しておき、HTTP サーバとの間で DECnet タスク間通信を用いて CGI のやり取りを行います。

### 3 DSM/CGI

CGI は、Web ブラウザからの要求に応じて、外部プログラムを起動する汎用インタフェースで、Web ブラウザに返す HTML(HyperText Makeup Language)テキストをダイナミックに生成するメカニズムを提供する。これを利用して、既存 M アプリケーションの情報をインターネットに発信することが可能となる。CGI を起動する HTML の書き方は、OSU Web サーバの場合、単にプログラム起動をするだけの時、htbin ディレクトリの CGI プログラム名を指定した<A>タグを用いる。CGI プログラム名は、或るディレクトリに格納された、\*.COM の DCL(Digital Command Language)コマンド名、または、\*.EXE の実行イメージ名です。

```
<A HREF="/htbin/DSM_CGI002_POST"> POST メソッドテスト</A>
```

Web サーバから、CGI プログラムに情報(データ)を渡す場合、<FORM>タグを用いる。

```
<FORM METHOD=POST ACTION="/htbin/DSM_CGI002_SUBMIT">
...
</FORM>
```

スクリプトサーバで DSM を起動する最も簡単な DCL コマンドは以下のとおりです。

```

$! DSM_CGI002_POST.COM
$! P1:method P2:url P3:protocol
$ DSM/ENV=IMAI/UCI=IMA /ERR=SYS$LOGIN:DSMCGI002_POST.ERR—
    /DATA=""P1"" ""P2"" ""P3"" POST^CGI002
$ EXIT

```

起動された CGI プログラム(DSM ルーチン)は、通常、次のステップをとる。

1. Web ブラウザから FORM タグで送られてきたデータや、CGI 環境変数をプログラム内変数に取り込む。この時、URL デコーディングと漢字コード変換を行う。
2. データベースアクセス等の個別プログラム処理を行う。
3. Web ブラウザに HTML テキストを出力する。この時、インターネット上の通信メッセージ文字セット規約 ISO-2022-JP [5] [6] に従って出力する。

OSU Web サーバの場合、HTTP サーバとスクリプトサーバ(DSM ルーチン)との間で情報交換を行う Server/Script プロトコル(<DNET...>コマンド形式)が定められている。これを用いて、DSM ルーチンは、FORM データと CGI 環境変数情報を取得する。例えば、環境変数 SCRIPT\_PATH を取得する場合、以下のコマンドで可能です。

```

U "net_link" W "<DNETPATH>","!
R A S ENV("SCRIPT_PATH")=$P(A," ")

```

HTTP ヘッダー情報は、次のコマンドで取得可能です。

```

W "<DNETHDR>","! F I=1:1 R A Q:A="" D
.S DNET("DNETHDR",I)=A
.I A?1"Content-type: "1.E S ENV("CONTENT_TYPE")=$P(A,":",2) Q
.I A?1"Content-length: "1.N S ENV("CONTENT_LENGTH")+=$P(A,":",2) Q
.I A?1"Accept: "1.E S ENV("HTTP_ACCEPT")=$P(A,":",2) Q
.I A?1"User-Agent: "1.E S ENV("HTTP_USER_AGENT")=$P(A,":",2) Q
S DNET("DNETHDR")=I-1

```

GET メソッドの QUERY\_STRING や、POST メソッドの Content データは、URL エンコーディングと言われる 7 ビット ASCII 印字可能文字列に変換されて送られて来る。URL デコード関数の DSM コーディング例を以下に示す。

```

URLDECODE(INP,TAIL,ERR); URL デコーディング関数
; "+"-> 1 space, %xx(16進)-> 1 Code(非英数 ASCII)
; Input : INP = 変換対象文字列
; Outout: 関数値 = 変換後文字列
; TAIL= %xx で変換出来なかった部分文字列 null or % or %x
; ERR=0:%xx 変換エラーなし
; 1:%xx 変換エラー=xx が 16 進数字でない %xx をそのまま、関数値に返す
NEW A,B,C,P,N,D
S TAIL="",ERR=0,A=$TR(INP,"+"," "),B="",C=1
; P は A の中の C 文字目以降の%位置+1(%なし時 0)
F S P=$F(A,"%",C) Q:P D
.S N=$E(A,P,P+1); %xx の xx
.I $L(N)<2 S A=$E(A,1,P-2),TAIL="%_N Q ; 接尾文字列が%xx の途中で切れる場合、
; %以降を TAIL に返す
.S D=$$HD(N); 16 進->10 進変換(エラーは*)
.I D="" S ERR=1,B=B_$E(A,C,P-1),C=P Q ; 変換エラー時、%.. のまま返す
.S B=B_$E(A,C,P-2)_C(D),C=P+2
Q B_$E(A,C,$L(A))

```

URL デコード後、漢字コード変換を行う。試行環境の場合、SuperDEC 漢字コードへの変換を行う。漢字変換関数の DSM コーディング例を以下に示す。

```

IJP2SDEC(INP,KMODE,TAIL,ERR);ISO2022-JP(JIS) から SuperDEC コード系への変換関数
; Input: INP = 変換対象文字列
; KMODE = 変換最初の漢字モード 0:ASCII/Roman 1:漢字 2:半角カナ
; Output: 関数値 = 変換後文字列
; KMODE = 変換終了時の漢字モード
; TAIL = 変換出来なかった接尾文字列
; ERR = 0: 変換エラーなし 1: 変換エラー=漢字文字列が奇数バイトのものがある
NEW A,B,P,C,E,J,STS
S STS=%&%KJOFF
S KMODE=%G(KMODE,0),TAIL="",ERR=0,A=INP,B="",C=1; P は A 中の C 文字以降の ESCAPE 位置+1
F S P=%F(A,%C(27),C) Q:'P D ; ESC=%C(27)検出
.D CV(C,P-2) ; C から P-2(ESCの手前)までの文字列確定->変換 ConVert
.S E=%E(A,P,P+1) ; ESCxx の xx の部分
.I E="(B"!(E="(J" S KMODE=0,C=P+2 Q
.I E="%B"!(E="%@" S KMODE=1,C=P+2 Q
.I E="(I" S KMODE=2,C=P+2 Q ; 半角カナ規定なし
.S B=B_%C(27),C=P ; JIS エスケープ以外の ESC
.I KMODE=1,$L(A)-C#2=0 D ; 漢字奇数バイト
.D CV(C,$L(A)-1) S TAIL=%E(A,$L(A)) ; 文字列末尾 1 バイト目
E D CV(C,$L(A))
I STS S STS=%&%KJON
Q B
;
CV(S,E); A の S 文字目から E 文字目までを B に結合 漢字モードの時、結合時コード変換を行う
I E<S Q
I KMODE=0 S B=B_%E(A,S,E) Q ;ASCII/Roman モード
I KMODE=2 D Q ;半角カナモード
.F J=S:1:E S B=B_%C(142,$A(A,J)+128) ;%C(142)=SS2
; 漢字モード
I E-S#2=0 S ERR=1 ;漢字で奇数バイトエラー
F J=S:1:E S B=B_%C($A(A,J)+128) ; 8 bit off 漢字 -> 8bit on 漢字
Q

```

URL デコード、漢字コード変換で注意する事は、DSM ローカル変数に、QUERY\_STRING や Content データを取り込む際、ローカル変数の文字列長の制限の為、データ中の%nn の途中や、変換前の漢字 2 バイトの途中で、文字列が分断されるケースを考慮する事です。CGI 環境変数や、Web サーバからのデータが、DSM ローカル変数へ取り込めれば、後は、必要な処理を行い、HTML を出力する。DSM の場合、デバイスに、漢字 ON/OFF のエスケープシーケンス文字列を指定し、JIS コード系で出力する機能がある。これを利用すれば、ISO-2022-JP での出力は、

```
U "net_link":(KANJI:KCODE="JIS":KON=%C(27)_"$B":KOFF=%C(27)_"(J")
```

を指定するだけで、通常の Write コマンドで、HTML を出力すればよい。

## 4 サンプルプログラム

漢字入力を含む複数行テキスト入力、表示したい DSM ローカル変数名を選択するボタン、ローカル変数をグローバルにコピー(MERGE)する場合、どのグローバルにコピーするかのグローバル選択プルダウンメニュー等を含む FORM の HTML を出力する DSM プログラムを 図 2 に示す。このプログラムの Web ブラウザ出力、データ入力時の画面を 図 3 に示す。SUBMIT 後の、DSM に取り込んだローカル変数値を Web ブラウザに書き戻した画面出力を 図 4 に示す。DSM に取り込んだローカル変数は以下のとおりです。

1. 変数名 ENV ----- CGI 環境変数を保持する変数。
2. 変数名 FORM --- FORM データ(テキスト、チェックボックス、メニュー選択)を保持する変数。  
FORM の POST Content データ長には制約がないため、  
FORM(name)=maxseq,  
FORM(name,seq)=value  
の形式でセットする。複数行テキスト入力の改行(CR,LF)、または、レコード(value)があるバイト長に達した時、次の seq へ飛ぶ。
3. 変数名 DNET --- HTTP サーバとスクリプトサーバ(DSM)との通信データを保持する変数。

```

CGI002 ; DSM/CGI Programming -- Sample program 2 (FORM Test)
; Copyright (c) Digital Equipment Corporation-Japan 1996. All rights reserved.
; Coded by T.Imai at 4/5/96
; DSM/CGI テスト 2 -- FORM METHOD=GET or POST
Q
;
GET ; Entry 1 Method=GET
; FORM=GET HTML--入力画面(DSM/CGI 生成)
; Called from [.BIN]DSM_CGI002_GET.COM
K (%)
S method="GET"
D WRTHTML1
Q
;
POST ; Entry 2 method=POST
; Called from [.BIN]DSM_CGI002_POST.COM
; FORM=FOEM HTML--入力画面(DSM/CGI 生成)
K (%)
S method="POST"
D WRTHTML1
Q
;
WRTHTML1 ; HTML 書き戻し
; Called from GET or POST
; Input : method = "GET" or "POST"
; DEcnet object WWWEXEC.COM の中で、sys$net(タスク間通信チャネル)が
; Open され、論理名 net_link が定義されている。
S link="net_link",crlf=$C(13,10)
O link U link:NOKAN;最初は、NOKANJI モードで Script Server とやり取り
W "<DNETCGI>";
; MIME charset name=ISO-2022JP(JIS) RFC1468 を利用
W "Content-type: text/html; charset=ISO-2022-JP",crlf,crlf;CGI header
; Context-type の後ろに、空行が必要
U link:(KAN:KCODE="JIS":KON=$C(27)_"$B":KOFF=$C(27)_"(J)")
W "<HTML><HEAD><TITLE>DSM/CGI テスト</TITLE></HEAD><BODY>";!
W "<IMG SRC=""/www/digital-logo.gif"" ALIGN=BOTTOM ALT=""digital"">"
W "<A HREF=""/www/dsm_cgi_home.html""><FONT SIZE=+1> DSM/CGI テスト ホームページへ</FONT></A><P>";!
W "<H3><FONT SIZE=+2>DSM/CGI FORM METHOD=<B></FONT><FONT SIZE=+3>".method,"</FONT></B>"
W "、日本語入出力、CGI 環境変数取得 テスト</H3>";!
; method は、実行時に決まる (GET or POST)
W "<FORM METHOD=""_method_" ACTION=""/htbin/DSM_CGI002_SUBMIT"">"
W "<OL>"
W "<LI>漢字入力 1 : <INPUT TYPE=TEXT NAME=""input1"">"
W "<P></P>";!
W "<LI>漢字入力 2 : <TEXTAREA NAME=""input2"" ROWS=4 COLS=60></TEXTAREA>"
W "<P></P>";!
W "<LI>ローカル変数 表示項目の選択 : "
W " <INPUT TYPE=CHECKBOX NAME=""ENVdisp"" VALUE=""yes"" CHECKED>ENV-CGI 環境変数の表示 "
W " <INPUT TYPE=CHECKBOX NAME=""FORMdisp"" VALUE=""yes"" CHECKED>FORM-FORM 変数の表示 "
W " <INPUT TYPE=CHECKBOX NAME=""DNETdisp"" VALUE=""yes"">DNET-通信用変数の表示 "
W "<P></P>";!
W "<LI>ローカル変数 ENV,DNT,FORM をグローバルにセーブしますか ?"
W " <INPUT TYPE=CHECKBOX NAME=""gblsave"" VALUE=""yes"">はい</P>";!
W "<P><CODE>M実行コード : KILL ^... MERGE ^...("ENV")=ENV, ^...("FORM")=FORM, ^...("DNET")=DNET </CODE></P>"
W "<P>セーブするグローバル名...の選択:</P>"
W "<P><SELECT NAME=""gblname"" SIZE=3>"
W " <OPTION>^CGISAVE"
W " <OPTION>^cgisave"
W " <OPTION>^cgisave(ENV("REMORT_HOST"))"
W " <OPTION>^TEMP"
W " <OPTION>^temp"
W " <OPTION>^temp(ENV("REMORT_HOST"))"
W " </SELECT></P>";!
W "<OL>"
W "<P><INPUT TYPE=SUBMIT VALUE=""実行開始"">"
W " <INPUT TYPE=RESET VALUE=""値リセット""></P>";!
W "</FORM></BODY></HTML>";!
W "<DNETCGI>";!
U link:NOKAN C link
Q

```

図2 サンプルプログラム

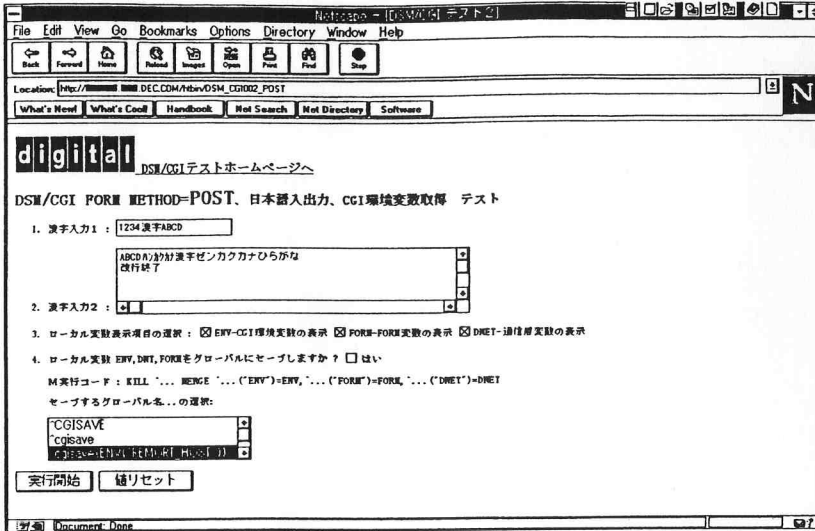


図3 FORM入力画面

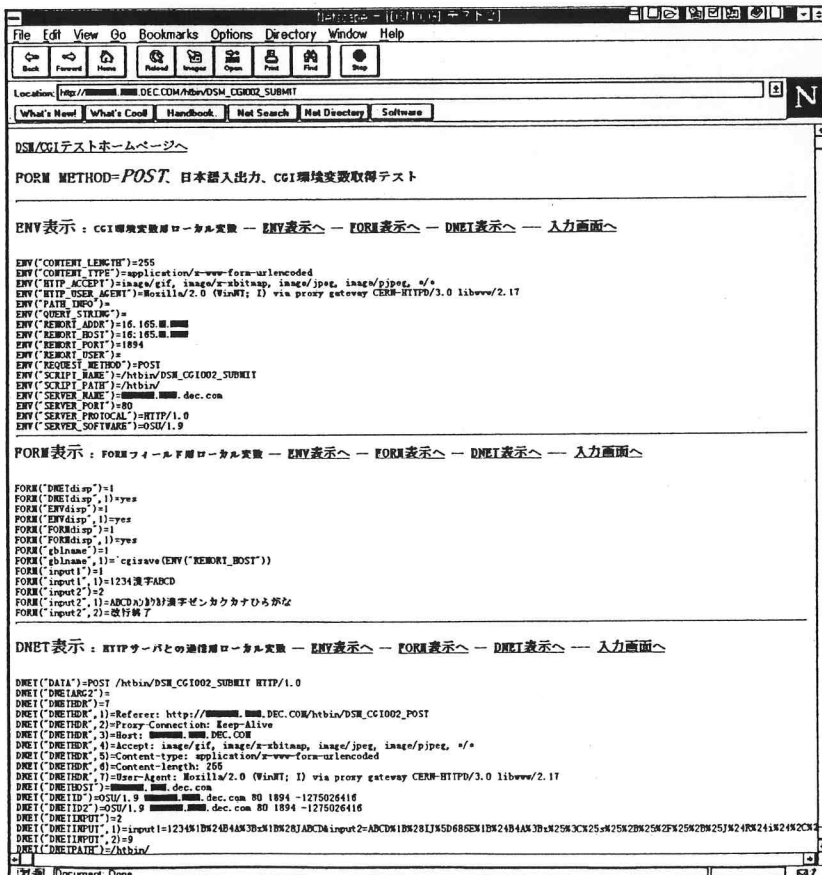


図4 ローカル変数表示出力画面



## 5 おわりに

CGI プログラムを書く場合、HTML の知識が必要です。しかし M のユーザにとってはそれほど困難でないと思います。しかし、メンテナンス上、M プログラムの中に、陽に HTML を書くべきかどうかは別の問題です。HTML コンバータや、HTML を隠した開発ツールなど、今後の検討課題です。

CGI はサーバのステータスレス、セッションが毎回切れるという Web の特徴を引きずっている(ある意味ではそのシンプルさが利点)。参照系からより業務アプリケーションに近い所で Web が利用される場合、CGI のコンテキスト保持、性能の問題が今後の課題です。これらの問題解決や、Web の応用範囲を広げる為に、今、インターネットは、急速に発展している技術です。Netscape、Microsoft、それぞれの Web サーバに固有で非標準 (Propriety) な仕様ですが、Web サーバベースの API(NSAPI,ISAPI)が提案されている [7]。さらに、Java/JavaScript、ActiveX/VB Script はコード移動(Mobile Code)という考えをもった、いわば、Application on demand という形で、HTML の世界を広げようとしている。M がどのようにインターネット技術に追随しえるか、M の大きな課題です。

## 文献

- [ 1 ] BusinessWeek : HERE COMES THE INTRANET  
(<http://www.businessweek.com/1996/09/b34641.htm>)
- [ 2 ] Vance McCarthy : The Web:OPEN FOR BUSINESS, Datamation, Dec 1,1995
- [ 3 ] The Common Gateway Interface (<http://hoohoo.ncsa.uiuc.edu/cgi/>)
- [ 4 ] VMS freeware from the WKU FILESERV の packages の HTTP\_SERVER  
([http://www.wku.edu/htbin/fileserv?HTTP\\_SERVER](http://www.wku.edu/htbin/fileserv?HTTP_SERVER))
- [ 5 ] Network working group RFC 1468 Japanese Character Encoding for Internet Messages,June,1993
- [ 6 ] Takada Toshihiro : Japanese Encoding Methods  
(<http://www.ntt.jp/japan/note-on-JP/encoding.html>)
- [ 7 ] D.Plummer:Web Server APIs:Challenging the Ubiquitous CGI,GartnerGroup Research Note, May 29,1996



# DSM/DASL テスト・ターミナル・エミレータ

## DSM/DASL Test Terminal Emulator

今井敏雄<sup>1</sup>  
Toshio Imai

A test terminal emulator has been developed in order to carry out an efficient and automated performance test on DSM/DASL online applications. The learning function of this tool gathers an operator's key input sequence into a sequential file, which is used later as an input file at a run time emulation phase through a file re-direction capability of the tool. This tool also has a script language to control the emulation runtime environments including operator's thinking time, loop execution control and custom MUMPS code execution. Using this tool, we could construct a lot of well-managed and reproducible test environments for the DSM/DASL applications.  
(Keywords : MUMPS, M, TEST TOOL, DASL, DSM)

### 1 はじめに

オンライン・システムの性能評価や負荷試験を行う場合、特に多端末のオペレーション環境をいかに作り出すかが問題です。数十、数百の端末を人海作戦でこなすことは、その労力のわりには、試験の再現性、本番環境との隔離(応々にして、過負荷となる)等の問題があり、非効率です。ある大規模な DSM/DASL ユーザで [1]、システムの性能評価/負荷試験を実施する必要があり、その為のツールとして、テスト・ターミナル・エミレータを開発した。ツールの目的は、DSM/DASL アプリケーションをほとんど変更することなく、かつ、端末オペレータを介さずに多端末試験の自動化環境を作ることです。業務オペレーションは、ツールの学習機能で、入力シーケンスをファイルに落とし、試験時に、端末入力のファイル・リダイレクション機能を用いて、そのエミレーションを行う。オペレータの Thinking time、連続ループ実行、シーケンスチェック、端末同期等々の試験環境を制御する為、一種のスクリプト言語を用意した。このツールを用いることにより、様々に管理された再現性のある自動化試験環境を作り出す事が出来る。このツールは、DASL(DSM Application Software Library)向けに開発されたもので、スクリーンの再コンパイルで機能拡張を図れる様設計されている。しかし、通常の DSM プログラムについても、端末 Read コマンドを一定の簡単なルールで変更を行うことにより同機能を付加する事が出来る。

### 2 目標と方針

ツールの目標は以下のとおりです。

1. 既存および今後作成される DSM/DASL アプリケーション・プログラムの変更を(極力)行わなくても利用可能な自動化テスト・ツールを提供する。
2. テスト環境は、通常のアプリケーション運用環境に近い状況を模擬出来るようにする。
3. 端末入力部のオペレーションを模擬(エミレート)自動化し、端末オペレータの介在なくテストが出来るようにする。
4. 回帰テスト(Regression test)、多端末テスト、および負荷環境でのテストを効率的に行えるようにする。

<sup>1</sup> 日本デジタルイクイップメント (株)  
西日本第一統合システム部  
〒530 大阪市北区中之島2丁目2番2号  
ニチメンビル  
TEL(06)222-9211 FAX(06)222-9408

Digital Equipment Corporation Japan  
Network and Systems Integration Services  
Nichimen Building, 2-2-2, Nakanoshima, Kita-ku,  
Osaka 530, Japan  
TEL(06)222-9211 FAX(06)222-9408

<sup>2</sup> このツールは DASL の標準外機能です。

この目標を実現するため、端末入力をシーケンシャル・ファイルからの入力へリダイレクトすることにより、端末入力をエミレートする方式を採用した。DASLは一種の4GLで、DASLがコンパイル生成するスクリーン・ルーチン内で、入出力処理が実行される。したがって、DASLアプリケーションを変更することなく、エミレーション機能を追加することが可能となる。

端末入力ツール内で制御可能なことより、アプリケーションから、端末入力部に制御が渡るタイミングで、キー・ストロークの学習や、入力装置のファイル・リダイレクト処理等の特殊処理を組み込む切り口が与えられる。この切り口を利用して、目標の実現を図る。通常、この切り口だけでは、アプリケーションの実行フロー自身は変えられません。しかし、エミレータは、DSM/DASLアプリケーションと同じプロセス・コンテキスト内で実行する為、その入力切り口で、エミレータの制御、アプリケーションの状態チェック、入力値の変更、カスタマイズMコマンドの実行等の処理が行える。そのための、エミレータへの指示(スクリプト)言語を用意する。このことにより、柔軟性のあるDSM/DASLテスト環境を構築することが可能となる。

### 3 機能

テスト・ターミナル・エミレータは、主に以下の3つの機能からなる。

- a) ターミナル・キー・ストローク学習機能
- b) ターミナル・エミレーション実行機能
- c) エミレーション・ログ・プレイバック機能

作業フェーズは、以下の4つからなる。

1. 学習フェーズ : オペレータ入力シーケンスを学習する。
2. スクリプト編集フェーズ : 学習シーケンシャル・ファイルにエミレータ制御用コマンドを組み込む。
3. テストフェーズ : エミレーション試験実行を行う。
4. プレイバックフェーズ : エラーチェック等の為、画面入出力パートのみ再現表示する。

ターミナル・エミレータの基本構成、作業フェーズを図1に示す。

### 4 スクリプト言語

エミレータの実行制御を行うスクリプト言語を以下に示す。

スクリプト構文:

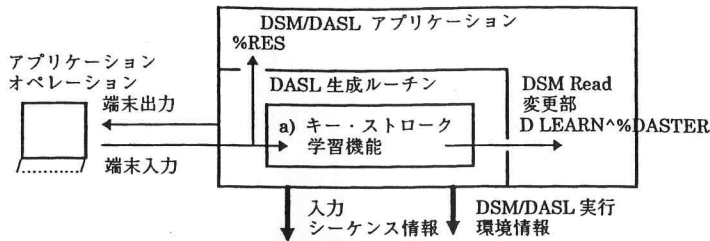
凡例 ::= は定義、| は または、[] はオプション、... は繰り返し

スクリプトファイル ::= レコード値[レコード値...]  
レコード値 ::= 入力値レコード | 宣言レコード  
入力値レコード ::= 入力値[/アクション...];[コメント]  
アクション ::= アクションコマンド[:アクション条件文][アクション引き数]  
宣言レコード ::= #宣言コマンド[:宣言条件文][宣言引き数];[コメント]  
アクション条件文 ::= 真偽値評価 DSM 式  
宣言条件文 ::= 真偽値評価 DSM 式

/アクションコマンド アクション引き数

/ABORT[ エラーメッセージ評価 DSM 式]  
/GOTO ラベル名  
/HANG[ 秒評価 DSM 式]  
/MUMPS DSM コマンド列  
/KEY キー・ニーモニック  
/VALUE 入力値評価 DSM 式

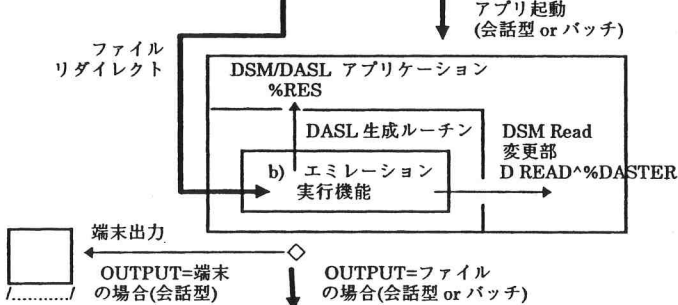
1. 学習フェーズ



2. スクリプト編集フェーズ



3. テストフェーズ



4. プレイバックフェーズ

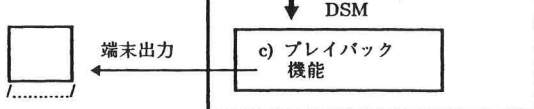


図1 基本構成と作業フェーズ

#宣言コマンド 宣言引き数

- #ABORT[ エラーメッセージ評価 DSM 式]
- #ACTION アクション指示子文字列
- #COMMENT コメント指示子文字列
- #GOTO ラベル名
- #HANG[ 秒評価 DSM 式]
- #LABEL ラベル名
- #LINE レコード行数
- #MUMPS DSM コマンド列
- #SCREEN DASL スクリーン名
- #SCRIPT 宣言指示文字列

## 入力値レコード例 1

06-222-9211/HANG 5

電話番号入力後、5秒間の thinking time を設定する。

## 入力値レコード例 2

AA/MUMPS D ^XXXX/GOTO:\$D(^STOP) EXIT

入力値に AA をセットし、M コマンド D ^XXXX を実行し、グローバル ^STOP が定義されていれば、スク립ト ラベル EXIT に入力シーケンスを移す。未定義の場合は、次レコードが次の読み込みとなる。

## 宣言レコード例

```
#mumps za ^SYNC($J) s kaisuu=0
#LABEL ODEMO_1
#SCREEN ODEMO
...
#m s kaisuu=kaisuu+1 h:$d(^STOP)
#goto:kaisuu<10 ODEMO_1
#LABEL ODEMO_2
```

入力シーケンスをループさせ、同じオペレーションを連続実行する。この場合、オペレーションを繰り返し実行しても、ループするオペレーション・シーケンスが変わらないように、データベースや内部変数等の環境について配慮する必要がある。

DASL に付録している「デモンストレーション・システム」を用いて、キー・ストロークの学習を行い、その結果、生成されたスク립ト・ファイル例を図 2 に示す。

学習オペレーションは、デモンストレーション・システムの「デモ メニュー ODEMO(オプション・スクリーン)」を、DASL スクリーン起動(DO ^%DAS)し、「住所録 ADBOOK(データ・スクリーン)」と、「スケジュール管理 MEET(データ・スクリーン)」のオペレーションを行なったものです。

## 5 Read コマンド対応

アプリケーションに、DSM Read コマンドを使用している場合、エミレータの学習機能が働きません。その場合は、スク립ト・ファイルをマニュアルで編集し、適切な入力シーケンス部に Read コマンドの入力値を埋め込む必要があります。しかし、以下の変更規則に従って、Read コマンドを変更すれば、エミレータの学習、および、実行機能を働かせる事が出来ます。

アプリケーションの変更規則：

変数 %DASTE 以外の変数名はサンプル例です。

## (1) シンプル Read の場合

R XXX

→

D

.I \$G(%DASTE)="R" D READ^%DASTER(XXX) Q ;エミレータ機能を有効にする

.R XXX

.I \$G(%DASTE)="L" D LEARN^%DASTEL(XXX) ;学習機能を有効にする

```

; DASLテスト・ターミナル・エミレータ 学習スクリプト 4/6/95 16:39:29 生成
; DASL_DEMO.DAT
;
; 直接起動例 $DSM/UCI=../VOL=../INPUT=DASL_DEMO.DAT/DATA="HANG_DEF=1 TRACE"
; 生成起動コマプロ DASL_DEMO.COM の利用
...
WRITE "エミレータ実行時オプション=","", $G(%), """, !
DO INIT^%DASTER($G(%)); エミレータ初期化処理
; 学習開始 16:39:30.27 ;0 ;0
DO ^%DAS
#COMMENT ;
#ACTION /
#SCRIPT #
#LINE 39 ;行番号ベース
#LINE 40
ODEMO ; ;2.8 ;.33
#LABEL ODEMO_1
#SCREEN ODEMO
1 ; ;4.9 ;1.06
#LABEL ADBOOK_1
#SCREEN ADBOOK
* ;NAME ;6.6 ;1.13
/KEY NEXT ; ;8.8 ;1.61
/KEY NEXT ; ;11.9 ;1.88
/KEY PREV ; ;12.9 ;2.13
/KEY DOWN ; ;13.5 ;2.15
/KEY DOWN ; ;13.8 ;2.18
/KEY DOWN ; ;13.9 ;2.19
/KEY DOWN ; ;14.1 ;2.21
/KEY DOWN ; ;14.6 ;2.22
/KEY UP ; ;15.1 ;2.24
1 ;EDQ ;17.6 ;2.33
;EDNAME ;18.3 ;2.36
;ZIP ;18.7 ;2.5
#LINE 60
;STREET ;19.3 ;2.54
ニチメンビル 8 F ;CITY ;31.3 ;2.7
;STATE ;32.3 ;2.74
;BTHDAY ;32.7 ;2.78
;PHHOME ;33.3 ;2.83
06-222-1111 ;PHWORK ;40.7 ;2.9
;PHOTH1 ;41.4 ;2.94
1 ;SEQ ;44.3 ;2.99
;NAME ;45.2 ;3.03
#LABEL ODEMO_2
#SCREEN ODEMO
2 ; ;46.5 ;3.12
#LABEL MEET_1
#SCREEN MEET
/KEY NEXT ;EQ ;49 ;3.68
/KEY PREV ; ;50 ;3.8
1 ; ;51.5 ;3.91
/KEY DOWN ;MDATE ;52.2 ;3.96
/KEY DOWN ; ;52.4 ;4
#LINE 80
/KEY DOWN ; ;52.5 ;4.03
/KEY DOWN ; ;52.7 ;4.07
/KEY DOWN ; ;52.8 ;4.09
/KEY UP ; ;53.2 ;4.11
/KEY UP ; ;53.6 ;4.14
/KEY F10 ; ;54.3 ;4.15
0 ;SEQ ;56.4 ;4.19
#LABEL ODEMO_3
#SCREEN ODEMO
1 ; ;57.8 ;4.3
#LABEL ADBOOK_2
... ;以降はコメント
...
...
DO EXIT^%DASTER; エミレータ終了処理
HALT
$! 学習終了 16:40:49.54 経過時間= 79.3 秒 CPU 時間= 6.11 秒
$!%DTE-END

```

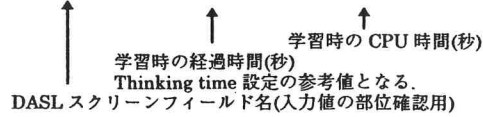


図2 学習スクリプト・ファイル例

(2) エスケープ機能あり Read の場合

```
R YYY S TERM=$ZB#256,FUNC=$ZB¥¥256
```

```
→
```

```
D
.I $G(%DASTE)="R" D READ^%DASTER(.YYY,.TERM,.FUNC) Q
.R YYY S TERM=$ZB#256,FUNC=$ZB¥¥256
.I $G(%DASTE)="L" D LEARN^%DASTEL(.YYY,.TERM,.FUNC)
```

(3) タイムアウト/入力幅指定ありの Read の場合

```
R RES#WIDTH:TIMOUT I S T=$ZB#256,F=$ZB¥¥256
E S T=27,F=41 ; (例) タイムアウト時 F10(Exit)キーと同等扱い
```

```
→
```

```
D
.I $G(%DASTE)="R" D READ^%DASTER(.RES,.T,.F) Q
.R RES#WIDTH:TIMOUT I S T=$ZB#256,F=$ZB¥¥256
.E S T=27,F=41
.I $G(%DASTE)="L" D LEARN^%DASTEL(.RES,.T,.F)
```

制限事項:

- 1) ローカル変数 %DASTE を専有使用する。アプリケーションで同変数が、Kill,New されないこと。
- 2) 1文字 Read(R \*X)はサポートされない。

## 6 おわりに

このツールは、DSM で作られています。M(UMPS)が、ミニコンピュータ上で動いていた黎明期、M 自身が TSS OS であった事と、もともと文字列処理が容易である事から、スクリーンエディタをはじめ、すべての開発支援ツールは M 自身で作っていた。様々なツールが M で容易に作れるということは、M ユーザのパワーアップが自ら可能ということで、そが M のひとつの大きな魅力です。開発支援ツールとしての M をもう一度見直す必要があると思います。

## 文献

- [1] 今井敏雄、佐藤真美 : M と TP モニターの統合の試み, Mumps, Vol 19, 1994



## インターネット技術とDSM/WWW接続ツール

### *Internet technology and DSM/WWW connection tool*

城崎 礼子、杉山 閑照

Reiko Jozaki, Shizuteru Sugiyama

日本デジタルイクイップメント (株) 西日本第一統合システム部  
Digital Equipment Corporation Japan Systems Integration

〒530 大阪市北区中之島2丁目2番2号 ニチメンビル  
TEL(06)222-9211 FAX(06)222-9408

---

We have developed a WWW/DSM interconnecting tool which generates a type of DSM CGI applications efficiently instead of writing a lot of HTML and M codes. This tool is a sort of 4GL tool for HTML M applications and also makes use of an ubiquitous WWW browser as its development environment. This means that the tool itself is a M GUI application using WWW browser as a practical and efficient alternative of M GUI approaches. Based on our experiences of the tool development, we found that M would become a suitable and effective language for WWW connection, mainly due to WWW's openness and simplicity, and M's powerful text processing and dynamic interpreter feature.

---

### はじめに

インターネット技術の中で、Mはどのような位置付けであり、どのような役割を果たし得るのかを、DSMアプリケーションをWWWと接続する支援ツールを開発した経験を通して考察する。

特に、Mの文字列処理能力はHTMLとの親和性が高く、インタプリタ言語である点と合わせて、開発するアプリケーションの柔軟性を高める役割をしたことについて述べる。

さらに、開発ツール自身をWWWブラウザで動作させることで、従来のMのGUI化手段であるMWAPI(M Windowing API), Visual M等のVisual Basicリンクツールに続く第3番目のGUI化手段として、WWWが利用できることを検証した。この方法は、クライアント側の制約が少なく、またMとの親和性も非常に高く有効であることがわかった。

### インターネット技術とM

インターネットで使用されている技術を、おおまかにカテゴリ分けすると、図1のようになる。

インターネットの技術の共通点は、一言でいうとオープンな技術であるといえる。インターネットの普及の一因は、ベンダーやプラットフォームに依存しないオープン性にあるといえる。

その中でも特にWWWは非常な成功をおさめている。クライアント側に何等作り込みを要さず、プラットフォームを選ばないオープン性は、従来のクライアント・サーバアプリケーションにはなかった部分であり、新しいGUI型アプリケーションとして広く普及した一番の理由であろう。

一方、この図の中で、MはDBと言語のほんの一角に位置して、CGIから呼ばれるのを待っているにすぎない。図2は、WWWとMの接続例を示す。

WWWとその周辺のスクリプト言語に要求される場所の「簡便な言語体系」、「強力なテキスト処理能力」、「プログラムのメンテナンス性」は、Mがもっとも得意とするところである。WWWを利用することでMがインターネット技術の中で一つの大きな位置を占め、Mのステータスアップをはかることができるのではないかと、またそうならなければならないのではないかと考えている。

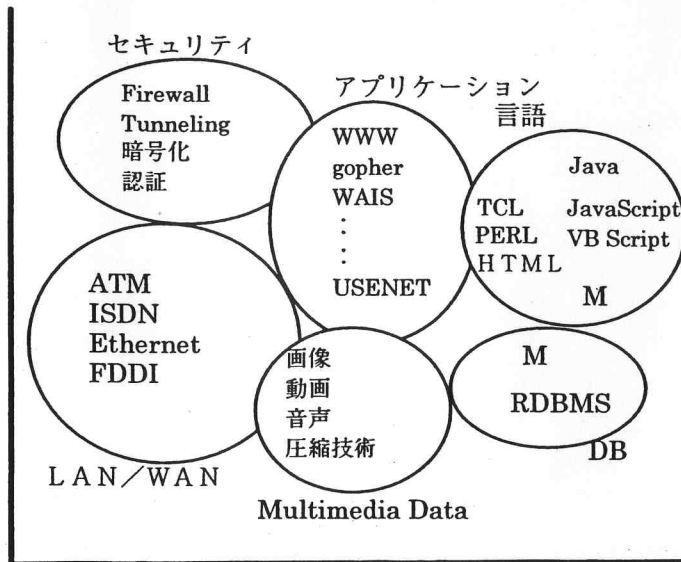


図1 インターネット構成技術 (一部)

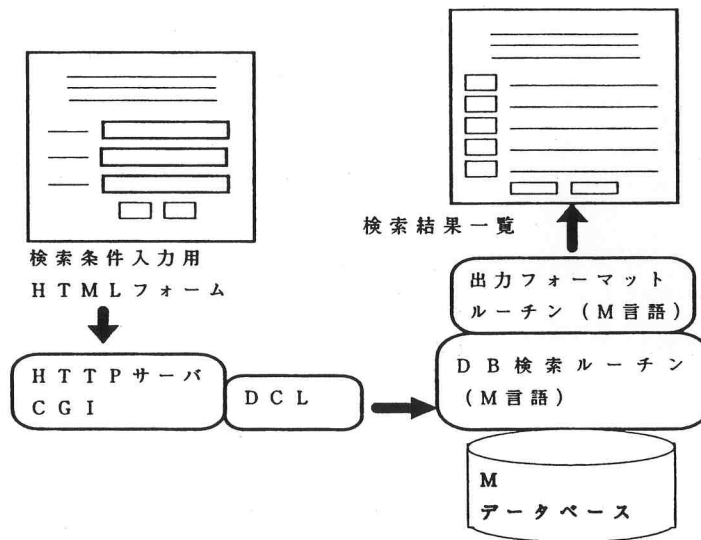


図2 WWWとMの接続例

M言語はインタプリタ言語であり、頻繁に更新されたり構造自身が変わるようなデータに対して、アプリケーション実行時にダイナミックにM自身のコードを生成することによって、これらの変更に対し柔軟に対応できるHTMLコンテンツ生成処理プログラムを書くことができる。これは、他のコンパイラ型言語でリレーショナル型データベースを検索する場合には実現の困難な手法であり、Mのフレキシビリティを生かした方法であると考えられる。

また、Mはテキストハンドリングが充実しているため、HTMLのようなテキストベースのプロト

コルの扱いきわめて容易である。

このように、WWWの技術とMの技術は、融合しやすいものであり、融合することによってさらにMの魅力が増すものであることがわかってきた。しかし、MのWWWサーバアプリケーションを作成する場合、通常、アプリケーション毎にHTML、シェル言語、Mルーチン（検索部分、HTML出力部分）をそれぞれ作成しなければならない。

そこで我々は、Mのインタプリタの late-binding 技術を応用して、MのWWWサーバアプリケーションの開発を効果的に支援するためのツールを開発した。このツールは一種のMでのHTML作成4GLであり、データ/フォーム生成をGUIベースのオペレーションで行うことにより、HTMLとMのコードを作成する。以下にその開発ツールの機能概要を記す。

### WWWベースの開発支援ツール

この開発支援ツール（製品名：WebMan）は、HTMLの入力フォームや、CGIとして動かすプログラムを、データ定義やフォーム定義をもとに生成するものである。また、開発支援画面にもWWWブラウザを使用しており、WWWを使ったM GUIアプリケーションでもある。

開発環境は以下の通りである。

- OS : OpenVMS AXP V1.5
- httpサーバ : CERN http server (V3.0 pre6)
- 言語 : M (DSM V6.2C for OpenVMS AXP)
- DCL (OpenVMS V1.5)

図3は、WebManの動作の概要図である。

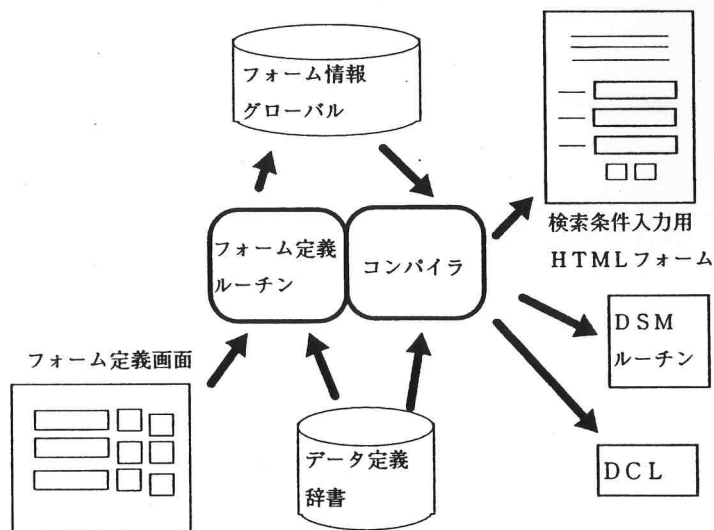


図3 WebMan概要図

WebManの機能概要は、以下のとおりである。

1. HTMLを記述せずにフォームを作成  
フォーム上に置く要素（テキストボックス、ラジオボタン、SELECTリスト等）を、フォームの

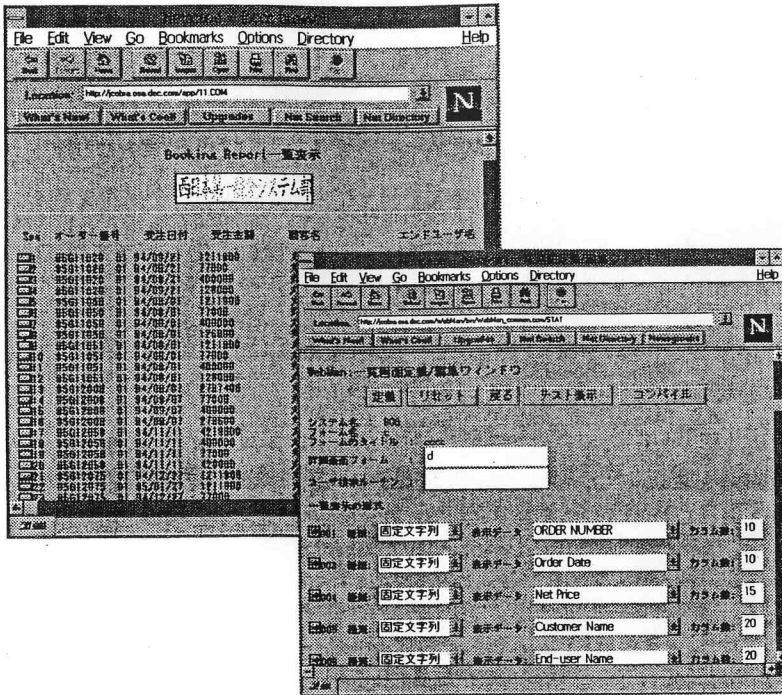


図4 WebMan操作画面

上から順に定義する。図4に定義画面(手前右)と作成された画面の例を示す。要素の種類はHTML 2.0に準じ、リストから選択することが可能で、各要素の長さやデフォルト値、対応するデータネーム、改行の数などは、要素ごとに別画面で細かく指定する。また、作成中のフォームをテスト表示する機能を持つ。

2. フォーム定義、データ定義のコンパイル  
「コンパイル」を実行すると、定義したフォームを表示するDSMルーチンおよび、同フォームを表示するHTMLのファイルを生成する。また、「データベース検索ルーチンを生成する」オプションと共にコンパイルを行うと、フォームからの検索条件をもとにデータベースへアクセスするルーチンを生成する。
3. 簡単な検索ルーチンを生成  
生成したフォームから検索条件のほかに、項目の前方/中間/完全一致、項目間のAND/ORを指定すると、ダイナミックに検索式を作成して検索を行う、汎用的なデータベース検索ルーチンを提供する。
4. ルーチンの呼び出し  
入力フォームから、既存のDSMルーチンを呼び出すことができる。フォームからの入力をローカル変数としてルーチンに渡し、別に定義したフォームへ表示するインタフェースを公開している。
5. DASLデータ定義の利用  
データネームの指定にはDASL(DSM Application Software Library)のデータ定義を使用するので、既存のDASLデータ定義を使用してデータにアクセスすることができる。また、WebManでもデータ定義画面を提供し、DASLがなくても新たなデータ定義を行うことができる。

## おわりに

以上、インターネットの技術とMの親和性、DSMのWWW接続ツールの機能について述べた。WWWサーバを構築してそれを活かしていくためには、コンテンツ(処理プログラムも含む)のメンテナンス(常に新しく)や、性能、アクセス容易性、操作性の維持(常に使いやすく)が特に重要である。Mはそれに優れている。さらに、Javaなどと同様にクライアント側で動作するMスクリプト言語のようなものができれば理想であろう。



## 日本語 DSM V6.4 と Visual M の紹介

佐藤 比呂志

Hiroshi Sato

### 1 はじめに

日本語 DSM for OpenVMS バージョン 6.4 が発表されました。バージョン 6.4 より、InterSystems 社が開発した Visual M 機能が DSM でも利用可能となります。Visual M、およびバージョン 6.4 にて追加された新機能について説明します。

### 2 DSM バージョン 6.4 の主な新機能

- Visual M サーバー機能のサポート
- TCP/IP による DDP のサポート
- OpenVMS V7.0 のサポート
  - キャッシュメモリー 2GB 超 4 GB 未満
  - メモリー・チャンネル・クラスタ
  - スパイアログ・ファイル・システム
  - SCSI Cluster サポート
- DSM イメージのメモリー使用量の削減およびパフォーマンス向上
- ローカル変数の最大長拡張(512->2040 byte)
- Alpha システムにおける倍精度の精度向上 (15 桁から 31 桁)

1 日本デジタルイクイップメント (株)  
西日本第一統合システム部  
〒530 大阪市北区中之島 2 丁目 2 番 2 号  
ニチメンビル  
TEL(06)222-9211 FAX(06)222-9408

Digital Equipment Corporation Japan  
Network and Systems Integration Services  
Nichimen Building, 2-2-2, Nakanoshima, Kita-ku,  
Osaka 530, Japan  
TEL(06)222-9211 FAX(06)222-9408

### 3 Visual M の概要

Visual M は、Windows PC 上で Visual Basic と組み合わせて、DSM にアクセスするアプリケーションを簡単に作成するための開発ツール（ミドルウェア）です。

Visual M を利用し、Windows PC からサーバーの DSM が持つほとんどの機能（言語機能、データベースアクセス機能、ルーチン起動、コマンド実行、関数実行等）にアクセス可能です。

アプリケーションの開発に際しては、Visual M と Visual Basic を組み合わせて使用します。

Visual Basic のカスタムコントロール(VBX)として機能が実現されており、面倒な手続きなしに（プロパティに値を設定するだけで）DSM にアクセス可能です。

Visual M を用いることにより、サーバーとなる DSM が持つ文字列処理の柔軟さ、データベース処理の高速性、開発効率の高さと Visual Basic が提供する強力な画面設計能力の両方の利点を合わせ持つクライアント/サーバー型のアプリケーションを開発できます。

### 4 Visual M アプリケーション開発の流れ

1. Visual Basic にてアプリケーションに必要となる各フォームのレイアウト/インタフェースを開発していく。
2. Visual M の機能を利用するために、1 で作成したフォーム上に Visual M カスタムコントロールを配置する。
3. Visual M カスタムコントロールの各プロパティの初期値をプロパティシート上にて設定する。
4. Visual Basic のイベント処理ロジックを開発する。このイベント処理中に Visual M の機能を利用したければ、Visual M カスタムコントロールを適切に制御する様に Visual Basic のプログラミングを行う。（ルーチン名の設定、入力パラメータの設定等非常に簡単な手続き）
5. 4.のイベント処理ロジック中に Visual M による DSM ルーチンの実行がある場合、そのルーチンのプログラミングを行う。  
この作業は、サーバーにログインして行うか、Visual M 付属の M/DeskTop エディターを利用して行うかのどちらでも可能。
6. 5 で実行したルーチン、関数等が値を返す時、4. のイベント処理にその値を操作するロジックをプログラミング。（Visual M ルーチン/関数呼び出し後処理の追加）
7. 4 から 6 の作業をイベント処理毎に繰り返す。



## 5 Visual M の今後のプラン

### 1. 32 bit OCX サポート

Visual M カスタムコントロールが 32 bit OCX にて実装されることにより、Visual Basic 以外の OCX 対応ツール(Delphi,PowerBuilder 等) からも簡単に利用可能

Windows95,WindowsNT の本来の能力(WIN32)利用可能

### 2. ActiveX Control 対応

MicroSoft VBscript による Visual M 機能の制御可能

Internet,Intranet での利用

### 3. Java 対応

Visual M 機能を Java object として実装

Internet,Intranet での利用



## データ・ウェアハウスとデータマイニング

### Data・Warehouse & Data Mining

今泉幸雄

Yukio Imaizumi

データ・ウェアハウス (Data・Warehouse)は、米国システム・コンサルタントのビル・インモン(Bill Inmon)氏が 1990年に提唱した新しい概念である。データの倉庫あるいは貯蔵庫という意味である。データベースが部品群を持つ工場のようなものであるのに対し、データ・ウェアハウスは製品の倉庫に例えられる。つまり、工場には正規化(バラバラになったデータ)されたデータつまり部品群があり、倉庫にはそれを組み立てられた製品群がある。企業レベルで考えるとデータベースが事業所や部門別に分断されていたため部門間でのデータの整合性や全社的な視点からの分析がしにくかった。その欠点を克服するものとして考えられたのがデータ・ウェアハウスである。全社レベルで詳細データを物理的かつ論理的に一元化することである。データベースとデータ・ウェアハウスを比較すると下記に示す表になる。

企業のコンピュータ・システムは大きく2つに分類される。基幹系システムと情報系システムである。基幹系システムは、日々発生するデータが対象となる商品の受注管理、在庫管理、生産管理、予約管理等でトランザクション中心の定型処理である、利用対象者は一般ユーザである。それに対して、情報系システムは経営者や管理職が、意志決定に使用するためのデータが対象になる。例えば“ここ5年間の特定製品の利益率と販売台数の関係は？”などの問い合わせである。業務は知っていてもコンピュータの専門家ではない。

データ・ウェアハウスの特長としては!目的別単位 : 基幹系の従来の複数アプリケーション、複数のデータベースから必要なデータを必要な形式で取り出す。”統合されたデータ表現 : 基幹系の異なるアプリケーションから統一されたデータ形式に変換する。#時系列的な編成 : 基幹系データは刻々と更新されたデータを重要とするが、データ・ウェアハウスは過去5、10年の長期データを必要とする。\$読み込み専用 : 基幹系で貯えられたデータを目的別に再編成する。そのデータは過去から現在までの企業の歴史であり、書き換えをするものではない。%メタデータ (meta data): これはデータベースにもある、“データに関する情報—データ”の特別なデータである。つまりデータの詳細、要約、データ間の関係の記述である。データ・ウェアハウス管理者とユーザの双方が、同じ認識で業務を遂行する基本である。

データ・ウェアハウスに入れるべき情報には!生情報 (生データ型)”集計・

加工情報（集計データ型）#マスター型情報がある。データ・ウェアハウスを管理するためにはデータ・ウェアハウス管理システム (DWMS)が必要になる。多くは DWMS の代わりに RDBMS を利用しているが、全てをカバーしてはるわけではない。これは RDBMS の正規化されたデータを DWMS において利用する時には、エンドユーザが利用しやすい非正規化に戻す必要がある。またデータの動的な更新も行わず、かつ動的に JOIN 操作を行わないので、スピードが大幅に向上する。これを OLAP (Online Analytical Processing) と言い、RDBMS の生みの親であるコッド博士 (E.F.Codd) により提案されていた、データ・ウェアハウスの一つのタイプである。OLAP の目的は情報の探索および分析であり、今までに蓄積されたデータを過去にさかのぼって調べ、地域別・季節別・製品種別・顧客別などのいくつかの次元から情報を分析する。これは多次元情報を扱う。ビル・インモン氏はデータ・ウェアハウスは方法論であり、完成されすぐに導入可能なシステムがあるわけではない。成功への道は“最初は小さく始めること”と助言している。小さい単位ではじめて、開発のサイクルを短くする。商品としては、日本 IBM の“Information Warehouse”、HP 社の“Open Warehouse”、日本 AT&T 情報システムの“EIF (Enterprise Information Factory)”、ソフトウェア・エージ社社の“ESSBase”、“ADABAS D”、“ADABAS C”等がある。

データベース(以下 DB と省略)とデータウェアハウス(以下 DW と省略)の違いを表にして説明する(表 1)。

表 1: データベースとデータ・ウェアハウスの比較

データベース (DB)	データ・ウェアハウス (DW)
更新あり	更新無し (読み込み専用)
日常的業務	判断業務
早いレスポンス	少々遅くとも良い
今の日常業務に答える (業務担当者)	経営スタッフのニーズに答える (経営者・管理職者)
定型処理 (月次・日次等)	問題発見・学習
基幹系業務 (OLTP)	情報系業務 (OLAP)
現在のデータ	長期的データ (過去 5、10 年)
正規化データモデル	非正規化データモデル (多次元データモデル)
動的	静的
工場 — 部品群 (バラバラデータ)	倉庫 — 製品群 (組み立てデータ)

次にDWの設計方法と経営者からの視点について説明する。

DWの設計方法はDBとのインターフェース設計とDW自身の設計に分かれる。DWのオリジナルのデータは日常使用されている運用DBのデータを抽出(スキャン)されてローディング(移動)されたものである。データのローディングはどのDBのどのデータがDWにあるのか、またいつ実施されたのかをDW管理者が知っていなければならない。抽出プログラム設計をする5つの注意点は、!時刻を記録したデータを抽出して、かつ実施日を保存する”変更部分(デルタファイル)のみの抽出方法#監査ファイルあるいはログファイルをもとに抽出\$抽出&ローディング後にDBに変更があった場合にDWの変更%抽出する量を制限するプログラム化である。次に何に注意してローディングするかであるが、3つのタイプが考えられる。!アーカイブ可能なデータ(例:月末処理の経理データ)”現在の運用環境に含まれるデータ#継続変更のデータ:最新ローディング後に起こったデータである。

DW自身の設計もRDBの設計と同じく大きくは論理モデルの設計と物理モデルの設計からなる。論理モデルの設計は大きくは4つの点を考慮する必要がある。最初の2点はRDBとほぼ同じである。!ERモデルの設計(Entity Relation)”DIS(Data Item Set)の設計:ERモデルで設計された一つ一つのボックスのデータ項目をリスト・アップする。#正規化と非正規化:DWは更新系はまれであるので、異なるテーブルでデータが重複していても問題ない。参照系が多いので非正規化が散在する。\$グラニュラリティ(Granularity):グラニュラリティとはDW内のデータ単位の詳細度を言う。ある一定以上になったならばデータ詳細度を2階層に分けてみる。1階層はDWの(詳細を平均した)データとして利用し、もっと詳細度の高いアーカイブ的なデータは、2階層として他のデバイス(例:磁気テープ等)に保存する。

物理モデルの設計は、従来RDB設計に加えて、時間をベースにデータを物理的に連続させたほうがよい。

経営者向けにはEIS(Executive Information System)がもっとも強力なコンピュータ支援の総称名である。経営者がほしがる経営分析のデータとは!傾向分析とその把握”主要比率指標の計測と追跡#問題の監視\$競合相手分析の4点である。経営者の注意と要求するタイミングは“いつ”であるかの答えは、!タイミングは無作為”今すぐに要求#最終的に求められたデータを統合化した形で要求\$常に気が変わる要求とバラバラである。これに応えられるのがDWである。運用DBがあり、そのデータを抽出とローディングするソフトウェアとデータウェア管理システム(DWMS)があれば、必要な時に設置され、統合化された詳細データと要約のデータが参照出来、かつ広範囲の項目をカバー出来て、傾向把握に必要な長期間のデータが利用出来る。分野的には財務、マ

マーケティング、営業が取り組みやすい。この時に対象となる期間の社会経済データを参照する仕組みにするとよい(図1)。

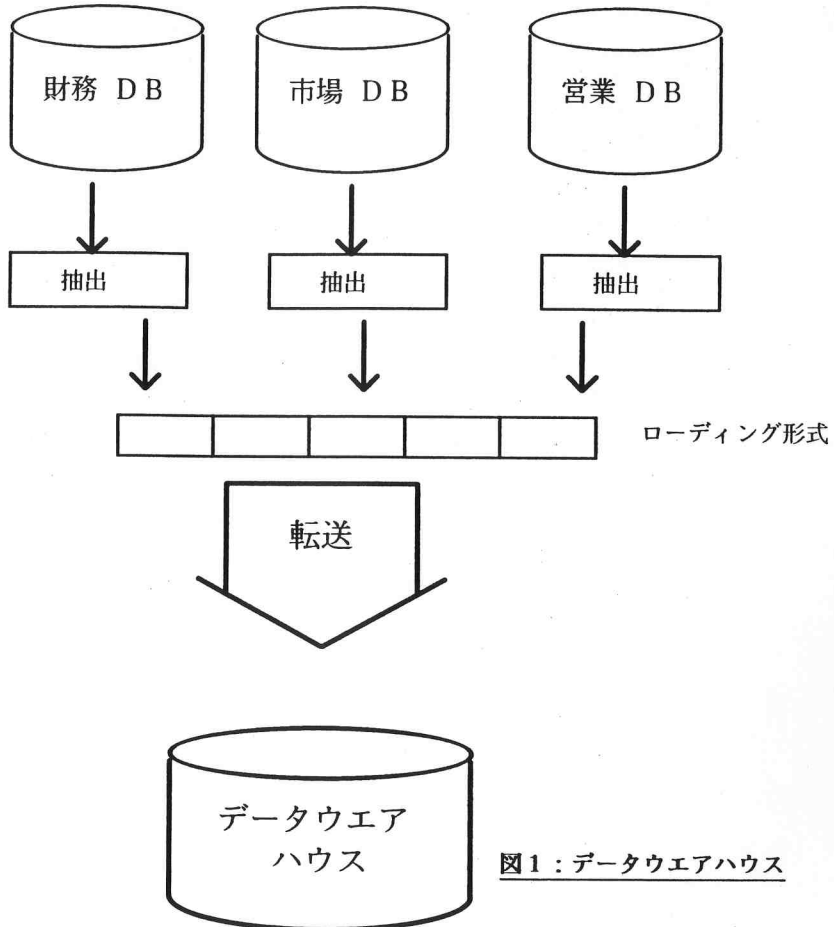


図1：データウェアハウス

サンド薬品では、予算、売り上げと経費を期間・経費単位・支店・勘定等の経営視点からデータウェアハウス(TM/1)を1994年7月に導入した。TM/1はPCベースのClient/Server方式で、最大16次元の角度からデータの分析が出来る。トランザクションデータは2種類あり、1メインフレームから売り上げデータの日次処理と月次処理、予算と経費実績はSAP/R3(UNIX版のClient/Server方式)である。これらメインフレームのデータとUNIXのデータは、抽出処理・変換処理とロード処理をし、Netware Serverを経由し、TM/1 Serverに集めて分析処理をする(図2)。

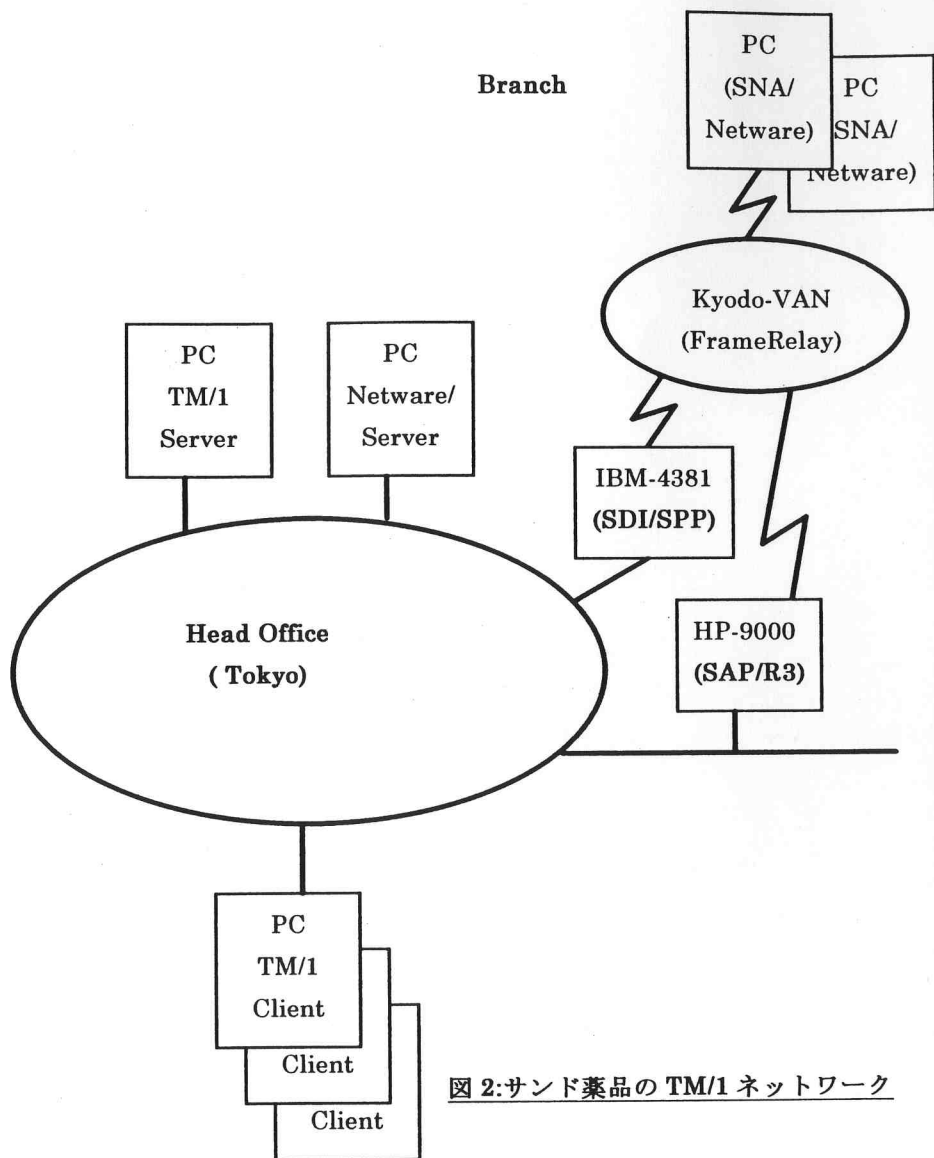


図 2: サンド薬品の TM/1 ネットワーク

データウェアハウスや巨大データベースなどに蓄積した膨大な情報の中から、人が分析や判断をして見分けられない法則や規則の発見と隠れた情報を導き出す技術をデータマイニングと呼ぶ(図 3)。マイニング(Mining は採鉱、鉱業の意味)は、広大な大地の中からダイヤモンドや金、石炭の鉱脈、石油を発

見するギャンプル的な要素があり、超能力と思える不思議なものに違いない。マイニングの場合は、データ間の関連を見つけ出し、深く掘り下げた分析が出来る。レコードを従来より高いレベルで扱うことになる。しかし現在のデータベースは検索・集計の簡易化や効率化を目標に作成されており、残念ながら規則生成を目的には作られていない。検索法則や規則の抽出の視点から見ると、データウェアハウスや巨大データベースからだけではなく人工知能 (Artificial Intelligence) の知識獲得分野や学習分野とも関連がある。自然界に存在する法則は、今までの統計手法である、同時に最大3変数を対象とする解析では解けない問題がたくさんある。例えば理想気体の法則 ( $pV/nT=K$  定数) やケプラーの法則などは、米海軍研究出資 BACON プログラムで再発見された。また学習分野の中のニューラルネットワークのバックプロパゲーション・アルゴリズムや決定木手法の Quinlan 教授が考えだした ID3 アルゴリズム等が法則や規則抽出手法となる。米国では、医療、金融、流通、通信業などで関心が高まっており、日本でも市場拡大が予想される。

例えば巨大顧客データベースから商品のダイレクトメールを顧客に出す場合、商品ごとにどのような顧客たちのグループに焦点を合わせれば、低価格の労力で最大の利益を得られるかを調べる。特定の商品ならば年齢等でデータを抽出すればよいが、一般の商品ではデータベースの解析がともなわないために、すべての顧客、または非常に漠然とダイレクトメールを出すのが日常茶飯時である。これに対して、データマイニングを使用すれば、要求にあったデータのパターンを見つけ出せる。適切なツールを選び出し活用すれば、利益は最大になる。例えば、信用貸しのリスクの判断、クレジットカード不正行為の察知、製品の保証書の管理、小売り店の在庫追跡、商品の陳列方法、新製品の取り扱い、新しい通信サービス、病院の患者の病歴の症状と治療、保険会社の過去の事故歴など、ビジネス戦略の基礎を築くことが出来る。

データマイニングによって導出する情報のタイプは4種類である。!アソシエーション：ある出来事に複数の出来事が関連ある。例えばコーヒーを購入するお客は砂糖とミルクを購入する。”シーケンス：一つの出来事から時系列に複数のことが発生する。例えば大学や高校に合格した学生の45%は2週間以内に新しいパソコンを購入し、一ヶ月以内インターネットにアクセスする。#分類：データがどのグループに属するかを認識する。例えば隣近所に新しいスーパーが開店して今までの固定客の一部が取られてしまった。その時に流れたお客の特徴を抽出する。\$クラスタリング：分類と関連するが、まだ分類出来ないデータ集合に適合できる。例えば年齢、職業、会社でのポジション、通勤時間等に関連した配置転換の選択等に適合できる。

実際にデータマイニングを実行するには、上記の4種類に分類されたデータ



を組み合わせ使用。多くの手法は、ニューラルネットワークのバックプロパゲーション・アルゴリズムや決定木手法のアルゴリズムとその両方を併用する方法が考えられる。SQL(Structured Query Language: 構造化問合せ言語)は必要ない。!バックプロパゲーション: 入力用データとその入力があったときに出力された期待値が組みになったアルゴリズムである(図4)。従って入力層、中間層、出力層の3層から構成される。各々 $m$ 個からなる入力層と $k$ 個からなる中間層の重みつきの $m:k$ 関係、 $k$ 個からなる中間層と $n$ 個からなる出力層の重みつき $k:n$ 関係と $k$ 個の中間層の値の3つが知識の要素となる。入力値と出力値はわかっているが、具体的に知識や法則として表現しにくいモデルに適合しやすい。不動産屋が賃貸マンションの貸し出し判断基準の入力条件として、年齢、性別、職業、年収、財産等で、出力条件は貸し出しする・しないの値になる。”決定木手法: データが持つ属性を使ってグループ分けする手法である。“もし…ならば…である”というルールが木構造を成す形である。木の根の最初の部分が条件によって2つ以上に分けられて、さらに次の判定条件で2つ以上に分けられる方法である。分ける範囲は条件の数に依存する。最終的には最終条件に対する判断値になる。例えば条件として検査値(年齢・眼底所見・血圧・コレステロール,,,)、最終条件値として疾患名(脳出血・狭心症・心筋梗塞,,)が判断値になる。

データマイニングのツールの選択方法としては下記の点があげられる。!データをモデル化するためにどんな解析手法を採用しているか。”どのような問題の解決対象を得意としているか。#どのようなハードウェアとOS上で動作するか。\$データの入力にどんなインターフェースを用意してるか。%分類データと時系列データをどのように解析できるか。&データの質(ノイズ等)にどれくらい対応できるか。'どのくらいのデータのサイズ(変数と行数)まで可能か。

成功の秘訣は、使用するツールの限界とデータウェアハウスや巨大データベースに入っているデータの質(欠損値、不明、ノイズ値)を理解すること。そのためには、3つの要素が必要である。幅(データが網羅しているテーマ数)、深さ(各テーマに含まれるデータの価値)、そして量(全般的なデータ量)である。ただ大量にデータをもっていけばうまくいくとは限らない。

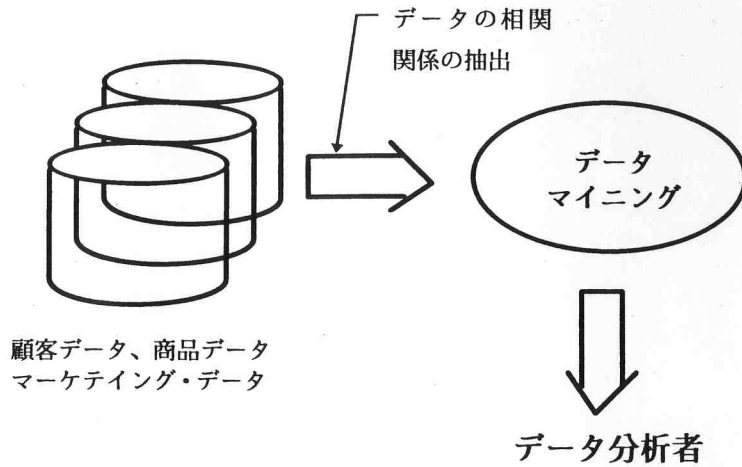


図3: データマイニングの概念

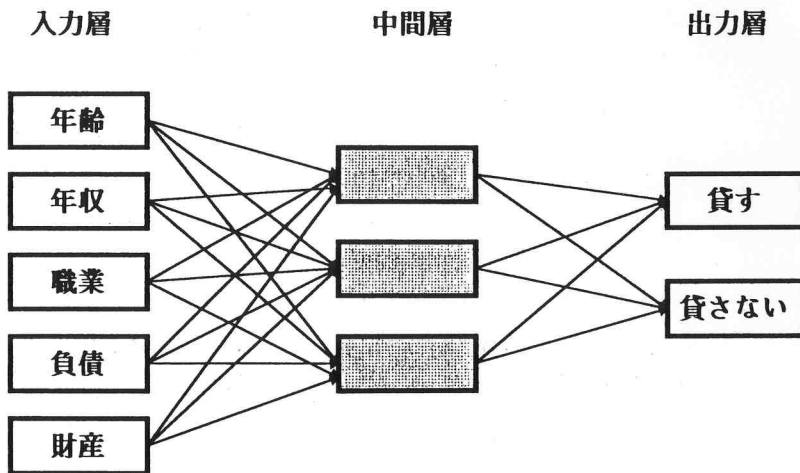


図4: ニューラルネットワーク

## 「日本Mテクノロジー学会」ご入会のご案内

日本Mテクノロジー学会（日本MTA）は、M言語（MUMPS）の利用・改良・普及を目的とした団体で、個人や法人が加入して活発な活動を行っております。M言語はANSIにFORTRAN及びCOBOLに続いて3番目の標準コンピュータ言語として制定され、米国連邦情報処理標準言語にも採用されました。さらに1992年5月にはISO標準言語として制定されるに至っております。一方、近年のコンピュータのダウンサイジングの流れにあって、ユーザーも着実に増えつつあります。

日本MTAは先に述べたような目的に向けて種々の活動を続けておりますが、貴方にも、是非とも日本MTAに参加し活動を盛り上げて頂きたいとご案内申し上げます次第です。

### A. 日本MTAの活動

- 1) 年次学術大会、研究会や講習会の開催
- 2) M言語に関する技術情報の提供
  - PC通信Nifty-SERVE上にMUMPSフォーラムを設置
  - MTAニュースの発行
  - 各種資料の配布
- 3) 学術雑誌「Mumps」の出版
- 4) M言語改良仕様の検討・・・米国M Development Committeeと連携
- 5) 国際MTA、各国MTA（MUG）との交流
- 6) M言語のJIS化推進
- 7) ソフトウェアの公開流通

### B. 会員の特典

会員になることにより次のような特典が考えられ、充分満足頂けるものと考えられます。

#### \* 個人会員の特典

- 1) 日本MTA年次大会、M言語関係学術集会、研究会、講習会のお知らせ
- 2) 日本MTA主催の学術集会、研究会、講習会などの参加費用の割引
- 3) M言語に関する各種資料の実費提供
- 4) 流通、ソフトウェア（MTAPAL）の低額頒布
- 5) 「MTAニュース」の無料配布
- 6) M言語ベンダーの折々のプロダクツ紹介・パンフレット・カタログ類の頒布
- 7) 雑誌「Mumps」の無料配布

- ・上記の各種活動を通じて、M言語に関する全世界の最新の技術情報が得られます。

\*法人会員の特典

法人会員は「日本MTAの目的に賛同する法人で、日本MTAの目的を遂行するために積極的に事業を後援する事を表明した者とし、正副各1名の代表者を登録し、正副代表者とも個人会員と同等の資格を持つ」こととなります。尚、正副代表者には正会員と同様の日本MTAの役員としての道があります。

- 1) 日本MTA主催の集会には5名迄、会場費、講習会費などを会員割引
- 2) 日本MTA主催の医療人、企業人を対象とする講習会へ法人会員から優先的に出講
- 3) 日本MTA主催の集会への出品、展示に関する料金の割引
- 4) 日本MTA学術大会論文集、MTAニュース等への広告費の割引
- 5) 法人会員のプロダクトのパンフレット、カタログ類の会員への頒布
- 6) ユーザー法人にはM言語ベンダーないしシステムエンジニアの紹介
- 7) 日本MTAの流通パッケージ(MTAPAL)を割引料金で利用
- 8) MTAニュースを単なる広告ではなく、新しいプロダクツの紹介等の質の高いPRのために利用可能

注意) 法人会員は、国際MTAが設けている施設会員と企業会員に相当するものですが、学校法人・国立施設など税法上非営利団体扱いの法人を非営利法人とし、国際慣例よりも40%低い基本会費を申し受けます。その他は企業法人ないしベンダー法人としての会費を申し受けます。ご入会の手続きは「法人会員入会申込書」によってお願い申し上げます。

- ・上記の各種活動を通じて、M言語に関する全世界の最新の技術情報が得られます。
- ・M言語ユーザ間、M言語を取り扱うベンダー・メーカー間とのコミュニケーションが充実します。

C. 会費

ア) 個人会員

入会費 ￥4,000.

年会費 ￥6,000.

イ) 法人会員

入会費 ￥10,000. (営利・非営利法人共通)

年会費 ￥50,000. (1口) ←営利法人

￥30,000. (1口) ←非営利法人

注意) 会計年度は、毎年4月1日から翌年3月31日までです。

D. ご入会手続き

- 1) 入会資料請求<電話・FAX・郵便>
- 2) 事務局から送付された「会員登録票」(法人会員の場合は正・副代表者の「会員登録票」及び「法人会員申込書」)に必要な事項を記入の上、事務局までお送り下さい。
- 3) 郵便払込、銀行振込で入会金、年会費を事務局に納金して下さい。
- 4) 事務局から会員登録完了通知と会員証、資料他をお送りします。  
(入会日は入会費・年会費納入日です。)

E. 入会費・年会費お支払方法

日本MTA事務局より会費の請求がございましたら、以下の何れか方法でお支払下さい。但し、お手数料は振込人払いとさせていただきますことをご了承下さい。

ア. 郵便振替      口座番号：01440-8-4520  
                    加入者名：日本Mテクノロジー学会

イ. 郵便貯金      口座番号：15330-12879211  
                    加入者名：日本Mテクノロジー学会  
                                事務局代表 山本 和子

ウ. 銀行口座      口座：山陰合同銀行 出雲支店  
                    口座番号：普通口座 027-2904176  
                    名義人：日本Mテクノロジー学会  
                                山本 和子 (やまもと かずこ)

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

卷之三

## 「日本Mテクノロジー学会」規約

### 第一章 総 則

第1条 本会は日本Mテクノロジー学会 (M Technology Association of Japan)という。

第2条 本会の事務所は幹事会の承認を経て、学会長が指定するところに置く。

### 第二章 目的および事業

第3条 本会は「M言語」並びにこれに関する情報システムの利用、応用、改良、並びに普及を行うことを目的とする。

第4条 本会は前条の目的を達成するため次の事業を行う。

- 1) 学会大会、フェア、研究会、講習会などの開催
- 2) 学会誌、ニュースなどの刊行物の発行
- 3) M言語の日本語装備の標準化
- 4) M言語の標準装備の監視
- 5) 海外のMTA (MUG) などとの連携活動
- 6) 内外の関連諸学会との連絡ならびに協力活動
- 7) M言語利用技術の相互交換の促進、本会に提供された資源の整備、管理ならびに会員への還元
- 8) 日本Mテクノロジー学会出版会に関する事業
- 9) その他目的達成のために必要な事業

### 第三章 会 員

第5条 本会会員は個人会員と法人会員からなる。

- 1) 個人会員は本会の目的に賛同し、本会の対象とする領域、又はそれと関連する領域において活動する個人とする。
- 2) 法人会員は本会の目的に賛同する法人で、本会の目的を遂行する為に積極的に事業を後援する事を表明したものである。法人会員においては正副各1名の代表者を登録するものとする。正副代表者は個人会員と同等の資格を有する。

第6条 本会に入会を希望する者は所定の申込書に入会金及び会費を添えて本会事務所に申し込まねばなら

ない。

第7条 本会会員は、毎年所定の会費を前納しなければならない。

第8条 本会会員で住所変更のあったものは速やかに住所変更届を、また退会しようとするものは退会届を本会事務所に提出しなければならない。本会会員で、住所不明となるか催促にも拘らず2か年を越えて会費納入遅滞のあったものは退会の扱いを受ける。物故会員は退会の扱いを受ける。

第9条 本会の規約に背く行為のあった会員は、幹事会の議決を経てこれを除名することができる。

#### 第四章 役員その他

第10条 本会に次の役員を置く

1) 学会長	1名
2) 日本Mテクノロジー学会大会長(以下「大会長」という)	1名
3) 日本Mテクノロジーフェア実行委員長(以下「フェア実行委員長」という)	1名
4) 幹事 庶務財務担当	1名
国際担当	1名
流通担当	1名
広報担当	1名
雑誌担当	1名
ネットワーク担当	1名
M言語標準化担当	1名
JIS・ISO担当	1名
5) 会計監事	1名
6) 評議員	若干名
7) 日本Mテクノロジー学会出版会理事長	1名
8) 日本Mテクノロジー学会出版会理事	若干名

第11条 各役員の選出または構成を次のように定める。

- 1) 評議員に欠員が生じた場合、学会長は評議員会の推薦者を総会に諮り、その承認を得て決定する。評議員の定数は学会長が定める。但し、各評議員の構成割合は会員の職域構成割合に近いものとする。
- 2) 学会長及び会計監事は、評議員会の推薦者を総会に諮り、その承認を経て決定する。
- 3) 幹事は学会長が推薦し、総会の承認を経て決定する。学会長と幹事は併任できない。
- 4) 大会長は学会長が幹事会の推薦者を総会に諮り、その承認を経て決定する。
- 5) フェア実行委員長は学会長が幹事会の推薦者を総会に諮り、その承認を経て決定する。
- 6) 出版会理事長並びに理事は学会長が推薦し、総会の承認を経て決定する。



第12条 各役員の任務は次のように定める。

- 1) 学会長は会を代表し、総会、幹事会、評議員会の議長となる。
- 2) 大会長は、年次日本Mテクノロジー学会大会を総括する。
- 3) フェア実行委員長は、年次日本Mテクノロジーフェアを総括する。
- 4) 庶務財務担当幹事は、本会に関する庶務及び全ての資金及び財産の管理を行う。また、最新の名簿の管理、総会その他の議事録の管理を行う。
- 5) 国際担当幹事は、海外のMTA (MUG) 組織との連携並びにM言語開発委員との協力を司り、その他の国際的協力を行う。
- 6) 流通担当幹事は、M言語応用プログラムのユーザー間相互交換の促進、MUGプロトタイプ・アプリケーション・ライブラリー (MUGPAL) などM言語資源の整備、管理、維持、会員に対する資料提供等のサービスを行う。
- 7) 広報担当幹事は、Mテクノロジーニュース等を通じ広報活動を行う。
- 8) 雑誌担当幹事は、学会誌「Mumps」の編集を兼ね、出版の進行を司る。
- 9) ネットワーク担当幹事は、ネットワークを活用した会員間のコミュニケーションの向上を図る。
- 10) M言語標準化担当幹事は、M言語の標準化を図る。
- 11) ISO・JIS担当幹事は、M言語のISOとJIS標準制定に関することを司る。
- 12) 会計監事は、年次会計の監査を行い総会に報告する。

第13条 各役員の任期を次のように定める。

- 1) 学会長、幹事、会計監事の任期は、4月1日より翌々年3月31日までの2年間とし再任を妨げない。
- 2) 大会長の任期は、前学会終了時に始まり学会の残務処理の終了までの期間とする。
- 2) フェア実行委員長の任期は、前Mテクノロジーフェア終了時に始まりMテクノロジーフェアの残務処理の終了までの期間とする。
- 3) 評議員の任期は特に定めないが、4年間続けて評議員会に出席しなければ評議員資格を失う。

## 第五章 会議および委員会

第14条 (総会)

- 1) 総会は本会の最高の議決機関である。
- 2) 総会は学会長が毎年1回召集する。但し、幹事会の議決による場合または会員の5分の1以上から請求された場合、学会長は臨時総会を召集しなければならない。
- 3) 総会の議長は学会長とする。
- 4) 次の事項は総会に提出してその承認を受けなければならない。
  - a. 事業報告および収支決算
  - b. 事業計画および収支予算
  - c. その他幹事会が必要と認めた事項

## 54 「日本Mテクノロジー学会」規約

- 5) 総会の成立に必要な出席者数は会員のうち50名または10%の少ない方を上回る数とする。
- 6) 総会の議決は本規約に別に定めるものの他、出席会員の過半数による。

### 第15条 (幹事会)

- 1) 学会長が必要に応じて召集する。但し、幹事の過半数から請求があった時は、学会長は幹事会を召集しなければならない。
- 2) 幹事会の議長は学会長とする。
- 3) 幹事会は学会長、大会長、フェア実行委員長、幹事、会計監事により構成される。
- 4) 学会長は必要に応じて各種委員会の委員長を出席させることができる。
- 5) 幹事会の議決は構成員の過半数による。

### 第16条 (評議員会)

- 1) 学会長が毎年1回召集する。但し、学会長は必要に応じて臨時評議委員会を召集する。
- 2) 評議員会は学会長の諮問に答え本会の重要案件を審議する。議長は学会長とする。
- 3) 評議員会は学会長、会計監事、Mumps 編集委員、新評議員を総会に推薦する。

### 第17条 (学会誌 Mumps 編集委員会)

- 1) 雑誌担当幹事は必要に応じて学会誌 Mumps 編集委員会を召集する。
- 2) 学会誌 Mumps 編集委員会の議長は雑誌担当幹事とする。
- 3) 学会誌 Mumps 編集委員は編集委員会が任命する。任期は3年とし、再任を妨げない。

### 第18条 (各種委員会)

- 1) 学会長は必要に応じて幹事会の議を経て各種委員会を設置、統合、分化、改廃することができる。

### 第19条 (日本Mテクノロジー学会大会)

- 1) 本会は年1回以上の日本Mテクノロジー学会大会を開催する。

### 第20条 (日本Mテクノロジーフェア)

- 1) 本会は年1回以上の日本Mテクノロジーフェアを開催する。

### 第21条 (日本Mテクノロジー学会出版会)

- 1) 日本Mテクノロジー学会出版会の規約は別途定める。

第22条 本会の資産は次の通りとする。

- 1) 本会の設立当初からの財産
- 2) 入会金および会費
- 3) 事業に伴う収入
- 4) 資産から生ずる利子など
- 5) 寄付金品
- 6) 負担金
- 7) その他

第23条 本会の資産は、学会長及び庶務財務担当幹事が管理する。

第24条 本会の重要な財産（基本財産）に関しては、これを消費し、または担保にしてはならない。但し、本会の事業遂行上止むを得ない理由があるときは、幹事会の出席者の2/3以上の議決と総会の出席者の3/4以上の議決を経てその一部に限り処分し、または担保に供することができる。

第25条 本会の事業計画およびこれに伴う収支予算は、年度毎に学会長および庶務財務担当幹事が編集し、幹事会の議決を経て総会の承認を得なければならない。

第26条 本会の事業報告書および収支決算は、年度毎に学会長および庶務財務担当幹事が作成し、会計監事が監査し、幹事会の議決を経て総会の承認を得なければならない。

第27条 本会支援のため各種団体よりの負担金、寄付、研究費などの交付があった場合、幹事会の承認により本会の資産として受け入れる。

## 第七章 規約の変更ならびに解散

第28条 本規約の改正は幹事会および総会において各々出席会員の2/3以上の議決を経なければならない。

第29条 会を解散するには総会において出席会員の3/4以上の同意を必要とする。

第30条 会の解散に伴う残余財産は、法律による制限のあるもの他は世界保健機構（WHO）に寄付するものとする。

## 第八章 付 則

第31条 本会の略称を日本MTA、英文略称をMTA-JPという。

第32条 本会の入会費、年会費は別に定めるものとする。

第33条 学会長は本会の発展に功績のあった特定個人に対し名誉会長、名誉会員の称号を与えることができる。

第34条

- 1) 本規約は1977年10月29日より発効するものとする。
- 2) 本規約は1979年 9月14日より改訂し発効するものとする。
- 3) 本規約は1987年 7月29日より改訂し発効するものとする。
- 4) 本規約は1991年10月31日より改訂し発効するものとする。
- 5) 本規約は1992年 8月 1日より改訂し発効するものとする。
- 6) 本規約は1992年10月29日より改訂し発効するものとする。
- 7) 本規約は1993年 4月 1日より改訂し発効するものとする。
- 8) 本規約は1994年 8月 6日より改訂し発効するものとする。
- 9) 本規約は1995年 9月30日より改訂し発効するものとする。
- 10) 本規約は1996年 9月15日より改訂し発効するものとする。

## 「Mumps」投稿規定

(1991年 7月10日制定)

(1994年12月 1日改正)

本規定は日本MUMPS学会誌「Mumps」に、会員が自発的に寄稿する論文（以下投稿論文という）に関する必要事項を定めたものです。学会誌「Mumps」には、編集委員会が依頼する原稿（依頼原稿）も掲載しますが、それについての必要事項はそのつど定めます。

### 1. 論文の主題

投稿を受け付ける論文の主題は、コンピュータシステム／言語であるMUMPSに直接、間接に関係するものとします。

例えば、MUMPSの利用技術についての考案や開発、MUMPS言語についての言語仕様や提言、MUMPSシステム装備、MUMPSと他の世界とのインターフェース、MUMPSの教育など、MUMPSに関係するあるいは関係しそうなテーマについて広く受け入れます。ただし、他の雑誌に掲載された、あるいは投稿中の論文はお断わりします。

### 2. 投稿論文の種類

投稿論文は次の6種類に限ります。

#### 1) 原著論文

未投稿で、論文の主要部分に独創性、独自性のある論文。既に発表した問題について別の視点からまとめた論文も未投稿原著論文であり得ます。また、応用開発、調査等であっても、その過程での創意工夫や独自性があれば原著論文の対象とします。

#### 2) 総説

ある主題について、過去の研究業績を詳細にまとめ文献を伴って記述し、その主題に関する現状と将来展望を明らかにした論文。

#### 3) 研究速報

新しい研究成果が原著になるほどにはまとまっていないが発表に価値があると考えられるもの。

#### 4) 技術ノート

作成したプログラムや新しいシステムの紹介など、MUMPS技術に関する論文で、会員の相互の利益になると思われるもの。

#### 5) フォーラム

意見、提案、提言、感想、著書や学術集会の紹介など、上記以外で会員の利益になると思われるもの。

#### 6) Letter to the editor

原著論文に対する質問やコメント、日本MUGの活動に関係のあるコメントなど。

### 3. 投稿論文の長さ

原則として下記の表の通りの長さとしします。原稿用紙（横48字×縦41行=1968文字）で刷り上がりページ1枚となります。ただし、これを越える場合でも、編集委員会が必要と認めた場合には別に定める超過料金を支払って掲載することができます。

## 58 「Mumps」投稿規定

論文の種類	論文のページ数（刷り上がり）
原著	10ページ（以内）
総説	30ページ
研究速報	6ページ
技術ノート	6ページ
フォーラム	4ページ
Letter to the Editor	1ページ

### 4. 投稿者の条件

- 1) 筆頭著者は日本MTA会員であること。
- 2) 共著者も原則として会員であることとします。

### 5. 原稿の送付

オリジナル原稿とそのコピー2部を下記編集委員会宛てに送ってください。原稿到着日を投稿の受け付け日としその日付を誌上に明記致します。

### 原稿送付先・連絡先

〒259-11

神奈川県伊勢原市望星台

東海大学医学部医学情報学内

雑誌Mumps編集委員会宛

Tel:0463-93-1121 FAX:0463-96-4301 NIFTY-serveID:E-Mail PFB00710@niftyserve.or.jp（岡田）

### 6. 掲載の採否

投稿された原稿は、編集委員会が依頼する2名の査読者が査読します。そしてその査読者の意見を考慮して編集委員会がその原稿の採否を決定します。査読の結果によっては、原稿の内容や論文の種類を修正変更することを投稿者をお願いすることもあります。

### 7. 原稿作成要領

#### 1) 原稿の構成

投稿原稿はおよそ次の構成に従って作成してください。

- a) 論文の題名
- b) 著者名、所属、所在地
  - a) と b) は日本語と英語の両方を記入して下さい。
- c) キーワード・・・8語以内（日・英）
- d) 和文要旨・・・200字から400字
- e) 英文要旨・・・200wordsから300words
- f) 本文
- g) 謝辞・・・・・・・・必要に応じて
- h) 文献リスト

文献の引用は本文中の引用箇所に出現順に通し番号 [1], [3-5] 等を記し、本文の末尾に一括

して引用番号順に並べて下さい。雑誌の文献は引用番号、著者名、論文題名、雑誌名、巻号、最初と最後の頁数、西暦年号の順です。

単行本の文献は引用番号、著者名、題名、書名、版数、引用頁、発行社、発行地、西暦年号の順です。  
(例)

1. 福井太郎：糖尿病患者管理システムの開発，医療情報学，10(2):30-35(1990).

i) 図表 … 図や表は別に一括して縮尺可能なカメラレディの図表原稿を添付し本文のどこでそれらに言及しているかを原稿のワク外に明示してください。

j) 特殊文字…特殊文字は原則として禁止しますが使用される場合は使用位置を通常の校正の記号等を用いて朱書してください。

## 2) 投稿原稿 (原稿用紙で提出)

原稿はワープロで、横48文字×縦41行を1頁として作成して下さい。手書きでも受け付けます。

なお、原稿には表紙をつけ、表紙にはつぎの事項を記入してください。

表紙…題名

連絡先 (氏名・住所・電話・FAX)

原稿の種類

原稿の枚数 (本文・図・表別に)

別冊希望部数 (50部の倍数)

その他…特殊文字等を使用されている場合は明記して下さい。

## 3) 印刷原稿 (フロッピーまたは電子メールで提出)

採用が決定した印刷原稿は、ワープロ (一太郎、MS-Word、またはMS-DOSのテキストファイル) のフロッピーまたは電子メールで提出して下さい。

(フロッピー作成時の注意事項)

\*横48文字×縦41行が1頁になって印刷されます。ただしワープロではスペースを入れないで字数に関係なく連続打ちをして下さい。改行印もスペースでなくリターンキーで入れて下さい。

\*別に横48文字で紙に印刷または手書きした原稿を添えて下さい。これを見本にして活字を組みます。横48文字目に「。」がこないように文章を工夫して下さい。

\*原稿は題名 (日本語)、題名 (英語)、著者名 (日本語)、著者名 (英語)、著者所属・住所 (日本語)、著者所属・住所 (英語)、和文抄録、キーワード (日本語)、英文抄録、キーワード (英語)、本文の順で同一ファイル名に保存して下さい。

\*ワープロ中には図表や図表の挿入位置の表示を入れないで下さい。図表の挿入位置は紙に印刷した原稿のワク外に朱書して下さい。イタリック等の特殊文字も紙の原稿の中に朱書して下さい。

\*図表は必ず1図を1枚の紙に印刷して下さい。そのままカメラレディで印刷します。

\*英数字は半角にして下さい。倍角その他特殊文字や罫線を使用しないで下さい。

## 8. 別刷

著者は別刷を最低50部買取ることとします。別刷の料金は別に定めます。別刷の部数は校正稿提出時に申し出ていただければ、50部単位で増刷いたします。

「Mumps」誌編集委員

1998年12月現在

- 編集委員長 岡田 好一 (東海大学医学部医用工学情報系)  
編集委員 河村 徹郎 (鈴鹿医療科学大学医用情報工学科)  
木村 一元 (獨協医科大学医学情報センター)  
山下 芳範 (福井医科大学医学情報センター)  
山本 和子 (島根医科大学医療情報部)

### 編集後記

雑誌 MUMPS 1998年版を出版することができ、喜ばしく思います。私、雑誌担当幹事の都合により、山本和子島根医科大学教授をはじめとするMTA会員みなさまの多大なご協力をいただきました。ここに厚く御礼申し上げます。

ここ2-3年のM言語の動きは、その直前の沈滞したムードと違い、GUIやネットワークの普及などでたいへん華やかなものとなっています。今が、M言語の区切りのひとつの時代と記憶されるのだと思います。この歴史的な転換点をダウンサイジングの最先端技術を包み込むM言語とともに共有することができたことに、幸運な巡り合わせを感じます。

(岡田)



Mumps(The Official Journal of M-Technology Association-Japan)

---

第 21 卷

1998 年 12 月 1 日発行

発行者 日本Mテクノロジー学会  
会長 大櫛陽一  
〒259-1193 神奈川県伊勢原市望星台  
東海大学医学部医用工学情報系  
Tel: 0463-93-1121(代表)  
FAX: 0463-96-4301

編集者 日本Mテクノロジー学会 編集委員会  
委員長 岡田好一  
〒259-1193 神奈川県伊勢原市望星台  
東海大学医学部医用工学情報系  
Tel: 0463-93-1121(代表)  
FAX: 0463-96-4301

事務局 日本Mテクノロジー学会事務局  
庶務財務担当幹事 山本和子  
〒693-8501 島根県出雲市塩冶町 89-1  
島根医科大学医学部医療情報学講座  
Tel: 0853-23-2111(内線 5380)  
FAX: 0853-20-2170

印刷 株式会社学術出版印刷  
〒554-0022 大阪市此花区春日出中 2-14-9  
Tel: 06-6466-1588  
Fax: 06-6463-2522

---





CONTENTS

■ Editorial

Future of M : What way do you choice?..... Norio Sasagawa 1

■ Articles

The Superiority of M-Technology for the Hospital Information System ..... 3

Hee Sun Jeon

Yun Sik Kwak

Hune Cho

Hyung Soo Kim

Soo Yong Jang

A case of information network system using DTM with the internet ..... 9

Seiji Hirose

DSM/CGI programming..... Toshio Imai 15

DSM/DASL Test Terminal Emulator..... Toshio Imai 23

Internet technology and DSM/WWW connection tool..... Reiko Jozaki 29

Shizuteru Sugiyama

Japanese DSM V6.4 and Visual M..... Hiroshi Sato 35

Data • Warehouse & Data Mining ..... Yukio Imaizumi 39

■ Other Information

Information for MUG ..... 47

Instruction for Authors ..... 57

Editor's Postscripts ..... 60