

病歴検索システム開発のためのプログラミング言語の評価

An Evaluation of IR-oriented Programming
Languages for the Implementation of Clinical
Record Retrieval Systems

齊 藤 孝

Takashi Saito

Résumé

This paper describes an evaluation of minicomputer-based programming languages from the standpoint of information retrieval technology. Three clinical record retrieval system, namely, a kidney dialysis patient medical system using TOD (time-oriented database) method, a cerebral stroke information retrieval system, and a diagnosis-oriented summary system using POS (problem oriented medical system), are shown as the evaluation sample.

At the design stage of these application systems, data elements, data attributes, and data structures are defined and several common functions and algorithms such as string manipulation, tree-structured database filing, logical operation and online feature are extracted.

Secondly, at the implementation stage, the selection and evaluation of the programming languages required for each clinical retrieval application are discussed. The check list chosen from the functions and algorithms is proposed. And such high level languages as COBOL, FORTRAN, APL, MUMPS, and BASIC are compared using the check list. MUMPS is selected as the best suitable language for clinical record retrieval programming.

This type of consideration plays a major role in the design and development of information retrieval systems.

はじめに

I. 病歴検索システム

- A. 事例1—人工透析型—
- B. 事例2—脳卒中発症型—
- C. 事例3—疾患名別サマリー型—

II. システムの設計

- A. 要求機能とアルゴリズム
- B. アプリケーションの構成

齊藤孝：東京芝浦電気株式会社電算機システム技術部

Takashi Saito, Computer System Engineering Department, Toshiba Corporation.

III. システムの開発

- A. インプリメンテーションの条件
- B. 高級言語の選択
- C. 評価と考察

IV. MUMPS の採用

- A. MUMPS によるプログラミング
- B. MUMPS の誕生と背景
- C. MUMPS の仕組
- D. 言語とプログラミング

おわりに

はじめに

医療情報の種類とその処理の方法は様々であるが、発生の源となるものは医師と患者の間で取り交わす診療録である。診療録はカルテ、疾患記録、病歴など数々の呼び名を持っているが、その目的と価値は次の6点とされる。¹⁾ すなわち、(1)患者の診療記録、(2)病院の医療評価と医療水準の向上、(3)医師の卒後教育、(4)病院と医師の法的証拠、(5)地域社会の公衆衛生、(6)疫学的研究などである。

本論は病歴そのものを本筋とするものではなく、著者の請け負った3事例の特殊なタイプの病歴検索システムの設計と開発の経験を中心に論じる。いちがいに病歴といっても対象とする診療科や疾患などの主題によって多種多様である。これをコンピュータアプリケーションとしてインプリメントするには、ソフトウェア工学的な観点による分析と評価が有効である。それには、まずデータの属性とデータの構造を洗いだし、データベースフェイルを構築し、処理の形態と機能のアルゴリズムを抽出して、さらに共通要素から標準化を試みるといったシステムの設計の段階を必要としよう。

次に実際のコンピュータに乗せて移動させるためのプログラミングを必要とする。そのインプリメントのツールとは、プログラミング言語のことである。つまり、病歴検索システムにとって最適な言語はどれであるかという言語の選択が決め手になろう。

本論の狙いは一種のプログラミング言語論である。ただその評価はコンピュータ科学的な範囲に止めず、情報検索の立場から病歴検索システムの事例を取り上げて焦点をおいた。

I. 病歴検索システム

A. 事例1—人工透析型—

1. 目的と特徴

慢性腎炎、腎不全、ネフローゼ症の患者を腹膜灌流や血液透析の方法によって治療することを人工透析と言うが、このシステムは人工透析の検査記録を蓄積してデータベースとする。検査は通常、年間に120回し、1日に時間毎に12回する。情報として重要なのは、時間軸を中心に变化するデータの推移なのである。おのずからデータベースは時系列情報の蓄積と検索を考慮したモデルを検討する。このモデルを TOD (Time Oriented Database) 方式²⁾と呼ぶ。

2. データ項目と属性

人工透析の時系列検査のデータ項目は、体重、血圧、ヘパリン、輸血量、摂取量、補液、排泄量、脱水量、脈搏などで時間単位に1日12回に渡り記録される。その特徴はくり返しのある数値となる。また血液生化学、尿、ガス分析、免疫血清、血漿蛋白、ECGなどの15種の一般検査項目がある。これは頻繁に記録されるものではないが時系列的に蓄積されなければならない。表1は人工透析のデータ項目の一覧を示したものである。これらのデータ項目を整理してみると次の6グループになる。

- (1)患者基本グループ…病歴番号、氏名、生年月日など20項目
- (2)透析基本グループ…透析日、方法、人工腎型など21項目
- (3)透析時系列グループ…時間、温度、回転数、脈搏など12項目
- (4)一般検査グループ…血液生化学、尿、ECG など12

表1 人工透析型のデータ項目 (要約)

グループ	項目名	内 容	データ属性	
			桁	タイプ
患者基本	ID	病歴番号	5	コード
	NAME	患者氏名	20	カナ
	SEX	性別	1	記号
透析時系列	DR	担当医師	可変	自由
	HDDAT	透析日	6	数値
	HDD	透析時間	3	数値
	HDT	透析回数	4	数値
	METH	透析方法	2	コード
	DIAT	人工腎型	2	コード
	DBW	体重 (前/後)	5	数値
	DBP	血圧 (前/後)	3	数値
	DIA	灌流	2	コード
	HEPA	ヘパリン	2	コード
	BTRA	輸血	4	コード
	INTA	摂取量	4	数値
	INFU	補液	4	コード
	URST	排泄量	4	数値
	UFV	限外	4	数値
	ULT	脱水量	4	数値
	DPUL	脈搏	3	数値
	DBT	体温	4	数値

項目

(5)治療グループ…入院経過, 薬歴表, 腹膜灌流記録など12項目

(6)既往歴グループ…腎疾患治療歴など19項目

3. データ構造とデータベースファイル

データ構造とデータベースファイルはデータ項目と属性及び処理の方法によって決る。データベースのモデルは, TOD方式を採用した。TOD方式とは, 1972年にスタンフォード大学の J. Fries らによって提案された。3属性として患者, 時間, パラメータ(項目)を3次的に展開することによって, 時系列的な観点からデータをファイリングする。図1はTOD方式の模型を示している。ある患者の検査データを時系列上に追跡する蓄積と検索には適している。一方, あるデータを中心に該当する患者を検索するといった内容検索には不向きである。TOD方式の人工透析データベースを論理構造によって示すと図2のようになる。図2は患者単位に病歴番号を根として6グループのデータ項目を枝としておらさ

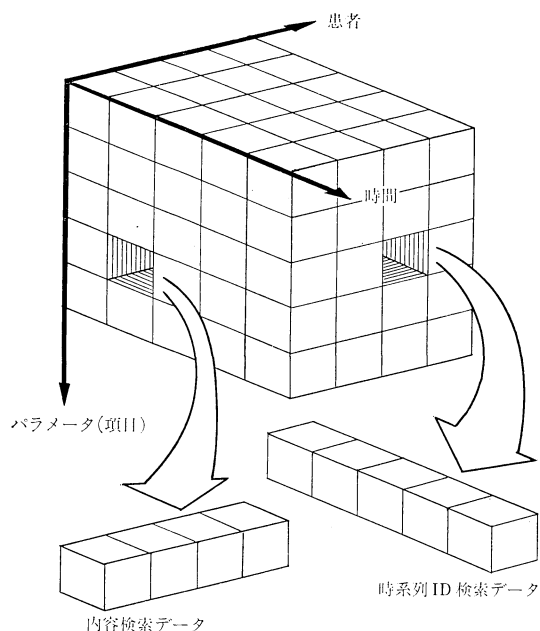


図1 TOD方式の模型

げる。検査データは年間120回の透析基本グループとそれぞれに1日12回の透析時系列グループをくり返し項目としておらさげている。このように樹木状に3次的に成長するデータベースファイルになる。

4. 処理の方法

データは毎日毎時間に刻々と発生する。それをオンライン入力し, データベースに蓄積する。検索と利用はある患者を追跡調査するためにバッチによって, また日常の透析検査と比較検討にはオンラインによる。アクセスの方法は木構造の根に相当するIDから入り, 透析日, 時間を指示することによって目的のレコードに達する。また, 研究目的を主とする処理は, 内容検索と統計処理である。たとえば, GOTの異常値を指定することによって患者情報を出力するとか, 統計的評価のための平均, 標準偏差, 度数分布, 相関の処理である。

5. 要求する機能

この人工透析型をインプリメントするためには, TOD方式を実現できるツール, 木構造データベースファイル, オンライン蓄積と検索, 内容検索, 統計処理などの機能を準備しなければならない。

B. 事例2—脳卒中発症型—

1. 目的と機能

病歴検索システム開発のためのプログラミング言語の評価

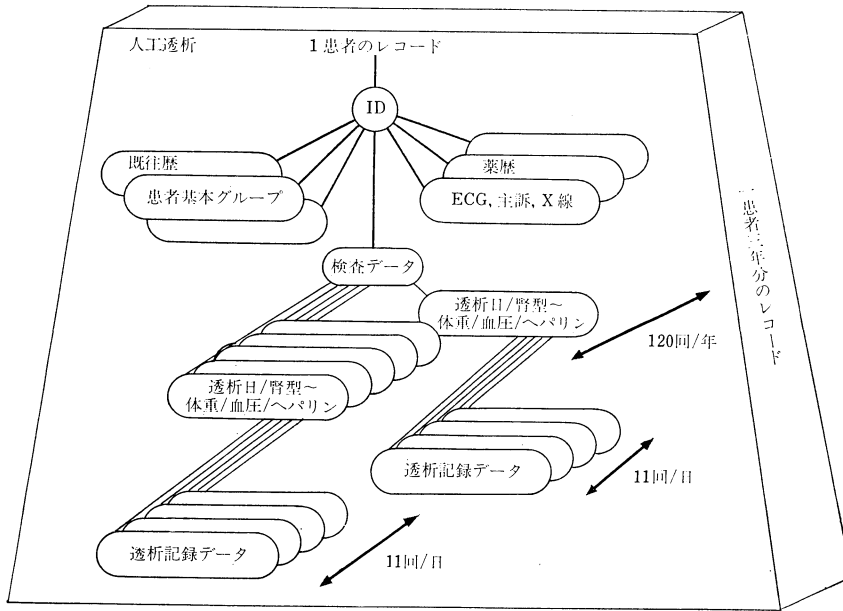


図2 人工透析型 データベースの論理構造

表2 脳卒中発症型のデータ項目

グループ	項目名	内容	データ属性	
			桁	タイプ
機関日時	ORGC	所属機関	5	コード
	REG ORD ATT	受付日時 指示日 発症時刻	8 8 10	数値 数値 数値
患者基本	ID NAME SEX AGE	患者コード 患者氏名 性別 年令	6 可変 1 4	コード 自由 コード コード
頭痛	HAG HAA HAS	頭痛程度 頭痛様相 頭痛状況	2 2 可変	コード コード 自由
嘔気・マヒ	VOM CONS PAG PAT PAS	嘔気 意識状態 マヒ程度 マヒ完成時間 マヒ状況	1 可変 1 2 可変	コード 自由 コード 数値 自由
血圧・呼吸	BP PUL RES RSS TEMP	血圧 脈搏数 呼吸数 呼吸状態 体温	3 3 3 可変 2	数値 数値 数値 自由 数値

治療	TRE PRO DIAG IND	治療 転帰 診断名 指示内容	4 4 4 4	コード コード コード コード
----	---------------------------	-------------------------	------------------	--------------------------

このシステムの目的は、脳卒中の急性期における脳神経外科的治療を補助することと、脳卒中の発生状況を疫学的に把握することである。特定地域内において発症した脳卒中患者は、受持の医院から電話によって連絡される。報告を受けたセンターの職員はオンライン端末からデータを入力し、データベースに蓄積する。要求があれば、センターの医師はデータベースの症例を検索してアドバイスする。また1ヶ月毎に発症月報を作成して、各保健所を通じて医師に配布する。

2. データ項目と属性

脳卒中発症のデータは29項目ある。頭痛、嘔気、意識、運動マヒなど表2に示す項目で、データ属性は程度を表現する独特なコードである。

3. データ構造とデータベースファイル

データ構造は木構造である。加盟医療機関別に発症の報告を受けると、患者をID基点にして発症時刻、指示日時を含む受付日時を根とする図3のような樹木になる。同一患者でも発症をくり返すと同一型の枝を転写する。

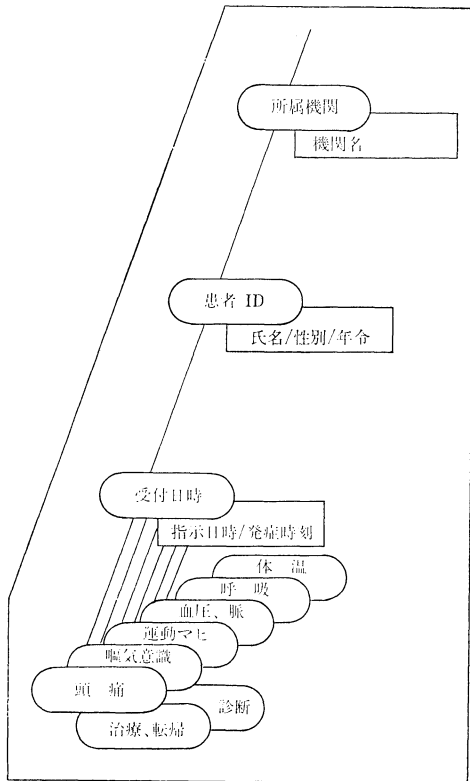


図2 脳卒中発症型データベースの論理構造

4. 処理の方法

発症時刻によってランダムに連絡を受けて29種のデータ項をオンライン入力させる。したがって動的に構築されるファイルとする。

治療、転帰、診断のデータは治療後に一括してコード入力されるが、出力は自然語に変換される。蓄積された情報は月報としてレポート編集の処理となる。またアドバイスと比較検討のためには対話型処理を必要とする。

5. 要求する機能

まず、オンライン処理を必要とする。動的な登録と更新、木構造のファイリングやコードの自然語変換のための記号処理や報告書の作成などを準備しなければならない。

C. 事例3 一疾患名別サマリー型

1. 目的と特徴

中規模な新設医科大学向けの総合的な病歴検索システムである。内科、外科、泌尿器科などの全13科を対象とする研究用途を目的とする。特色は疾患名別にサマリーシートを設計し、個別にデータ項目とチェックタグを設けて記載入力する。これは従来の方式が、患者別にマスターシートを準備して、疾患名を記載していく患者中心の病歴管理なので、全科を対象にすると、記載様式の統一と、標準化を試みることに必ずしも歩調の合うものではない。しかも科別、疾患別に固有の情報の記述が

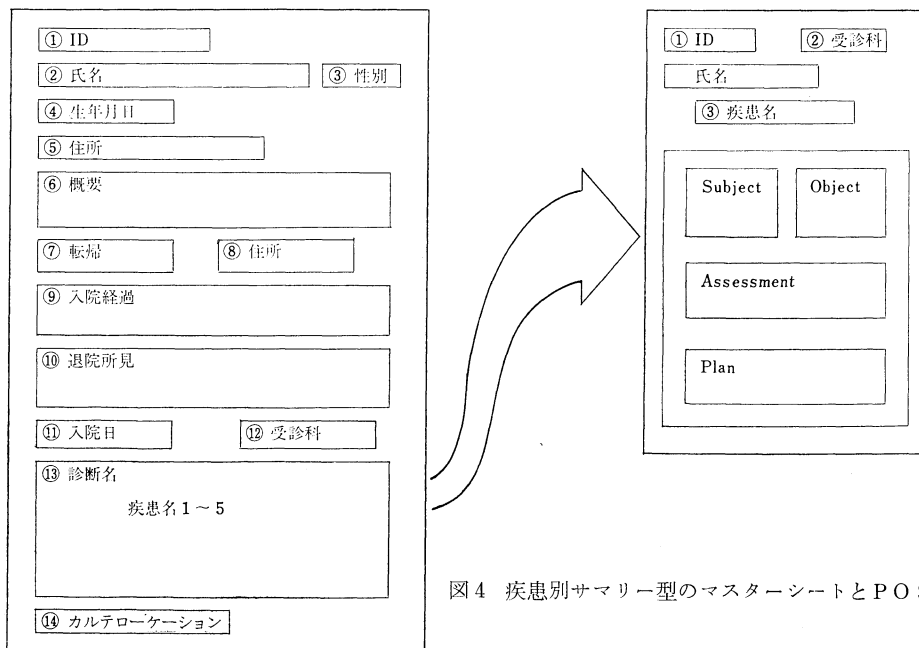


図4 疾患別サマリー型のマスターシートとPOSシート

不十分に終る。これに対し疾患別サマリー型は、疾患名別であるから特有の用語によって深い情報の記述ができる。教育研究向きの病歴として最適とされる。なお、記載方法はPOS方式³⁾を一部採用する。

2. データ項目と属性

科別に平均5件の疾患名を選択し、それぞれ個別にデータシートを準備する。これを疾患別サマリーシートと呼ぶ。このサマリーシートの内容は疾患名によって異なる。図4のように患者マスターシートと関連を持つ。

データの記載はPOS (Problem Oriented System) 方式である。POS方式はweedの提案によるもので、実験ノートをつける時と同じ思考のサイクルにより病歴を記載しようとする。つまり問題の設計、解決手段の選択、実験、成績結果、そして考察のサイクルを通じて、

実験計画を立案するものである。特に問題をSOAPと呼ぶ次の4項目によって簡潔化する。

- (1) S(Subject) : 患者の主訴など患者が直接提供する情報
- (2) O(Object) : 医師や看護婦が検査の結果から取り出す客観的な情報
- (3) A(Assessment) : 医師によるSとOから何をどうするかを判断
- (4) P(Plan) : 処置の内容

3. データ構造とデータベースファイル

構造はインバーテット構造となる。疾患名を索引として抽出し、マスターファイルとチェーニングする。図5はチェーニングの関係を示している。

4. 処理の方法

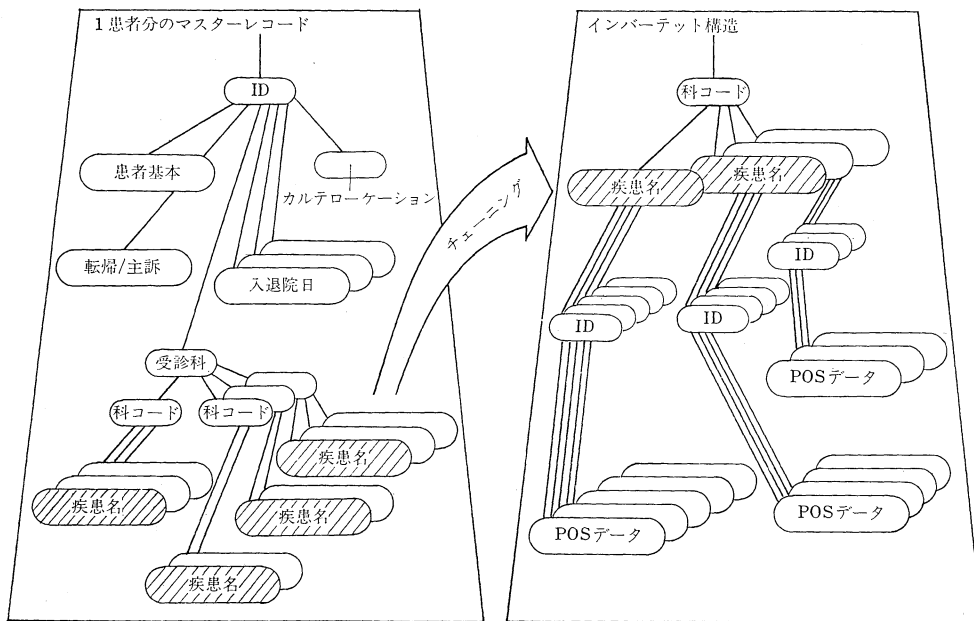


図5 疾患名サマリー型のインバーテットデータベースの論理構造

複雑な処理は入力である。というのは疾患名別のサマリーシートの数科平均5件としても13科では65件にもなり、それに対応して65種の入力処理を設計しなければならない。これではプログラミングが堪らない。そこで入力データの様式と処理の手順をあらかじめ定義しておき、その定義表(ディレクトリー)を参照するだけで65種の内任意の処理プログラムを動かすような工夫をす

る。検索はインバーテット構造を介しての内容検索を必要とする。

5. 要求する機能

以上から整理すると機能は、データ定義、疾患別のインバーテット構造の作成、内容検索、統計処理、POS方式記載の自然語処理などとなる。

II. システムの設計

A. 要求機能とアルゴリズム

3事例の病歴検索システムのそれぞれの要求する機能を整理してみると表3のようになる。さらにデータ属性、データ構造、データ処理の3項目によって分析すると表4のような機能を抽出できる。これを病歴検索システムのためのアルゴリズムと呼ぶ。

表3 要求機能の一覧

機能	人工透析型	脳卒中発症型	疾患別サマリー型
データベースモデル	TOD方式	なし	POS方式
木構造データベース	3次元4階層	3階層動的	多枝3階層
オンライン蓄積と検索	一部バッチ	全てオンライン	検索の一部バッチ
内容検索	多項目論理	シーケンシャル論理	インバーテット論理
大容量ファイル	200MB (100人3年間)	30MB (18,000人)	60MB (50,000人)
記号処理	テキスト操作	コード変換	POSテキスト操作
統計処理	時系列分析	なし	病名統計
インバーテット構造	なし	なし	疾患名の索引
報告書の作成	なし	月報	なし

表4 抽出アルゴリズムの一覧

データ属性	データ構造	データ処理
TOD方式によるくり返し数値データ	IDを根とする木構造	記号処理
連続量、離散量の数値	時系列の3次元展開	入力データ定義処理
POS方式による自然語テキスト	オンライン動的構造	オンラインとバッチの混在
可変長の文字とコード	疾患名によるインバーテット構造	オンラインデータベースファイリング
数値と文字の混在	木型と線型のアクセスメソッド	頻頻な更新
「あり」と「なし」の2値コード		内容検索
病名、所見、転帰のコード		多項目の論理演算
容態所見のキーワード		数値演算
		コードテキスト変換

B. アプリケーションの構成

次に目的と用途別に機能とアルゴリズムをまとめるとアプリケーションプログラムが構成できる。

1. DDP (Data Description Program)

疾患別サマリー型のためのもので個別に設計し準備されている入力データ項目とその様式をデータ定義する。たとえば入力画面の種類、画面内の行と桁の位置、データ項目の名前と値、チェックの方法そして入力後のデータ

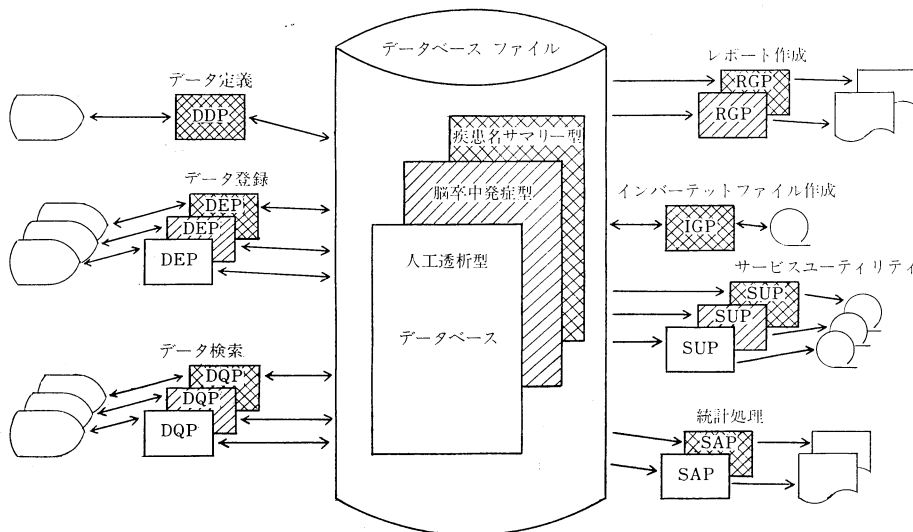


図6 アプリケーションプログラムの全体構成

ベース構造との対応づけである。DDPの働きはDBMSのスキーマ⁴⁾とかマッピングに似ている。

2. DEP (Data Entry Program)

データを端末からオンラインに入力し、データベースに登録する。人工透析型、脳卒中発症型、疾患別サマリー型と入力画面の様式は異なっているが内部処理の手順は共通化できる。ただしデータは検査のような数値であったり、程度を表現するコードであったり、診断名、POS記載の自然語であったりする。記号処理を有効に活用する。

3. DQP (Data Query Program)

データベースファイルに対する問合せ検索プログラムである。木構造に蓄積されたデータを対象にIDを糸口に枝をたどり目的とするデータ項目にアクセスし、記号処理を働かせる内容検索をする。また複数個の条件を論理演算により探索する。

4. RGP (Report Generation Program)

検索回答を一定の書式に編集し出力するプログラムである。ID順や患者氏名順の配列のためのソートや書式のための桁揃いがある。

5. IGP (Inverted Generation Program)

疾患別サマリー型はインバーテッド構造を採用する。その生成と維持のプログラムである。まずID中心の木構造のマスターファイルを構築した後に、含まれる疾患名を抽出して索引づけをする。

6. SAP (Statistical Analysis Program)

統計処理のプログラムのことである。平均、分散、相関などの基本的な統計のために数値演算の機能を持つ。

7. SUP (Service Utility Program)

データベースファイルの管理と保全を目的とするプログラムで、磁気テープへの転写とファイルへの再現の機能を持つ。

以上のアプリケーションプログラムの全体構成を図6に示した。

III. システムの開発

3事例から要求機能を整理し、アルゴリズムを抽出して用途別にアプリケーションプログラムを構成した。これまでがシステムの設計である。システムの開発はインプリメンテーションと呼ぶ、コンピュータ上にソフトウェアを装備する作業である。それには作業者とソフトウェアツールを十分に検討しなければならない。

A. インプリメンテーションの条件

条件の第1は、ミニコンピュータによるエンドユーザ開発指導型とする。ハードウェアの低価格はミニコンピュータの出現によって実現されたものの、ソフトウェアの価格はいぜんとして高い。むしろますます高価になる。ソフトウェアの費用というのは人件費のことなので無理もない。専門のプログラマに頼るからにはしかたがない、安くあげるには、OSと高級言語といった必要最小限のツールのみの提供を受けて自主開発することになる。

ミニコンピュータ時代の情報処理はプロ（職業的）グラマに代ってアマ（素人）グラマ⁵⁾の活躍の場なのである。以上を整理すれば、条件とは

- (1)できるだけ病歴検索システムの要望とアルゴリズムを満たすソフトウェア
- (2)アマグラマ向きのプログラミングシステムであること
- (3)日曜大工的に素人が手造りできること
- (4)ミニコンピュータによった適性規模なハードウェア構成であること
- (5)対話型に試行錯誤のプログラミングができること
- (6)オンラインでデータベース指向であること
- (7)コンパイラかインタプリタ水準の高級言語を装備していること

である。特に(7)の影響は大きく、システム開発の決め手になる。そこで、実際にミニコンピュータに装備されている高級言語はどのような種類のものがあって、どの程度まで病歴検索システムのインプリメンテーションに適するのかを評価する。

B. 高級言語の選択

高級言語という定義は、習得、プログラミング、保守と文書化、変換などの容易性を持ち、アプリケーション向きの表記法を持つもの⁶⁾とされる。ここではミニコンピュータ向きの高級言語に絞る。それぞれのプロフィールを次に示す。ただしFORTRANとCOBOLは常識にもなっているので省いた。

1. ミニ PL/1

PL/1はFORTRAN, ALGOL, COBOLのすぐれた機能や特徴を集大成して誕生した。アセンブラに頼らなければならないようなビット操作を含んでいる。言語の規模が大き過ぎてミニコンピュータにとって荷が重いことからサブセットの仕様を持つミニ PL/1⁷⁾がある。

2. APL

簡潔性、論理性、機能に多様性を備えたユニークな言語であって、これまでのCOBOLなどのような英語的

な表記と違った Iverson 記法を基礎とする特殊な記号を使う。対話型インタプリタの本命とされる。⁸⁾

3. PASCAL

アルゴリズムの記述に適すと言われる ALGOL の仕様にデータ構造の機能を加えたポケット版の ALGOL である。⁹⁾

4. SNOBOL

人文科学の分野におけるコンピュータ応用を目的に考案された記号処理を得意とする言語である。¹⁰⁾

5. LISP

リスト処理を目的に設計し開発され、帰納的な関数の理論に基づく数学的なエレガンスがあるとされる。¹¹⁾ 最近は人工知能の研究に実績を認められている。

6. MUMPS

医療情報処理の現場で考案された問題向き言語とされるが、非医療界においても応用は多い。¹²⁾

7. BASIC

FORTRAN に酷似しているが、そもそもの発想は時分割オンライン用の対話型言語である。¹³⁾

C. 評価と考察

評価の観点には、3つの型の病歴検索システムのアプリケーションのプログラミングである。そのために最適な高級言語はどれであるかを評価し選択する。評価の方法はアプリケーションの要求機能とアルゴリズムを集約して設定した評価項目によって、高級言語を対象にそれぞれの満足の度合を吟味する。

1. オンライン機能と対話型プログラム

最近のアプリケーションはバッチ処理は稀なくらいでほとんどはオンライン処理である。またプログラミングもカードによって入力しデバッグをする一昔前の手順は不経済である。COBOL, FORTRAN はカード入力でバッチデバッグの旧世代言語である。これに対し、BASIC, APL, MUMPS などはオンライン対話型プログラミングを前提にする新世代言語である。

2. データ構造の表現とデータベース機能

情報検索はファイル指向のアプリケーションの一種と言えるくらいにデータ構造の多種多様の表現能力を要求する。ディスク装置をファイルとするとランダム、シーケンシャル、インデックスシーケンシャルなどの基本的なアクセスメソッドとその応用によるデータベース機能は不可欠である。LISP はリスト表現として十分なデータ構造の表現能力を持つ。しかし残念なことにデータベースの概念を持っていない。MUMPS は木構造のデータ

ベース機能を強力なアクセサリーにしている。FORTRAN, BASIC などは乏しい。インバーテッド構造はファイルの応用である。

3. 記号処理と混合演算

自然語などの可変長の文字列を連結、分離、抽出、照合といった文字操作を記号処理と言う。混合演算はデータの値の違い、たとえば文字ならば記号演算を、数値ならば数値演算をするが、その混合した文字列、50MG, 10CMなどを数値と単位を示す文字とをそれぞれ分離して処理する。LISP, SNOBOL は記号処理を得意とするし、MUMPS はそれに加えて混合演算を得意とする。

4. 帰納機能と間接実行

帰納とはプログラムが自分自身を呼ぶといった回帰的なアルゴリズムを意味する。LISP は最適である。間接実行はデータをプログラムとみなし実行できる機能である。MUMPS, SNOBOL などのインタプリタ言語は得意である。これに対し COBOL, FORTRAN, BASIC などは不得手である。

5. 数値演算と論理演算

統計処理は平均、分散、相関などの数値演算である。FORTRAN, APL などの算法言語は強力である。論理演算は内容検索、多項目の組合せ検索には必要である。どの言語も装備している。

6. バッチ処理とソート機能

報告書の作成や統計処理後の回答はバッチ処理になる。ソートやレポート機能の程度が目安となる。バッチ処理は旧世代言語である COBOL, FORTRAN が得意である。

7. 使い勝手とエンドユーザ指向

アマグラマ向きに簡単な言語の仕様を持っているか、使い方に不便さがないかなどを目安とする。自然語的な表現では COBOL, BASIC, MUMPS などが優れている。APLは全てを特殊な記号にしていることも慣れると便利とされる。

8. 大規模プログラムと効率

本格的なアプリケーションはいくつものサブプログラムから構成される大規模プログラムになる。また実行効率も速くなければ実用に適さない。この点、LISP, SNOBOL, PASCAL などは趣味の言語とされる。

以上の評価を一覧し得点によって示したものは表5である。得点の意味は、5: 非常によい、4: よい、3: 普通、2: どうか、1: よくない、0: なし、である。

病歴検索システム開発のためのプログラミング言語の評価

表5 高級言語の評価表

機 能 \ 言 語	ミニ PL/1	PASCAL	COBOL	FORT RAN	BASIC	APL	MUMPS	LISP	SNOBOL
オンライン機能	3	1	3	0	5	5	5	0	0
対話型プログラム	0	0	0	0	4	5	5	0	0
データベース機能	3	2	3	2	1	0	4	1	1
インバーテットファイル	1	0	2	1	0	1	2	0	0
記号処理	2	2	1	0	1	3	4	5	5
混合演算	1	1	0	0	1	4	5	4	4
帰納機能	3	4	0	0	0	3	3	5	5
数値演算	5	4	4	5	4	5	2	3	3
論理演算	3	3	2	1	1	3	3	3	3
バッチ処理	4	3	5	5	4	0	1	3	3
使い勝手	3	3	3	3	4	5	4	1	1
大規模プログラム	4	3	5	5	1	0	1	0	0
効率	4	4	4	5	3	2	2	1	1
機密保護	1	0	1	0	0	0	2	0	0
	37	30	33	27	29	36	43	26	26

表6 MUMPS アプリケーションのプログラムサイズ一覧

アプリケーション 型	DDP	DEP	DQP	RGP	IGP	SAP	SUP	合 計
人工透析型	—	88	53	—	—	74	50	265
脳卒中発症型	—	40	50	43	—	—	50	183
疾患別サマリー型	50	50	60	—	48	68	73	349

評価の結果は表5によると、1位はMUMPSで、2位ミニ PL/1、3位APL、以下COBOL、PASCAL、BASIC、FORTRAN などとなった。

IV. MUMPS の採用

A. MUMPS によるプログラミング

3事例のアプリケーションをMUMPSによってプログラミングしてみた。そのプログラムのコーディングのサイズを表6に示した。

表6はアプリケーション単位にステップ数で一覧している。見て分るように平均50ステップである。ステップとはFORTRANなどのステートメントに相当する実行の最小単位のことである。ステップ数は極端に少ない。3事例はそれぞれ合計で265、183、349のステップ数でプログラミングされている。参考までにCOBOLによって同一内容をコーディングすると約50倍にもなる。この秘密を明らかにするには、MUMPSを十分に説明しな

ければならない。

B. MUMPS の誕生と背景

MUMPS とは、MGH (マサチューセッツ総合病院) Utility Programming System の略で1967年に医療情報処理の実践の場¹⁴⁾において誕生した。その背景の医療情報の特質を言語の設計に反映した。すなわち、診断とか治療には洞察、記録、検索、比較、選択、判断、そして観察といった患者を原点とするめまぐるしい情報の流れがある。その中から多くの兆候と事実をつかみ病因と症状を的確に把握し治療の方針をたてなければならない。具体的には言語の機能として、次を組み込んだ。

- (1)オンライン対話型のインタプリタ
- (2)コマンド型の高級言語
- (3)木構造データベースファイリング
- (4)記号処理
- (5)ミニコンピュータ装備

C. MUMPS の仕組

MUMPS のハードウェアはメモリ容量96~256KB のミニコンピュータにより構成される。

ソフトウェアの構成は、人工透析型を例とすると図7のようになる。図7はメモリマップを示し、まずモニタ、IOCS、インタプリタ、データベース管理から構成する MUMPS の OS がある。その下にアプリケーションが走るパーティションと呼ぶ専用領域がある。

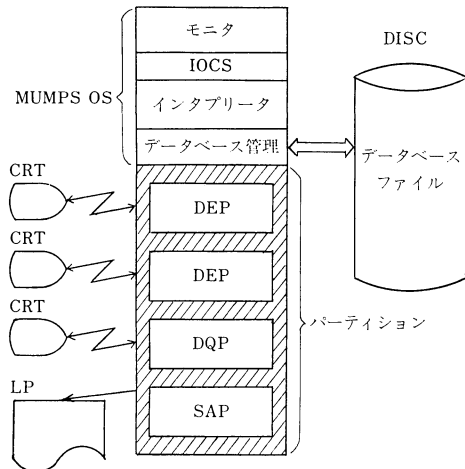


図7 MUMPS マシンのソフトウェア構成

DEP が2端末のデータ入力に対応して、2本同時に走り、DQP と SAP が併用して時分割に稼動する。このような専用ハードウェアとソフトウェアを持つシステムを MUMPS マシンと呼ぶ。

D. 言語とプログラミング

MUMPS の言語は国際的な統一と標準化を受けていて、標準 MUMPS 文法¹⁵⁾を制定されている。言語はコマンドと呼ぶ指令動詞とファンクションと呼ぶ記号処理や統計処理のための組込関数と、オペレータから構成する。

表7、表8は言語の一覧である。プログラムの単位をラインと呼び図8を言う。プログラミングは宣言無用、混合演算、スパース型データベース定義、多次元添字、間接技法など独特である。¹⁶⁾図9はDEPの簡単なコーディング例を示している。端末から病歴番号 (ID)、患者氏名 (NAME)、性別 (SEX)、生年月日 (BDATE)、疾患名 (DIAGNOSIS) を順に入力し、IDを根とする木構造のデータベースに蓄積する。プログラム番号の1.10から1.50までは入力データの読取りである。2.10から2.40まではSETコマンドによるデータの蓄積である。ADBはデータベースファイルの名前のことであり、^記号はグローバル変数と呼ぶ。

G 3.10 : I < O S ^ADB(1,I) = \$P(A,"*",1)

コマンド アーギュメント 論理式 コマンド グローバル 添字 ファンクション アーギュメント
変数

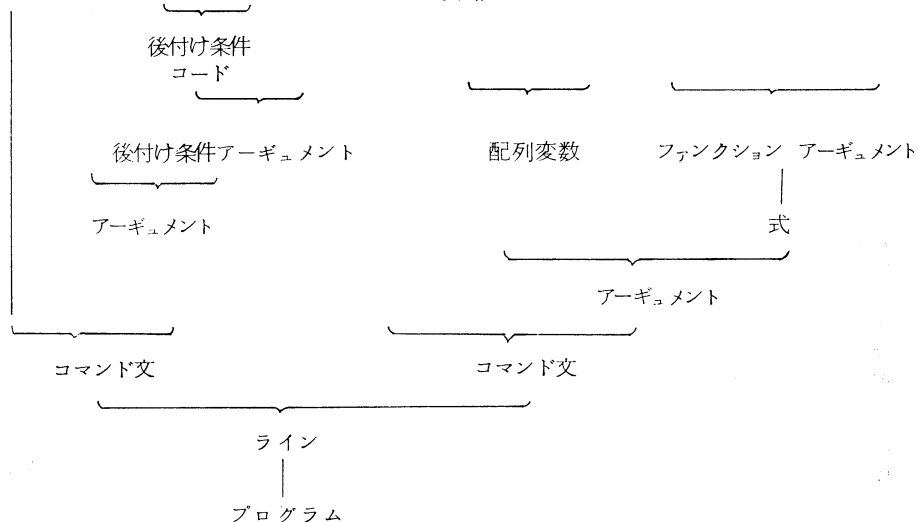


図8 MUMPS のプログラム構成例

病歴検索システム開発のためのプログラミング言語の評価

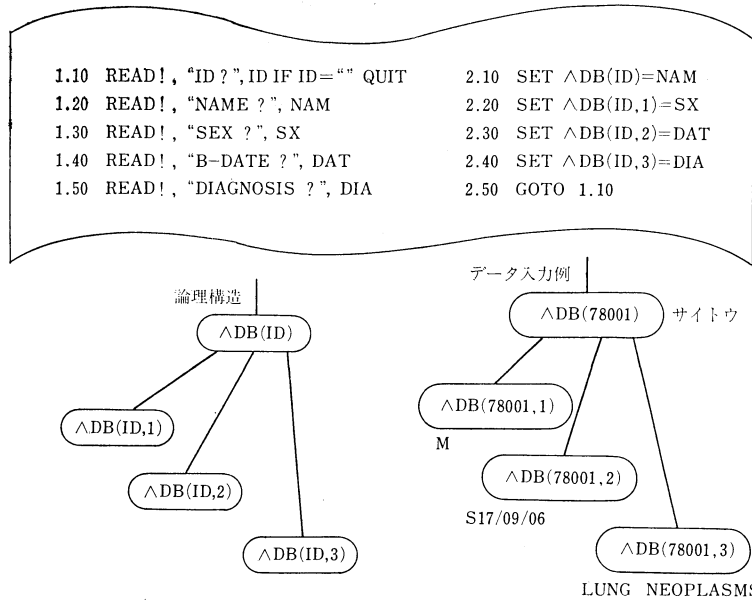


図9 アプリケーション プログラミングの例

表7 MUMPS 言語の一覧 (1)

コマンド

種 類	コマンド名	機 能	例
変数操作	SET	変数値セット	S A=5
	KILL	変数削除	K A
実行制御	GOTO	無条件分岐	G 2:10
	IF	条件分岐	I A<B
	DO	サブルーチン起動	D 3
	FOR	くり返し	F I=1:1:10
	XECUTE	間接実行	X A, B
	QUIT	ルーチン終了	Q
	HALT	プログラム終了	H
	ELSE	条件実行	E D 5
	HANG	時間付実行	H 360
	ZRUN	他プログラム起動	ZR PRO 1
入出力制御	READ	データ入力	R"ナマエ", NAM
	WRITE	データ出力	W!, A, B
	PRINT	プログラム出力	P
	OPEN	デバイス占有	O 11
	CLOSE	デバイス解散	C 11
	USE	デバイス指定	U 10:30
	LOCK	アクセス権指定	L TID

システム制御	FILE	プログラム登録	F PRO
	LOAD	プログラムロード	L PRO
	ERASE	プログラム削除	E 5.50
	MODIFY	プログラム修正	M 2.01:1 "1" YOU
	BREAK	実行中断	B :A<O
	GO	中断プログラム続行	G
	ZSEARCH	指定テキスト出力	Z S 2:5, 15:N=
	ZCHANGE	指定テキスト交換	ZC 2:5:N=/NAM
	VIEW	コアの参照	V VBF+I:O

表8 MUMPS 言語の一覧 (2)

文字オペレータ

記号	機 能	例
=	完全一致	A="MUPS"
[包含	"ABC"["B"
]	後続	"A"]"B"
-	連結	A-B
?	パターン点検	"123/AB"?3D/2A

ファンクション

種 類	ファンクション名	機 能
変 数 処 理	\$DATA \$NEXT \$ZQUERY	データタイプチェック 昇順添字探索 並び添字探索
文 字 処 理	\$LENGTH \$FIND \$ASCII \$PIECE \$EXTRACT \$JUSTIFY \$CHAR \$TEXT	ストリング長算出 文字位置算出 文字の数値化 文字の抽出 パターン抽出 文字編集 数値の文字化 ラインのデータ化
数 値 処 理	\$FSQUAR \$FEXP \$FLOG \$FALOG	平方根 指数 自然対数 常用対数
その他	\$SELECT \$VIEW	論理値選択 コアの読出し

添字は構造とディスク装置上におけるデータのアドレスを示す。ADB (ID) のように ID の内容によって根のアドレスを決める。そして次々に枝を作り、患者名、性別、生年月日、疾患名を蓄積する。

おわりに

本論は情報検索の立場から実際の病歴検索システムの設計と開発を例にとり、プログラミング言語の評価を試みた。データ構造論やオンライン情報検索論などのように情報検索のプログラム言語論を意図した。

結論として MUMPS などの新世代の高級言語の誕生は、これまで大規模なアプリケーションの範ちゅうに属していた病歴検索システムを非常にコンパクトにまとめあげてことを示した。十分にアマグラマなどによる手造りのプログラミングにより達成できる範囲になった。この決め手は高級言語の選択なのである。

MUMPS マシンなどの高級言語マシン¹⁷⁾の出現の背景は、L S I 技術の進歩と成果による。

著者の経験をふり返ってみると、5 年前までは COBOL によって 2, 3 千ステップになるアプリケーションはさらにあった。さらに 10 年前には 1 万ステップにもなるアセンブラのプログラミングも常識であった。その作成は

1 年くらいになることは当然とされた。プログラミングの内容は、記号処理、論理演算など MUMPS の 1 オペレータやファンクションに相当する機能プログラミングすることには大半を費やした。昔に比較すると驚異である。

最近のミニコンピュータは高級言語と一体となって一昔前の中大型機並の実力を持つ。

今後はこれまでの高価で不経済な大型コンピュータによる集中処理から、低価格で高能率のミニコンピュータによる分散処理に移行する。情報検索の技術も分散処理に対応したものを次々に提案していくべきであろう。

- 1) 澤井清. "病歴管理 展望," 医学図書館, vol.20, no. 3, 1973, p. 210-233.
- 2) Fries, J.D. "Time oriented patient records and a computer databank," Journal of American Medical Association, 222, 1972, p. 1535-1542.
- 3) 日野原 重明. POS 医療と医学教育の革新のための新しいシステム. 東京, 医学書院, 1975. 144p.
- 4) CODASYL Data Description Language Committee. NBS Handbook 113. 1974. 105p.
- 5) 木村 泉. "プログラミング方法論の問題点," 情報処理, vol. 16, no. 10, 1975, p. 841-847.
- 6) サメット, E. 竹下 享訳. プログラミング言語ハンドブック, 東京, 日本経営出版会, 1971. 885p.
- 7) 大筆 豊. "ミニコン用 PL/1 サブセット TPL-40," 情報処理, vol. 19, no. 5, 1978, p. 406-411.
- 8) 竹下 享. "APL," 情報処理, vol. 19, no. 1, 1978, p. 78-85.
- 9) 和田 英一. "プログラム言語 PASCAL," bit, vol. 10, no. 1, 1978, p. 25-31.
- 10) 岸田 孝一. プログラミング SNOBOL 3 入門. 東京, 日本生産性本部, 1972. 234 p.
- 11) 雨宮 綾夫. LISP とその応用例. 東京, 産業図書, 1972. 280p.
- 12) 若井 一朗. 標準マシンの言語マニュアル. 東京, コロナ社, 1977. 172p.
- 13) ファリーナ M. 著, 関根 智明訳. タイムシェアリングプログラミング BASIC. 培風館, 1969. 160p.
- 14) Barnett, O. "Design and implementation of clinical data management system," Computers and biomedical research, 2, 1969, p. 469-485.
- 15) O' Neill ed., MUMPS Language Standard, NBS Handbook 118 1975. 64p.
- 16) 齊藤 孝. "連系操作作用の新言語 MUMPS," 計量国語学, vol. 11, no. 5. 1978, p. 198-205.
- 17) 島田 俊夫. "高級言語マシン," 情報処理, vol.18, no. 4, 1977, p. 386-394.