

プログラム仕様記述法PSDMを用いた プログラム量産化

橋本正明 伊吹公夫 (NTT電気通信研究所)

1. はじめに

情報化社会の進展にともない、ソフトウェア需要がますます増大し、ソフトウェア分野にも、経済的に量産化を達成できる環境が成熟するものと予想される。そのため、量産化投資に十分なコストをかけても、全体としてソフトウェア生産コストを下げる技術が望まれて来る。

筆者らは、POS(Point Of Sales)等の業種・事業規模等で分けた領域毎に業務知識を知識ベース化して、その知識ベースから多数の業務プログラムを合成するためのプログラム量産化技術を研究中である。そして、計算指向 EAR(Entity Attribute Relationship) モデル¹⁾に基づくプログラム仕様記述法PSDM(Program Specification Description Method)²⁾を用いた量産化の原理が説明されてきたので報告する。

2. PSDMを用いたプログラム量産化の考え方

PSDMを用いた量産化の考え方を以下に述べる。

(1)量産化の基本原則

本プログラム量産化とは、POS等の業種・事業規模等で分けた領域毎に業務知識を知識ベース化して、その知識ベースから多数の業務プログラムを合成することをいう。プログラム合成とは、知識ベースからプログラム仕様を自動抽出して、その仕様からCOBOL等のプログラムを自動生成することをいう。図2-1は在庫、製品、売上、顧客、請求等に関する知識ベースから、情報システムA,B,C,...上の在庫、販売、請求等の多数の業務プログラムを合成することを示す。

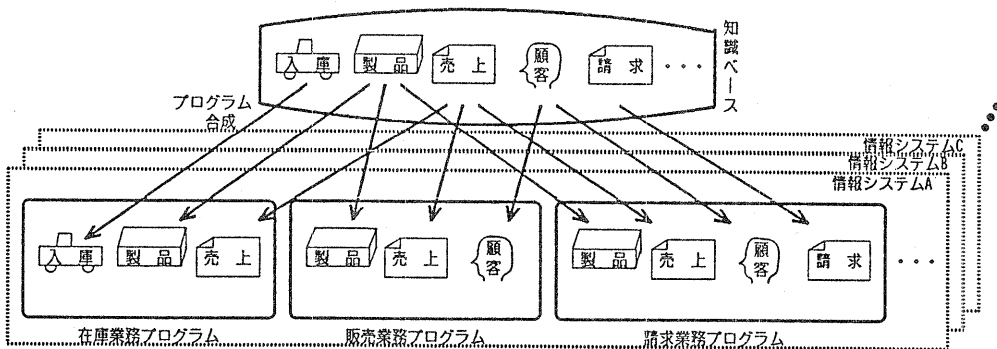


図 2-1 量産化

図 2-2を用いて本量産化の基本原則を説明する。図 2-2に示す業務知識において、①～③はおのおの断片的な知識であるが、④も加えて一体化した知識となる。量産化技術においては、一体化した知識①～④に相当する(1)～(4)を知識ベースへ格納する。したがって、プログラム作成者がプログラムの入出力のみを指定すれば、その入出力関係を表すプ

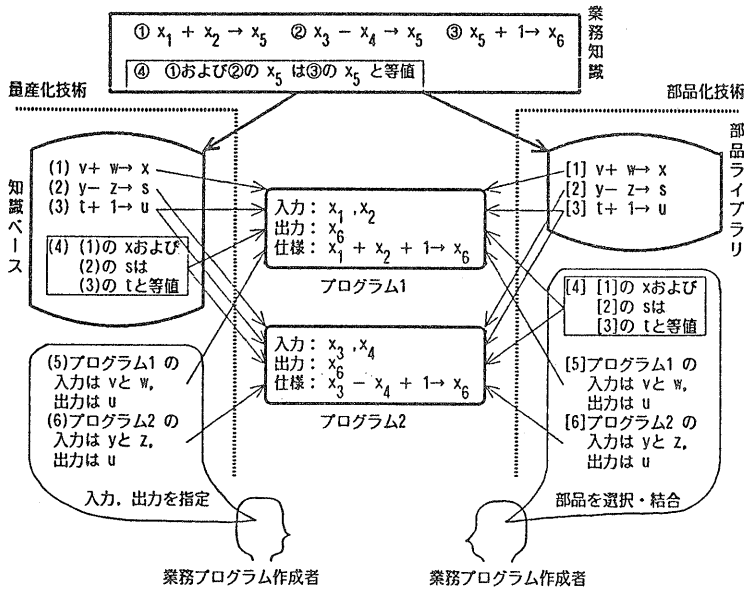


図 2-2 量産化の基本原則

プログラム仕様を知識ベースから抽出でき、プログラムを合成できる。

一方、部品化技術においては、断片的な知識①～③に相当する [1]～ [3]のみを部品ライブラリへ格納する。したがって、プログラム作成者はプログラム単品毎にその仕様や構造を設計の上、[1]～ [3]の部品を選択・結合してプログラムを作成しなければならない。

量産化技術と部品

化技術の本質差は、上記のように断片的な知識を一体化するための知識④を計算機処理するか否かにある。また、量産化技術においては、プログラム単品毎にその仕様や構造を設計することは不要なので“量産化”という。

(2)業務知識の形式的な記述

業務知識を知識ベースとして計算機に蓄積し、その業務知識から計算機を用いてプログラムを合成するには、計算機処理可能な形式で業務知識を記述しなければならない。そのため、業務知識を以下の 2つに分けて、おのおのに適した記述法を設ける。

- ①製品、売上、顧客、請求のような“概念”に関する知識。
- ②売上傳票、請求伝票のような“帳票形式”に関する知識。

上記①の“概念”に関しては、製品や顧客のような“物”，および、売上や請求のような“事”，つまり，“事物”を単位として概念を捕える。また、売上は製品や顧客との関係を考慮して意味を持って来るように、概念は相互に関係を持っている。この関係を“事物相互の関係”として捕える。そして、計算指向 EARモデルを用いて概念に関する記述を行なう。上記②の“帳票形式”に関しては、COBOLのデータ部とおおむね同じ考え方で記述する。以上の“概念”および“帳票形式”に関する記述法は次章のPSDMで定める。

(3)プログラム・インプリメント情報

プログラム合成時に指定するプログラム入出力を含めた以下のプログラム・インプリメント情報も知識ベースへ加える。したがって、プログラム作成者は業務プログラム名のみを指定して、プログラムを合成できる。

- ①合成対象の全業務プログラム名。
- ②各業務プログラムの入出力を示す入出力ファイル定義
- ③業務プログラムの実行効率（実行時間、メモリ量）向上に有効な情報。

(4)ソフトウェア・ライフサイクル

ある領域の知識ベースを作成する場合、最初からその領域全体を見通して知識ベースを

作成することは困難である。このため、以下に述べるようなライフサイクルを経て知識ベースを洗練していく。

最初は典型的な情報システムを事例にとり、分析を加えて知識ベースを作成する。それから、新たに開発する情報システムへ知識ベースを適用しながら、知識ベース自体を改編していく。改編に当っては業務知識を以下の3つに分けて、①②を知識ベースへ反映する。

- ①業務知識原型の固定部。
- ②業務知識原型の代替部。
- ③業務知識の情報システム固有部。

個々の情報システム開発時に上記②に関しては選択を行ない、③に関しては業務知識原型へ変更を加える。そして、開発対象情報システムに適合した業務知識記述を得る。

(5) 量産組織

ハードウェアの量産工場において技術開発と製造の2つの組織があるように、ソフトウェアの量産化においても組織上の配慮が必要である。

量産組織においては、個々の情報システム製造グループの他、適用領域毎に知識ベース開発グループが必要となる。知識ベース開発グループは知識ベースの作成、改編を行なうとともに、知識ベースの適用を高めるように、情報システム製造グループに属する業務プログラム作成者を指導する。量産組織を図2-3に示す。

(6) 量産化の適用領域

量産化においてはプログラム合成システムの開発、知識ベース作成グループの設置、および、知識ベースの作成という初期コストが必要なため、従来のプログラム作成法とのコスト比較は図2-4のようになる。同図から解るように、量産化は多くの業務プログラムを作成する領域で特に有効となる。そのため、適用領域の選択に注意しなければならない。

現在、量産化の最適領域として、システム数が多く、多数の業務プログラムが必要なPOS領域があげられる。今後の情報化社会進展にともない、量産化の有効な領域がさらに出てくるものと考えられる。

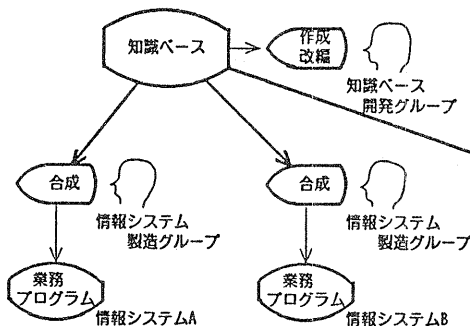


図 2-3 量産組織

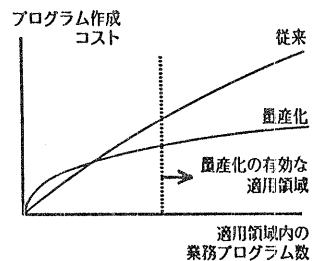


図 2-4 量産化の適用領域

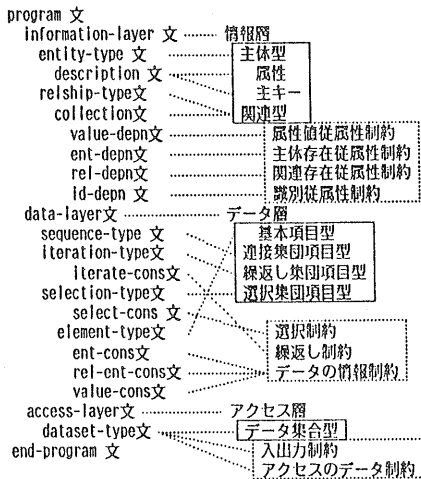
3. プログラム仕様記述法PSDM

PSDMは、業務知識に現れる概念や帳票形式を用いたプログラム仕様記述法である。従来は順編成ファイルを使用するプログラムを対象にして研究してきたので、その範囲のPSDMについて述べる。

3.1 PSDMの階層構成

PSDMにおいては、プログラム仕様を情報層、データ層およびアクセス層に分けて階層的

に記述する。情報層では業務知識に現れる概念に関して記述する。データ層では業務知識に現れる帳票形式を含めた入出力データ形式に関して記述する。アクセス層では入出力ファイルに関して記述する。



(1) PSDL文の種類 (2) PSDMの記述要素

図 3-1 PSDM記述要素とPSDL文

PSDM各層の記述要素と、PSDM適用の実験用プログラム仕様記述言語PSDL(Program Specification Description Language)の文を図 3-1に示す。また、請求書作成プログラムの仕様を図 3-2に例示する。

3.2 情報層

業務知識に現れる概念のうち、事物を“主体”(Entity)といい、事物相互の関係を“関連”(Relationship)という。そして、計算指向 EARモデルに基づく以下の記述要素を用いて、概念に関する記述を行う。

主体型は主体の集合を定める型である。たとえば、図 3-2に示す主体型“製品”はテレビ、ラジオ、ステレオ等の個々の主体の集合を定める型である。属性は個々の主体の性質を表す項目であり、同じ型に属する主体は同じ属性を持つ。たとえば、

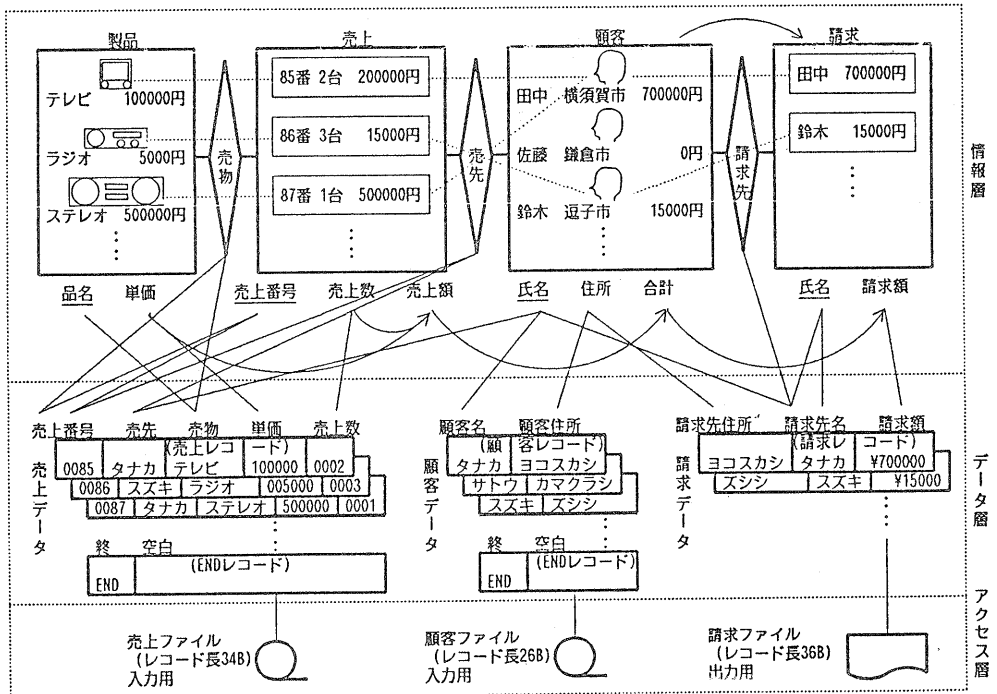


図 3-2 請求書作成プログラムの仕様

主体型“製品”に属する主体は属性“品名”および“単価”を持つ。そして、個々の主体の性質は属性の値で表す。たとえば、テレビという主体の属性“単価”の値は 100,000円である。主体型の中で個々の主体を識別するのに用いる属性を主キー属性という。たとえば、主体型“製品”に属する主体の主キー属性は“品名”である。そして、その属性値“テレビ”、“ラジオ”、“ステレオ”等で個々の主体を識別する。

関連型は関連の集合を定める型である。個々の関連は、関連で関係付けている主体の主キー属性値の組で表す。たとえば、図 3-2に示す関連型“売物”は、主体型“製品”に属する主体と、主体型“売上”に属する主体を関係付けている関連の集合を定める型である。そして、“(テレビ, 85番)”という関連は、テレビという主体と85番売上という主体を関係付けている。

属性値従属性制約はある主体の属性値に基づいて、その主体と関連で関係付けている他の主体の属性値を得るための計算方法を定めるものである。たとえば、図 3-2に示す主体“売上”の属性“売上額”の値は、関連“売物”が一對一に対応付けている2つの主体“売上”と“製品”に関して、属性“単価”の値と“売上数”の値の積として得ることを定める。

関連存在従属性制約はいくつかの主体の存在に基づいて、その主体間に存在する関連を得るための計算方法を定めるものである。たとえば、図 3-2に示す関連型“請求先”においては、主体“顧客”の主キー属性“氏名”の値と、主体“請求”の主キー属性“氏名”の値が等しければ、両主体間に関連が存在することを定める。

主体存在従属性制約はある主体の存在に基づいて、他に存在する主体、および、その両主体間に存在する関連を得るための計算方法を定めるものである。たとえば、図 3-2に示す主体型“顧客”に属する主体の属性“合計”の値が正ならば、主体“請求”が存在すること、および、その両主体間に関連“請求先”が存在することを定める。

3.3 データ層

業務知識に現れる帳票形式を含めた入出力データ形式は、COBOLのデータ部とおおむね同じ考え方で、以下の記述要素を用いて記述する。

COBOLの基本項目は基本項目型で記述する。基本項目型データ内の記号の並び方はCOBOLのPICTURE句で定める。COBOLのOCCURS句を用いて定める集団項目は繰返し集団項目型で記述する。COBOLのREDEFINES句を用いて定める集団項目は選択集団項目型で記述する。また、OCCURS句、REDEFINES句は用いずに、同じレベル番号で続けて定める集団項目は直接集団項目型で記述する。以上の型をまとめてデータ型という。

繰返し集団項目型におけるデータ繰返し個数の決定方法は繰返し制約で記述する。繰返し個数は、指定した主体型に属する主体の個数、あるいは、指定した基本項目型データを主キー属性値として持つ主体の指定した属性の値に等しいものとする。また、指定した基本項目型のデータ、あるいは、指標値に対してある述語が真になれば、そのデータまたは指標値の直前で繰返しが終るものとする。

選択集団項目型におけるデータ型の選択方法は選択制約で記述する。選択対象データ型へ対応させた述語を以下の値へ作用させ、ただ一つ真となる述語へ対応するデータ型を選択する。①指定した主体型に属する主体の個数。②指定した基本項目型データを主キー属性値として持つ主体の指定した属性の値。③指定した基本項目型のデータまたは指標の値。

入出力データは情報層で定めた型に属する主体や関連を表しているものとする。したがって、データの情報制約を用いて、基本項目型や指標と、情報層で定めた型を以下のよ

うに対応させる。①ある主体型に属する主体を表す基本項目型および指標は、その主体型の主キー属性に対応させる。②ある関連型に属する関連、および、その関連が関係付けている主体を表す基本項目型および指標は、その主体の主キー属性に対応させる。③ある主体型に属する主体のある属性の値を表す基本項目型および指標は、その属性に対応させる。

3.4 アクセス層

ファイル名およびレコード長はデータ集合型で定める。ファイルの入出力の区別は入出力制約で定める。ファイルと、そのファイルのレコードを定めているデータ型との対応はアクセスのデータ制約で定める。

4. プログラム生成

PSDLから COBOL等のプログラムを自動生成して、計算機で実行可能なプログラムを得る。現在作成中の実験用プログラム・ジェネレータ PSD-CG(Program Specification Description-COBOL Generator) 第 2版に基づいて、生成の基本原則、および、生成したプログラムの処理方式について述べる。

(1)生成の基本原則

PSDLから以下の原則で COBOL等のデータ域および手続きを生成する。データ域に関しては、情報層の主体型および関連型毎にテーブルを生成して、主体および関連を表す属性値を入れるための作業域とする。また、データ層のレコード対応データ型毎にレコード域を生成して入出力作業域とする。手続きに関しては、テーブル間の計算手続きを情報層の制約毎に生成する。また、そのテーブルとレコード域の間のデータ転送手続きを、データ層の情報制約毎に生成する。ファイル入出力命令はアクセス層のデータ制約毎に生成する。

計算には一度しか使わない属性値のみを持つ主体型に対しては属性値を重ね書きできるので、メモリ量効率化のため、テーブル 1 行分のみデータ域を生成する。データ域がテーブル全体分必要か、1行分でよいかは図 4-1の有向グラフを作成・解析して判定する。PSDMは高度な仕様記述法なので、実行効率のよいプログラムを生成するための一般的なアルゴリズムは得るに至っていない。第 2版は、比較的単純な入出力データ構造を持つプログラムに限定して作成中である。

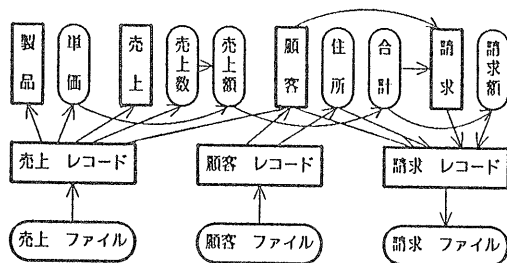


図 4-1 有向グラフ

(2)生成したプログラムの処理方式

PSDLから生成した COBOL等のプログラムは原則として、以下の順序で実行する。

- ①入力ファイルから入力レコード域へレコードを入力する。
- ②入力レコード域から各テーブルヘデータを転送する。
- ③各テーブル間で計算を行ない、計算で得た値をテーブルへ設定していく。
- ④各テーブルから出力レコード域へデータを転送する。
- ⑤出力レコード域から出力ファイルへレコードを出力する。

5. PSDMを用いたプログラム量産化技術

PSDMを用いた量産化技術について述べる。

(1)知識ベース

一つの適用領域を対象にして、業務知識原型に現れる概念を情報層に記述し、伝票形式を含めたデータ形式をデータ層に記述する。そして、その情報層記述およびデータ層記述を知識ベースへ入れる。業務知識原型は固定部と代替部に分けているので、後者に関しては代替部であることの標識を付ける。たとえば、簡略化したPOSの知識ベース概念図を示す図5-1では、業務知識原型の固定部を実線で示し、代替部を破線で示す。代替部について以下に述べる。

情報層に示した製品に関しては、たとえば、製品毎に一律に単価を決める場合もあれば、顧客対応に各製品の単価を決める場合もある。前者の場合は図5-1に示した主体型“製品”の属性“単価”を用いる。後者の場合は主体型“顧客別製品単価”の属性“単価”を用いる。データ層に示した製品売上リストに関しては、たとえば、帳票1行に製品1件分のデータを印刷する場合もある。前者の場合は左側の製品売上リストを用い、後者の場合は右側を用いる。

以上の業務知識原型へプログラム・インプリメント情報を加えて知識ベースができる。

(2)業務プログラムの合成

以下の順序で、知識ベースから業務プログラムを合成する。

①業務知識原型代替部の選択…知識ベース中の業務知識原型代替部に関して、開発対象情報システムに最も適合するものを選択する。たとえば、上記の単価に関して、製品毎に

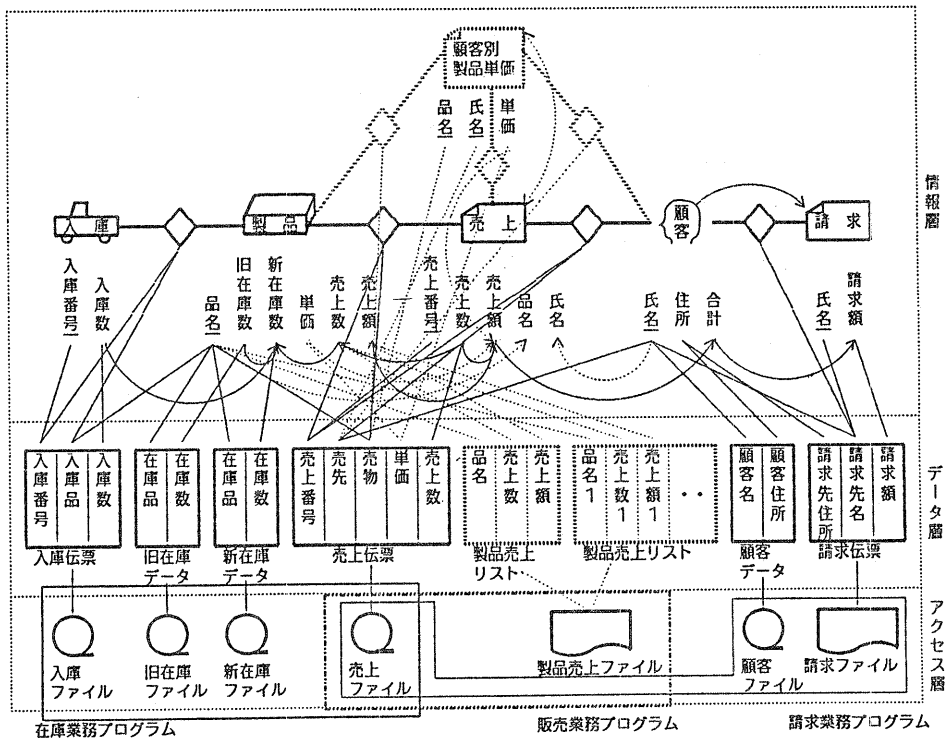


図 5-1 知識ベースの概念図

一律に決めた単価を選択する。

②情報システム個別部の反映…代替部の選択を終えた業務知識原型へ変更を加えて、開発対象情報システムへ適合させた業務知識記述を得る。プログラム・インプリメント情報へも必要に応じて変更を加える。

③業務プログラムの生成…開発対象情報システムへ適合させた業務知識記述およびプログラム・インプリメント情報から、各業務プログラムの仕様を自動抽出して、その仕様から COBOL等のプログラムを自動生成する。

(3)量産組織

知識ベースの作成・改編を行う知識ベース開発グループには、適用領域の業務知識に精通する要員、および、量産技術に精通する要員が必要である。しかし、知識ベース開発グループは適用領域毎に設ければよいので、その全要員は比較的少数でよい。したがって、知識ベース開発グループは少数精鋭で運営することになる。

一方、多数の情報システム製造グループに属する多くの業務プログラム作成者は、業務知識原型代替部の選択、および、情報システム個別部の反映により業務プログラムを合成できるので、その作業は比較的容易となる。

(4)マンマシン・インタフェース

知識ベース開発グループ要員の知識ベース作成・改編用、および、業務プログラム作成者の業務プログラム合成用に、図 5-1に示すような画面上の図形を用いたマンマシン・インタフェースが必要になる。

6. おわりに

計算指向 EARモデルに基づくプログラム仕様記述法PSDMを用いたプログラム量産化技術に関して原理を述べた。今後の課題として、順編成ファイル以外のファイルおよびデータベースを使用するプログラムも対象としたPSDMへの拡張、プログラム生成アルゴリズムの強化、知識ベースの構成法、知識ベースからプログラム仕様を自動抽出する方法、量産組織の具体的な形態、業務知識ベースの流通等に関する研究が必要である。実用化に関しては、量産化の有効性発揮およびプログラム生成アルゴリズムの実現性等の観点から、POS領域において比較的単純な入出力データ構造を持つシステムの業務プログラムの量産化から着手するのが妥当である。

参考文献

- 1)橋本正明：プログラム仕様記述のための計算指向 EARモデル、情報処理論文誌61年3月掲載予定。
- 2)橋本正明：EARモデルに基づく情報構造記述を用いたプログラム仕様記述法PSDM、情報処理学会ソフトウェア工学研究会資料40-11(1985)。
- 3)伊吹公夫、馬場康彦、劉畑朗：事例研究に基づく方法論の検討、情報処理学会情報システム研究会資料7-1(1985)。