

情報システム開発方法論のデータモデルと実現

佐藤亮
筑波大学

本研究のアプローチは、種々の情報システム分析開発方法論を定式化することにより分析し、そのエッセンスの情報を明確化するとともに、定式化されたものをデータモデルとしてモデル化して情報資源辞書システム (IRDS) に実現し、情報システムの (再) 構築でのマネジメントに有効な情報構造の分析と有効利用を考察することである。そのため分析設計方法論として、C-NAPIIと鳥田氏の事務分析技法の2つを取り上げる。データモデリング機能としてTHデータモデルを採用し、その表による表現を用いて各々の分析方法をモデル化する。その結果を市販のSQLデータベースに入れて、業務変更の例を実行しその有効性についてについて考察した。

DATAMODELS OF INFORMATION SYSTEMS METHODOLOGIES
AND THEIR IMPLEMENTATION

Ryo SATO

Institute of Socio-Economic Planning, University of Tsukuba
1-1, Tennoudai 1 chome, Tsukuba city, Ibaraki 305, Japan

This paper shows how an information resource dictionary system (IRDS in short) can be used to design organizational structure. In order to do it, two information systems methodologies are modeled into datamodels using TH data modeling facility. After a formulation of TH model is given, each of the datamodels of the two is described in details. Those models have the form of tables and are implemented in a commercial SQL DBMS. Based on an example of re-organization and decision of development of an information system, the structure of this type of IRDS and its effectiveness are considered.

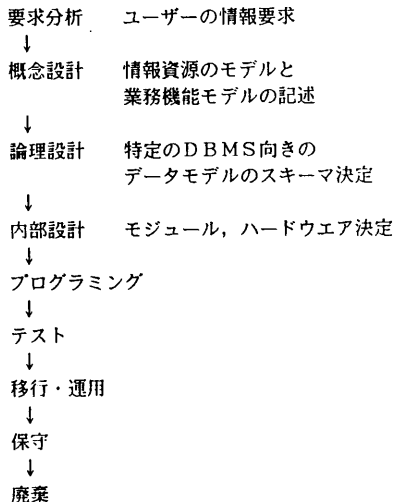
1. はじめに：組織と情報システムの関わり

組織はさまざまな情報システムを持つ。責任と権限のしくみ、トップどうしの人的ネットワーク、生産管理システム、予算統制、インフォーマル・グループなど多様なものがあるが、本小論での情報システムは、コンピュータを処理の中心とするものに限定する。

情報システムを使う側の組織にとって、情報システムは目的ではなく手段である。伝票を統廃合して業務の効率を上げるためとか、ネットワーク構築により原料市場から生産をへて消費者を結ぶ流れの中での経済的な効率性を上げて参入障壁にするなど、目的はさまざまである。

しかし、ビジネスシステムを有効にサポートする情報システムの構築法は、現在、情報システムの再構築の時期にかかっているといわれるように必要は認められているが確立された理論・手法はないようである。情報システムの技術的なソフトウェアとハードウェアは日々高度化しているが、ビジネスシステムやマネジメントコントロールと充分に整合する形での理解が完成されているわけではない。

データベースを中心とした情報システムの構築の流れは、ライフサイクル・アプローチでは下のようになっている[9]。



このような情報システムのライフサイクルの、上流過程のデータ分析を支援することが、いわゆる分析技法(手法)の目的である。ビジネスのサポートシステムとしての情報システムの設計には、業務分析と設計が特定のビジネスやマネジメントに適合する形でなされる必要がある。

この状況に切り込む行き方として、①業務とマネジメントコントロールのモデル、②情報システム開発のモデル、③2種類のモデルの整合性をはかるモデルの3種類のモデルを構築して行くというのは、ひとつの理想的な演えきのアプローチであろう。ここでモデルというのは、形式的なモデル以外に、対象の記述や論理的構造を明らかにするひとつまとまりの概念や手順・方法論も含む。

マネジメントコントロールの研究は、内部組織の問題として経済学や経営学あるいは管理会計学で種々の側面やパラダイムのもとに行われており、情報システムということばも使われるが、それはデータの共有や要約を通して業務を効果的に遂行することや種々の計画の意思決定の支援を

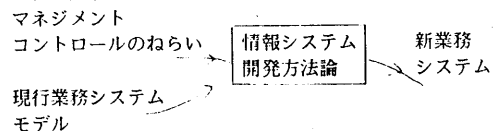
はかるためのデータベースシステムや意思決定支援システムといった、本小論の意味での情報システムやその外部設計には直接は結び付かない。組織活動はいくつかの決定からなり、それら決定問題間には相互作用がある[10]。相互作用の客観的表現としてファイルシステムがある[12]というような直接的関係を扱う研究が必要と思われる。

本研究のアプローチは、種々の情報システム開発を定式化することにより分析し、そのエッセンスの情報を明確化するとともに、定式化されたものをデータモデルとしてモデル化して情報資源辞書システム(IRDS)に実現し、情報システムの(再)構築でのマネジメントに有効な情報構造の分析と有効利用を考察することである。

今回の発表で定式化の対象として取り上げるのは、C-NAPII[3,4,5]と島田[6]の事務分析技法の2つである。データモデリング機能としてTHモデル[7]を採用し、その表による表現を用いて各々の分析方法をモデル化する。その結果を市販のSQLデータベースに入れて、[1]の例を実行する。

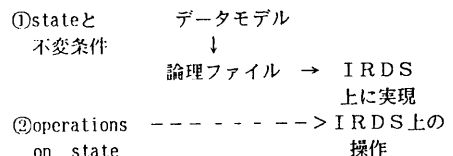
2. 情報システム開発法のモデルについて

我々は情報システム開発を変換であるとみる。変換の入力はマネジメントコントロールのねらいと現行業務システムであり、出力は情報システムが入り込んだ新業務システムである。定式化アプローチにより、情報システム開発の外部設計(組織活動と情報システムとの関わりを定義すること)に関する知見を得ることを目的としている。



定式化は状態表現を構成する。状態表現はstate(状態)とその上のoperationから成る。開発の方法論が客観化された手順を書いた文書として存在する理由は、ひとつには開発という組織的な協同活動を一貫性のあるものにするために、活動の一定段階ごとにおいて開発に関連する情報を諸ドキュメントとして規定し、それ以降の段階は規定された情報を共通の与件として各自の活動を行うというコントロールを働かせるというねらいがある。stateを構成する情報とそれに対する任意のoperationが満たすべき不変条件が、情報システム開発で用いられる一連のドキュメントが表現している情報のモデルである。IRDS上に実現するにはデータモデルを定義し、ファイルとして持つ必要があるが、stateと不変条件がデータモデルに対応する。

情報システム開発方法論の定式化とIRDS上の実現の関係は下の通り。



要求分析技法であるC-NAPIIと島田氏の事務分析技法の場合のstateと不変条件の定式化とデータモデルは後述する。現在のところ情報システム開発における本質的な情

報を語るほどのおおきの知見はそろっていないが、このような具体的な定式化作業によって、明らかになってくると考える。

3. 業務システム分析へのIRDSの利用

情報システム開発を効果的に進めるためのIRDSの利用は、たとえば、[1],[2]が述べている。[1]が一番明確に記述している。企業モデル（処理と関連情報）を一方でIRDSのIRDレベルのデータとしていれておき、やはりIRDレベルに保持している計画している業務システムのモデルと突き合わせて、情報システムと企業活動の整合性を人間のマネジメントが検討するというものである。IRDSのcross-reference能力を重点的な利用を試みるという。

これを実行するには、IRDDレベルに、情報システム開発方法論がもつ概念枠組みをモデル化し企業モデルや業務モデルを保持する。その結果、IRDレベルにどのような種類の表を持つかは方法論ごとに異なる。我々の定式化の結果からすると、方法論ごとにやや異なった情報構造（と実行手順）を持つように見える。ということは、ある種の情報システムに向く方法論と向かないものがある可能性がある。したがって、現段階でいえるのは、実際にIRDSで動かし、知見を蓄える必要があるということである。

また、企業モデルの構築がやはり自明ではない。分析が、したがって分析結果が、ビジネスごとに、情報システムごとに異なるだろうと予想する文献もある[6]。

4. 島田事務分析技法[6]

4.1 概要

コンピュータ導入のための事務詳細分析手法

事務の業務を入力出力システムとしてみる。すなわち、事務作業過程の各ステップを「インプット→変換→アウトプット」として見るというものである。この分析の結果として得られるドキュメントは

- 事務分析フローチャート
- 作業明細記述書
- 帳表詳細分析表
- 総合システム分析図

の4種類である。帳表の動きではなく帳表の中のデータに注目する。

4.2 事務分析フローチャート

業務の事務処理作業の過程を流れ図でしめすもの。流れ図に記述されるものは、

- 1) 業務内容を大別する。たとえば、引合、納期照合、注文書発行など。
- 2) 大別した各区分についてフローチャートを作成する。まず、より細かい作業区分を設定してステップに分ける。
- 3) 各ステップの処理に必要な帳表類を、そのステップへの入力として取り出す。
- 4) 各ステップが作成したデータ、アウトプットデータを明確にする。

4.3 作業明細記述書

事務分析フローチャートにあらわれた各ステップについて、作業内容を簡略に言葉で書いたものである。フローチャート作成と同時に並行して作成する。

4.4 帳表詳細分析表

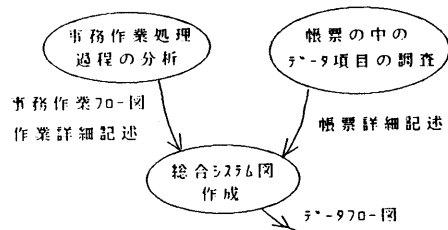
分析者は、あらかじめ仕事の流れにそった帳表をすべて集めてファイルしておく。

帳表詳細分析表は、事務処理ステップに対応して、インプット帳表のどのデータ項目が、アウトプット帳表のどのデータ項目に関係を持つかを記述したものである。単に転記されるものと、いくつかの入力項目からの演算により定まる出力項目とがある。

4.5 総合システム図

ビジネス活動全体をある粗さで一貫して記述する図である。データフローダイアグラムのデータフローとして帳表を描き、データストアとしてファイルを描いたものである。基本的な作成手順は；

- 1) 組織、業務内容を検討して業務処理区分のまとめをする。粗すぎると実体の特殊性がわからなくなり、こまかすぎると全体的な観点が失われる。
- 2) 各部分業務を結合する情報は、帳表データである。こうしてできた総合システム図は、新システムを発想する際のたたきだいに使える。



島田氏の事務分析技法

5. 情報システムの要求分析技法 C-NAP II

5.1 概要

C-NAP IIは、①ニーズ分析、②業務分析、③データ分析という一連の段階からなる。(以下、C-NAP IIをC-NAPと略す。)C-NAP IIの手順は簡単には以下の通りである[4]。

準備作業：スタッフチームの編成、実施要領の検討など。
情報収集：エンドユーザーから問題点やニーズについての情報を収集し、整理。

業務分析：仕事の流れを明確にし、さらに業務で注目する対象（管理対象）を明確にする。

問題分析：問題相互の因果関係を分析し、真の問題点を見つけて出す。

目的分析：達成すべき業務目的を明確にする。

手段検討：目的達成のための手段をトップダウンで検討し、重要手段をシステム要求として設定する。

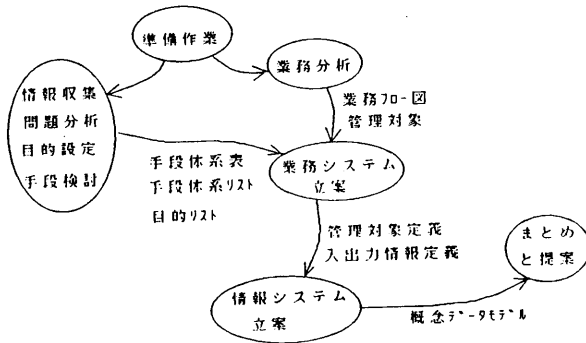
業務システム立案：設定したシステム要求を現行業務の分析結果に帰着させ、情報システムも含めた新たな業務システムの立案を行う。

情報システム立案：新業務システムの入出力情報の流れ（データフロー）と業務で扱う対象間の関連（データモデル）を明確にすることで、情報システムの要求仕様を作成する。

まとめと提案：情報システムの開発計画書をまとめ、提

案する。

以上の手順をデータフローで表現すると、図のようになる。



2. 業務フロー

業務フロー図と対象分類図は、業務システム分析のためのモデルである。システムの利用者となるひと自らが、主体的に行う業務分析・業務立案の活動に用いる。したがって、情報システムの厳密なデータモデル図として使うのではない。あとで、システムアナリストが情報要求を抽出・定義するのに使うことができる。

2. 1 業務フロー図

業務フロー図は拡張DFDである。拡張というのは、定時に発生する処理を取り入れているからである。

構成要素；

- 事象、タスク、ファイル（通常のDFDの構成要素）
- 仕事の流れ、（一連のタスクのつながり）
- ものの流れ、（データフローだけでなく、システム内で加工や移動される人、物の流れも記述する）
- キャラクター（ものとの情報以外に、入出力変換を支援・促進するもの。例は、設備、機器、倉庫、技術、標準、在庫商品）

業務フロー図のかきかた；

- 1) 現行業務の中から外部環境に起因する事象を起点とする仕事の流れ（特定の業務目的）を識別し、その一連のタスク連鎖を縦方向に書き下す。
- 2) 次いで、左右に、入出力となっている物流・人流・情報流を識別する。
- 3) 最後に入出力変換である仕事の流れが必要とする「もの」（キャラクター）、必要とする情報、その情報源を左右に抽出する。

以上の現行システムから、ヨリ効果的な新システムを求めている分析・立案をおこなう。いくつかの視点がある（略）。業務の目的とタスクの目的を明示化して、新システムを、情報要求をつくるたきたき台に使う。

5. 2. 2 対象分類図による業務の分析

業務フローの図から始めて、特別なER図である対象関連図をつくる。エンティティを実体として形をもつ「もの」に限る。「受注」などの活動は除く。リレーションとしては、取引・事象などの「できごと」に限り、部品構成

などの組み合わせの関係を除くことで、ERの概念を簡単にしている。

対象分類図の作成手順

- 1) 「もの」の抽出
- 2) 「できごと」の抽出
- 3) ツリー構造図の作成
- 4) 詳細化
- 5) 構造化表記への整理
- 6) 対象関連図（ER図）の作成

以上で作成した対象分類図や関連図をもとに、分析対象とする業務の構造に着目した分析（ニーズ抽出）と新システムの立案（要求定義）を行う。その際のいくつかの観点が（略）。

対象分類図に基づき、構築される情報システムが管理の対象にするものを識別し、必要となる項目（キーとなる項目、区分コード、主要データ項目）を情報要求として定義する。

5. 2. 3 イベント・シーケンス分析という手法

概念ファイルに属性項目をアサインするのに、イベントシーケンス図を用いる。

概念ファイルにたいして、その1インスタンスレコード(tuple)が影響を受けるイベントを時刻順に並べ、その状態遷移を明らかにする。その上で、「そのファイルがどの時点で何を知ることができるか」という観点で属性項目をファイルにアサインする。このイベントシーケンス分析結果から、繰り返し項目群の分類と遷移従属関係の排除を行うことで、概念データモデルが得られる。

6 THデータモデル[7]

6. 1 実体、集合実体、実現関数

$ES = \{x \mid x \text{は実体(entity)}\}$, entityのset.

$SS = \{x \mid x \text{は集合実体}\}$, set entityのset.

SSはESの部分集合と定める。

$S: SS \rightarrow P(ES)$, 実現値対応関数(実現関数とよぶ)。

ここで $P(ES)$ はESのべき集合。

Sは、SS上では必ず値をもつ。Sが属する空間を

$[SS \rightarrow P(ES)]$ とかく。

$x \in S$ に対して、 $e \in S(x)$ をxの実現値のという。

6. 2 データ操作命令(変換関数)

追加(add)

$add: [SS \rightarrow P(ES)] \times SS \times ES \rightarrow [SS \rightarrow P(ES)]$

任意の $S \in [SS \rightarrow P(ES)]$, $x \in SS$, $y \in SS$, $e \in ES$ について、

$$add(S, y, e)(x) = \begin{cases} S(x) \cup \{e\}, & (x=y \text{のとき}) \\ S(y), & (x \neq y \text{のとき}). \end{cases}$$

削除(del)

$del: [SS \rightarrow P(ES)] \times SS \times ES \rightarrow [SS \rightarrow P(ES)]$

任意の $S \in [SS \rightarrow P(ES)]$, $x \in SS$, $y \in SS$, $e \in ES$ について、

$$del(S, y, e)(x) = \begin{cases} S(x) \setminus \{e\}, & (x=y \text{のとき}) \end{cases}$$

$$S(y), \quad (x \neq y \text{ のとき}).$$

実体型

$x \in SS$ が実体型であるとは、勝手な add と del に対して、 $S(x)$ が不変なこと；すなわち、任意の $e \in ES$ について、 $\text{add}(S, x, e)(x) = S(x)$ かつ $\text{del}(S, x, e)(x) = S(x)$ なること。

実体型の集合を ETS とかく (entity-type set)。

6.3 ファイル

ETS の部分集合で KTS なるキー実体型の集合とよばれるものとする。このとき、ファイル型関数 $FT: KTS \rightarrow ETS$ が存在するものとする。 x をキー実体型とする。 $S(FT(x))$ の要素を $f(x)$ とかく、つまり $f(x) \in S(FT(x))$ 。 $f(x)$ の f は file の先頭の文字の f である。通常は $FT(KTS)$ 上で S は 1 対 1 対応である。

FT は 1 対 1 対応であるとする。つまり、ファイル型とキー実体型は同一視される。

$FT(x)$ を x のファイル型、 $f(x)$ を x のファイルという。キー実体型を管理対象型ということもある。

$S(f(x))$ の要素をレコード (厳密にはレコード実現値) と呼ぶ。すなわち、レコードは $f(x)$ の実現値である。

ファイル型の集合 FTS とファイルの集合 FS を次のように定義する。

$$\begin{aligned} FTS &= FT(KTS) \\ FS &= \{f(x) \mid x \in KTS\} \\ &= \{S(FT(x)) \mid x \in KTS\} \end{aligned}$$

6.4 属性

属性の集合、準属性の集合

属性の集合を AS であらわす。準属性の集合を QS であらわす。 AS と QS は互いに素である、すなわち $AS \cap QS = \emptyset$ 。 $CS = AS \cup QS$ と定め、座標の集合と呼ぶ。属性も準属性も、集合実体でない (SS に属さない) 実体である。

属性名関数

$A: FTS \rightarrow P(AS)$ 、ただし $P(AS)$ は AS のべき集合。

ドメイン

実体型であって複合定義域か、または、原子定義域であるものである。ドメインの集合を DS 、複合定義域を CDS 、原子定義域を ADS とあらわす。 CDS と ADS は互いに素である。 $DS = CDS \cup ADS$ が成り立つ。

原子定義域値域

$S(ADS)$ を原子定義域値域という。

座標の要素と対応するドメインを与える関数 dom がある：

$$\text{dom}: CS \rightarrow DS$$

$\text{dom}(a)$ を a のドメイン (定義域) という。

複合定義域関数

$$Q: CDS \rightarrow P(QS) \setminus \emptyset$$

属性 a が複合属性であるとは、 $\text{dom}(a) \in CDS$ であること、つまり、 a がさらにいくつかの準属性に分解されることである、いいかえると、 $Q(\text{dom}(a))$ はいくつかの準属性から成る。同様に、準属性 a の定義域が CDS の要素であるとき、 a を複合準属性という。

以下の条件の成立を仮定する：

- (1) キー実体型の集合 KTS はドメイン集合 DS の部分集合であるときだめる。
- (2) $x \in CDS$ に対して、 $(Q \cdot \text{dom})$ を可能な要素に対して 1 回以上有限回の適用によって、結果はすべて ADS に入ること。

多次元実体の (ここで次元とは構成する属性・準属性数のこと) 集合を MDS とかく。

$$MDS = FTS \cup CDS$$

任意に KTS から x をとる。(つまり x が $FT(x)$ と $f(x)$ を持つ) $FT(x)$ の属性を a_1, \dots, a_n とする、 $A(FT(x)) = \{a_1, \dots, a_n\}$ 。ファイル $f(x)$ の実現値 $y \in S(f(x))$ に対して

$$\begin{aligned} V(y, a_i) &\in S(\text{dom}(a_i)), \\ &(i = 1, \dots, n) \end{aligned}$$

が定義されているものとする。 $V(y, a_i)$ を、 y の属性 a_i に関する属性値と呼ぶ。つまり $V(-, a_i)$ は、実現値に対する属性 a_i 上の射影である。また、このとき、 a_1, \dots, a_n をファイル型 $FT(x)$ の属性とも、ファイル $f(x)$ の属性とも呼ぶ。

属性名関数 $A: FTS \rightarrow P(AS)$ を拡張して $A: FTS \cup FS \rightarrow P(AS)$ とする、ただし、ある $y \in KTS$ があって、 $x = FT(y)$ または $x \in S(FT(y))$ のとき、 $A(x) = A(FT(x))$ と定める。

$A(FT(x)) = \{a_1, \dots, a_n\}$ であるとき、任意の $y_1, y_2 \in S(f(x))$ に対し、 $y_1 = y_2$ であることを、各 $a_i (i = 1, \dots, n)$ に対して $V(y_1, a_i) = V(y_2, a_i)$ であることと定める。

若干の注意；

$x \in CDS$ 、 $Q(x) = \{a_1, \dots, a_n\}$ とするとき、 $S(x) (S(\text{dom}(a_1)) \times \dots \times S(\text{dom}(a_n)))$ と考えると当然、任意の $y \in S(x)$ について

$$\begin{aligned} V(y, a_i) &\in S(\text{dom}(a_i)), \\ &(i = 1, \dots, n). \end{aligned}$$

同様に

$x \in KTS$ 、 $S(FT(x)) = \{a_1, \dots, a_n\}$ 、 $f(x) \in S(FT(x))$ とするとき、 $S(f(x)) (S(\text{dom}(a_1)) \times \dots \times S(\text{dom}(a_n)))$ とすると当然、任意の $y \in S(f(x))$ について

$$\begin{aligned} V(y, a_i) &\in S(\text{dom}(a_i)), \\ &(i = 1, \dots, n). \end{aligned}$$

と考えたくなるが、 $S(x)$ の要素の順番を指定する必然性がないので、ordered pair の集合とは見なさない。

レコード (タプル) の集合とは

$LSE = S(S(FTS)) \cup S(CDS)$
 である。これは実体であって、列実体の集合である。

6.5 主キー, Rキー, 継承, 汎化, 専化

$key: FTS \rightarrow AS$ は、ファイル型 $y = FT(x)$ に対してキー(または主キー)を与える関数である。すなわちすべてのファイル型はキーを持つ。ここで $key(y)$ がファイル型 y のキーであるとは;

任意の $S, f(x) \in S(y), z1, z2 \in S(f(x))$ に対して $[V(z1, key(y)) = V(z2, key(y))]$ ならば $z1 = z2$]。

$y = FT(x)$ のとき、 $dom(key(y)) = x$ である。
 また、 $dom(key(FT(x))) = x$ iff
 $key(FT(x))$ がファイル型 $FT(x)$ のキー。

以下の条件の成立をさらに仮定する:

(3) $ETS = DS \cup FTS$ (直和) = $MDS \cup ADS$ (直和)

(4) $ES = SS \cup FS \cup LES \cup CS \cup S$ (直和)

参照キー (参照整合性)

$x1, x2 \in KTS$ をとる。

$a \in A(FT(x2))$ が、 $FT(x2)$ の $FT(x1)$ への参照キー(Rキー)であるとは、

- ① $dom(key(FT(x1))) = dom(a) = x2$, かつ、
- ② $f(x1) = FT(x1), f(x2) = FT(x2)$ と任意の S に対し、
 $S(f(x2)).a \in (S(f(x2)).key(FT(x1)))$ 。

このとき $FT(x2) \leftarrow FT(x1)$ とかく。

参照キーは、データベースのファイル間の整合性条件としての参照整合性を表現するのに使われる。

参照関係

$REFER (AS \times AS)$

$(a, b) \in REFER$ とはある $x1, x2 \in KTS$ があって次の2条件を満たすこと:

- ① $a \in A(FT(x2)), b = key(FT(x1))$,
- ② a が $FT(x2)$ の $FT(x1)$ へのRキーである。

継承(inheritance)整合性

$x1, x2 \in KTS$ をとる。

$FT(x1)$ は $FT(x2)$ の部分ファイル型、あるいは $FT(x2)$ は $FT(x1)$ の汎ファイル型、あるいは、
 $FT(x1)$ は $FT(x2)$ の属性を継承するとは、

- ① 任意の S に対して
 $V(S(f(x1)), key(FT(x1))) \subseteq V(S(f(x2)), key(FT(x2)))$;
 ファイル $f(x1)$ の任意の組 y に対して、 y のキー値と等しいキー値を持つ $f(x2)$ の組 z がある、かつ、
- ② $A(FT(x1)) \setminus \{key(FT(x1))\} \subseteq A(FT(x2)) \setminus \{key(FT(x2))\}$;

$FT(x1)$ は $FT(x2)$ のキー以外の属性を含む、つまり、
 $FT(x1)$ はキー以外の属性を継承する、かつ、

- ③ 任意の $a \in A(FT(x2)) \setminus \{key(FT(x2))\}$,
 $y1 \in S(f(x1)), y2 \in S(f(x2))$ に対して、
 $[V(y1, key(FT(x1))) = V(y2, key(FT(x2)))]$ ならば
 $V(y1, a) = V(y2, a)$;
 $FT(x2)$ のキー値以外の継承属性値は一致する。

上のとき、 $FT(x1) \leftarrow FT(x2)$ とも、
 $FT(x1) \text{ inherits } FT(x2)$ とかく。

継承関係

$INHRTS (FS \times FS)$ を
 $(z1, z2) \in INHRTS$ iff $z1 \text{ inherits } z2$ で定める。
 S が $FTS \rightarrow FS$ なる1対1対応の時は、
 $INHRTS$ を FTS 上の2項関係ともみなせる。

$x1, x2$ を DS からとる。 $x1$ が $x2$ の部分型(あるいは $x2$ は $x1$ の汎型)であるとは、
 $S(x1) \subseteq S(x2)$ 。

上のとき、 $x1 \text{ isa } x2$ とか、 $x1 \leftarrow x2$ とかく。

$x1, x2$ が FTS の要素の時は、 $x1 \leftarrow x2$ を
 $FT(x1) \leftarrow FT(x2)$ で定める。

ISA (DS × DS)

$(x1, x2) \in ISAS$ を $x1 \text{ isa } x2$ で定める。

7. 分析技法のデータモデル

各分析技法の定式化は紙幅の関係上、省略する。[13]を参照頂きたい。

7.1 鳥田方法論のデータモデル

7.1.1 鳥田方法論の表の概略

(1) 作業担当表

[ステップ番号] (作業名, 担当名, 業務名)

(2) 作業入力表

[作業矢印番号] (入力帳票, ステップ番号)

(3) 作業出力表

[作業矢印番号] (出力帳票, ステップ番号)

(4) 入力帳票詳細

[帳票項目] (桁数, ステップ番号)

帳票項目: (データ項目, 入力帳票)

(5) 出力帳票詳細

[帳票項目] (桁数, ステップ番号)

帳票項目: (データ項目, 出力帳票)

(6) 作業詳細説明

[ステップ番号] (作業名, 処理手順文)

(7) 帳票

[帳票番号] (帳票名)

(8) 業務区分

[業務番号] (業務名)

(9) 総合システム表

(a) 総合システム入力

[システム矢印] (帳表, 矢印元環境, 矢印先業務)

(b) 総合システム内部

[システム矢印] (帳表, 矢印元業務, 矢印先業務)

(c)総合システム出力
 [システム矢印] (帳票, 矢印元業務, 矢印先環境)
 (10)環境表
 [環境番号] (環境名)

7. 2 島田方法論のメタデータベース
 THデータモデル機能を使い, 方法論に使われるドキュメントを表として, 実体型表, 表表, 列表, 参照表, 副表表の5つによって示す.

7. 2. 1 実体型表
 ドメイン (DS) を表現する. キー実体型KTSはDSの部分集合である. キー実体型であるドメインを, 実体型種別の値にCを取ることで示す.

実体型	実体型識別子	実体型名	実体型種別	コメント
	1	帳票名		
	2	業務名		
	3	文章		
	4	帳票データ	C	複合属性 (データ項目, 帳票名)
	5	作業名		
	6	担当名		
	7	段階No	C	
	8	矢印No	C	
	9	システム矢印No	C	
	10	帳票No	C	
	11	業務No	C	
	12	データ項目		4を構成するAD
	13	環境#	C	
	14	環境名		

7. 2. 2 表表
 ファイル型 (FTS) を表現する表.
 表

表識別子	表名
511	作業担当
512	作業入力帳票
513	作業出力帳票
514	入力帳票詳細
515	出力帳票詳細
516	帳票一覧
517	業務一覧
519	作業詳細
520	総合システム入力
521	総合システム内部
522	総合システム出力
523	環境

注: 520,521,522の表を得ることが分析の目標で, それぞれ (511,512,513,516,517,523), (511,512,513,516,517), (511,512,513,516,517,523)から得られる.

7. 2. 3 列表
 属性 (AS) と対応するドメイン (dom (AS)) を表現する表.

列	列名	表識別子	定義域識別子	キー性質
列識別子	列名	表識別子	定義域識別子	キー性質
101	ステップ#	511	7	Y

102	作業名	511	5	
103	担当	511	6	
104	業務	511	2	
105	矢印#	512	8	Y
106	入力帳票	512	10	
107	ステップ#	512	7	
108	帳票項目	514	4	Y
109	桁数	514		
110	ステップ#	514	7	
111	帳票#	516	10	Y
112	帳票名	516	1	
113	業務#	517	11	Y
114	業務名	517	2	
118	ステップ#	519	7	Y
119	作業名	519	5	
120	処理手順文書	519	3	
121	矢印#	513	8	Y
122	出力帳票	513	10	
123	ステップ#	513	7	
124	帳票項目	515	4	Y
125	桁数	515		
126	ステップ#	515	7	
127	システム矢印	520		Y
128	帳票	520	10	
129	矢印元環境	520	13	
130	矢印先業務	520	11	
131	システム矢印	521	9	Y
132	帳票	521	10	
133	矢印元業務	521	11	
134	矢印先業務	521	11	
135	システム矢印	522	9	Y
136	帳票	522	10	
137	矢印元業務	522	11	
138	矢印先環境	522	13	
139	環境#	523	13	Y
140	環境名	523	14	

7. 2. 4 参照表(バックマン図では, 参照元←参照先と描かれている)

関係REFERを表現する表.
 参照

参照元列識別子	参照先列識別子	コメント
104	113	104←113
106	111	帳票No
107	101	ステップNo
108	111	複合属性の一部
110	101	
124	111	複合属性の一部
126	101	
122	111	
123	101	
126	101	
128	111	
132	111	
136	111	
130	113	←業務No
133	113	
134	113	

137	113	
129	139	←環境No
138	139	

/*環境から物が物流へ*/

物流処理環境フロー
 [矢印番号] (物流処理, 環境, 物, 業務フロー)
 /*物流から物が環境へ*/

7.2.5 副表表
 関係INHR TSを表現する表.
 副表

副表識別子 正表識別子 コメント
 (データなし)

環境業務処理フロー
 [矢印番号] (環境, 業務処理, 情報, 業務フロー)
 /*環境から情報が業務へ*/

業務処理環境フロー
 [矢印番号] (業務処理, 環境, 情報, 業務フロー)
 物流処理業務処理フロー

7.3 C-NAPIIのデータモデル

7.3.1 C-NAPIIの表
 NA

[矢印番号] (物流処理, 業務処理, 情報, 業務フロー)
 業務処理物流処理フロー
 [矢印番号] (業務処理, 物流処理, 情報, 業務フロー)

業務処理情報フロー
 [矢印番号] (業務処理, 情報, 業務フロー)
 /*業務で情報が作られる*/

1)目的リスト

テーマ表

[テーマ番号] (テーマ名)

サブテーマ表

[サブテーマ番号] (サブテーマ名, テーマ名)

目的リスト

[目的番号] (目的, 順位, 効果, 達成時期, サブ
 テーマ, テーマ)

達成基準

[尺度番号] (測定尺度, 目的, 現状値, 目標値)

2)手段体系

手段表

[手段番号] (手段名)

手段体系表

[情報番号] (必要情報, 手段, 目的, 方法)

手段方法リスト

[方法番号] (方法, 手段, 優先順位)
 /*省略部分あり*/

情報業務処理フロー
 [矢印番号] (情報, 業務処理, 業務フロー)
 /*情報が業務で使われる*/

管理ポイント

[管理ポイント番号] (管理ポイント名, 業務フロー)
 /*各業務フローにおける管理したい物の名前, 物表の専化*/

3)管理対象カードと管理対象分類

管理対象クラス例

[クラス例番号] (管理対象クラス名)

/*典型的な管理対象の例示の表*/

管理対象

[管理対象番号] (管理対象名)

管理対象属性

[属性番号] (属性名, 管理対象)

管理対象分類観点

[分類観点番号] (分類観点名)

下位管理対象

[管理対象番号] (管理対象名, 上位管理対象, 分
 類観点) /*管理対象名表の専化*/

SA

1)業務概観関連

サブ組織

[サブ組織番号] (サブ組織名, 区分)

業務概観取引組織

[矢印番号] (矢印元, 矢印先)

業務概観物流

[矢印番号] (矢印元, 矢印先, 物)

業務概観情報流

[矢印番号] (矢印元, 矢印先, 情報)

2)業務フロー

業務フロー一覧

[業務フロー番号] (業務フロー名, バージョン番号)

環境

[環境番号] (環境名)

物流処理

[物流処理番号] (物流処理名)

業務処理

[業務処理番号] (業務処理名)

情報

[情報番号] (情報名)

物

[物番号] (物名)

環境物流処理フロー

[矢印番号] (環境, 物流処理, 物, 業務フロー)

4)新規業務フロー

新規業務フロー

[業務フロー番号] /*業務フロー一覧表の汎化*/

/*入力情報定義, 出力情報定義, 管理対象定義書は省略
 する. 新規かどうかは, 3)の管理対象関連の表にさらに業
 務フロー番号という属性を付け加えることで出来ると思わ
 れる. また, 入出力定義書は, 島田事務分析技法の入出力
 帳票詳細に酷似している*/

DA

/*THモデルで, ERAモデルのデータモデルを作る必
 要があるが, いまは興味がない. CNAPIIの目標は, 業
 務システムのモデルとそこにある情報システムのデータ
 のデータモデルの2つを記述することである. 業務シス
 テムはこれまでのNA, SAの記述枠組みで記述される. デ
 ータのデータモデルは, THモデルを使いたいのだが, その
 モデル化の記述枠組みはすでに存在する. 次の7.4節の
 5つの表である.

とはいえ, 応用データベースのTHデータモデルにおけ
 る属性, Rキー, 汎化を見いだすのは自明なことではない
 ため, CNAPIIのイベントシーケンス図, あるいはその
 考え方をを用いることが検討されるべきである. */

7. 4 C-NAPIIのメタデータベース

実体型表, 表表, 列表, 参照表, 副表表の5つを示す.

7. 4. 1 実体型表

実体型

実体型識別子	実体型名	実体型種別
1	テーマ#	C
2	テーマ名	
3	目的#	C
4	目的文	
5	順位	
6	効果文	
7	時期	
8	尺度#	C
9	測定法	
10	水準	
11	必要情報#	C
12	情報	
13	方法#	C
14	方法文	
15	手段#	C
16	手段文	
17		
20	矢印#	C
21	組織#	C
22	組織	
23	区分	(客, 営業書, 本社)
24	環境#	C
25	環境	
26	物流処理#	C
27	物流処理	
28	業務処理#	C
29	業務処理	
30	情報#	C
31	物#	C
32	物	
33	管理対象#	C
34	属性#	C
35	管理対象	
36	属性	
37	業務フロー#	C
38	業務フロー	
39	バージョン#	
40	分類観点#	
41	分類観点	

7. 4. 2 表表

表識別子	表名	コメント
501	テーマ	
502	サブテーマ	
503	目的リスト	
504	達成基準	
505	手段一覧	
506	手段体系表	
507	手段方法リスト	
508	サブ組織	
509	業務概観	
510	業務フロー一覧	

511	環境	
512	物流処理	
513	業務処理	
514	情報	
515	物	
516	環境物流処理フロー	環境から物流処理への流れ
517	物流処理環境フロー	物流処理から環境への流れ
518	環境業務処理フロー	環境から業務処理へ
519	業務処理環境フロー	
520	物流処理業務処理フロー	
521	業務処理物流処理フロー	
522	業務処理情報フロー	業務処理からの情報の出力
523	情報業務処理フロー	情報が業務処理へ入力
525	管理対象クラス例	
526	管理対象	
527	管理対象属性	
528	管理対象分類観点	
529	下位管理対象	526の専化
530	新規業務フロー	
531	業務概観物流	509のdisjoint専化
532	業務概観情報流	509のdisjoint専化
533	管理ポイント物	515の専化
534	管理ポイント情報	516の専化

7. 4. 3 列表

列識別子	列名	表識別子	定義域識別子	参照	キー性質
101	テーマ番号	501	1		Y
102	テーマ名	501	2		
103	サブテーマ番号	502	1		Y
104	サブテーマ名	502	2		
105	テーマ	502	1	101/*105←101*/	
106	目的番号	503	3		Y
107	目的	503	4		
108	順位	503	5		
110	効果	503	6		
111	達成時期	503	7		
112	サブテーマ	503	1	103	
113	テーマ	503	1	101	
114	尺度番号	504	8		Y
115	測定尺度	504	9		
116	目的	504	3	106	
117	現状値	504	10		
118	目標値	504	10		
119	手段番号	505	15		Y
120	手段	505	16		
121	情報番号	506	11		Y
122	必要情報	506	12		
123	手段	506	15	119	
124	目的	506	3	106	
125	方法	506	13	126	
126	方法番号	507	13		Y
127	方法	507	14		
128	手段	507	15	119	
129	優先順位	507	5		
130					
131	サブ組織番号	508	21		Y
132	サブ組織名	508	22		
133	区分	508	23		
134	矢印番号	509	20		Y
135	矢印元	509	21	131	
136	矢印先	509	21	131	
138	業務フロー番号	510	37		Y
139	業務フロー名	510	38		
140	バージョン番号	510	39		

(途中、一部省略)

185	矢印番号	523	20		Y
186	情報	523	24	141	
187	業務処理	523	28	145	
188	業務フロー	523	37	138	
191	クラス例番号	525	33		Y
192	管理対象クラス名	525	35		
193	管理対象番号	526	33		Y
194	管理対象	526	35		
195	属性番号	527	34		Y
196	属性名	527	36		
197	管理対象	527	33	193	
198	分類観点番号	528	40		Y
199	分類観点	528	41		
201	管理対象番号	529	33		Y
202	管理対象名	529	35		
203	上位管理対象	529	33	193	
204	分類観点	529	40	198	
205	業務フロー番号	530	37		Y
206	矢印番号	531	20		Y
207	矢印元	531	21	131	
208	矢印先	531	21	131	
209	物	531	31	149	
210	矢印番号	532	20		Y
211	矢印元	532	21	131	
212	矢印先	532	21	131	
213	情報	532	24	141	
214	管理* イト番号	533	31		Y
215	管理* イト物	533	32		
218	業務フロー	533	37	138	
216	管理* イト番号	534	30		Y
217	管理* イト情報	534	12		
219	業務フロー	534	37	138	

7. 4. 4 参照表(バックマン図では、参照元←参照先と描かれている)

参照

参照元識別子 参照先識別子 コメント

(* 列表に参照として記した*)

7. 4. 5 副表表(副表←正表とかかれる)

副表

副表識別子	正表識別子	コメント
529	526	下位管理対象←管理対象
531	509	業務概観物流← 業務概観取引組織disjoint
532	509	業務概観情報流← 業務概観取引組織disjoint
534	515	管理* イト情報←情報
533	515	管理* イト物←物

8. おわりに

2つの情報システム分析方法論を、THデータモデリング機能を使いデータモデル化し、市販のSQLデータベースで実現し、マネジメントシステムと整合した情報システムの構築のサポートについて実験した。

7節の方法論の表はIRDDレベルの表となる。それらの表を使って文献[1]の戦略的システム計画の例を記述してある範囲で実現してみた結果、[1]と同程度に、企業目的に

らして組織構造を変更すること、すなわち業務や業務が必要とするデータを実現するための情報システムを提案するマネジメント活動を支援しうることがわかった：

①IRDSとしての実現の方法は、IRDDレベルで方法論をモデル化し、IRDDレベルにその方法論に基づいた特定の業務システムの分析結果がモデル化されて存在することになる。今回扱った方法論の1つ(C-NAP II)の場合には、最終アウトプットに概念ファイルの設計を含む。それらは特定目的の応用データベースのファイルになる。

②一方、応用データベースの業務システムにおける位置づけに関連するデータも方法論による業務のモデル化の結果として存在する。すなわち、応用データベースのファイルシステムのデータモデルだけがIRDDにあるのではない。

③また、業務システムが扱うデータと業務システムの(再)構築に使うためのデータは一般には異なると考えられるが、レベルの異なるそれらの情報を整合させるかどうかはIRDSではなく、方法論の問題である。

業務システム及びマネジメントと整合する情報システムの分析・設計をねらった方法論は[11,15]で考察されているが、分析のアウトプットとして概念ファイルを明確に打ち出していない。これはデータベースシステムが業務システムのサポートのため、また活動の記録とモニタのために広く使われていることからみて好ましくないが、明示的にマネジメントの観点を打ち出しているのがよい。その場合でも、やはりIRDDレベルの一群の表としてデータモデルが作られることになろう。

今後[8,14]を用いて、バージョンやワーキングセットを利用した場合の有効性に付いて、さらに多くの方法論と例題を扱っていく予定である。

参考文献

- [1]NBS500-152: Guide to IRDS Applications: General Concepts and Strategic Systems Planning, 1988. [2]ISO/TC97/SC21/WG3: INFORMATION RESOURCE DICTIONARY SYSTEM, TUTORIAL-Revision 4, 1986. [3]富士通(株):C-NAP I I/NA, C-NAP II/SA, C-NAP II/DA, 1989. [4]森, 永田:システム要求分析技法C-NAP II, 情報システム, 19-6, 1988. [5]橋本, 永田:データ中心アプローチに基づく上流工程支援, 情報システム, 21-2, 1988. [6]島田清一:システムエンジニアのための業務分析の手法, 日刊工業新聞社, 1988. [7]穂鷹良介:データシステムとデータモデル, オーム社, 1989. [8]岩崎一正:SQLによるIRDSの実現, 本予稿集. [9]国友義久:効果的プログラム開発技法(第2版), 近代科学社, 1983. [10]Mesarovic, M.D., Takahara, Y.: Abstract Systems Theory, (Lecture Notes in Control and Information Sciences 116), Academic Press, 1989. [11]Wilson, Brian: Systems: Concepts, Methodologies, and Applications, John Wiley, 1984. [12]高原康彦: Data Processing Systemの設計に対するシステムからの検討, O.A学会 D.P.S研究部会, 高原康彦(編)『システム設計の理論と実際』, 111/122, 1988. [13]佐藤ほか:情報システムのライフサイクルアプローチのIRDS上の実現について, 情報処理学会第40回全国大会講演論文集, 917/918, 1990. [14]Lo, A.: A Specification of IRDS1, internal report, 1990. [15]木嶋恭一:サイバネティック情報システム分析:組織内情報システムの設計と診断のための方法論, 情報通信学会年報, 51/65, 1987.