

セマンティックデータベースモデルに基づくデータベース管理システム：SIMにおける情報の表現

仁井谷 智也子

日本ユニシス株式会社

セマンティック情報管理ソフトウェア（SIM）は、セマンティックデータベースモデル（SDM）にもとづくデータベース管理システムである。SIMでは、データベースの対象をエンティティと呼んでいる。エンティティを表現する方法として”クラス”、”属性”、”特性”を導入している。”クラス”、”属性”、”特性”を用いることで、データの意味やデータ間の関連を定義し、データ間の整合性をとることが可能となった。

REPRESENTATION OF INFOMATION IN SIM

Ciyako Niitani

Nihon UNISYS Limited

1-10-9 Sinonome, Koutouku, Tokyo 107, Japan

The Unisys Semantic Infomation Manager(SIM) is a data base management system. The data model of SIM is similar to SDM. Entity is an object of data base in SIM. "class", "attribute" and "property" are mechanism to represent an entity. With "class", "attribute" and "property" you can define a semantic of a data and relationship between datas, and maintain the integrity of data relationships.

1 はじめに

これまでのデータベース管理システムでは、データ構造がもつ論理的制約や物理的制約により、現実の適用業務環境の構造を正確に反映することが難しかった。さらに、データの意味やデータ間の関連は、データベース管理システムではなく、データを操作する適用業務プログラムで管理していた。

これらの問題点を解決する手段として、ユニシス社のセマンティック情報管理ソフトウェア（SIM）が開発された。

SIMは、セマンティックデータモデルのひとつであるセマンティックデータベースモデル（SDM）を基礎としたデータベース管理システムである。SIMでは、1000、東京都、のようなデータの値そのものだけではなく、データの持つ意味やデータ間の関連も構造として表現できる。

2 情報の表現方法

SIMでは、データベースにおける対象物の最小単位をエンティティとよんでいる。エンティティとは、実世界の一部分を取り出し、データベースを構築する際に、興味の対象となる事象や物である。ある会社組織のデータベースを考えると、それぞれの部署や従業員ひとりひとりがエンティティとなる。

それぞれのエンティティの特徴や、エンティティ間の関連を表すために、次に説明する“クラス”、“属性”、“特性”を導入している。

2.1 クラス

クラスはエンティティの型（タイプ）をあらわす。各エンティティを、共通した特徴を持つエンティティごとに分類して作成した集合が“クラス”である。

あるクラスを構成するエンティティは、みな同じ性質を持っている。あるクラスを構成するエンティティは、エンティティを表現する“属

性”で、一意に識別できなくてもよい。すなわち、エンティティにはキーは必要ない。世界中の人間というクラスを考えてみる。人間クラスには、たくさんの鈴木さんがいるだろうし、誕生日が1989年10月7日の人もたくさんいるだろう。氏名や誕生日が同じだからといって同じ人物とは限らないのだから氏名や誕生日で世界中からたった一人を選びだせなくてもかまわない。世界中の人間の中からたった一人を選び出すための属性など実世界には存在しない。

クラスを構成するエンティティを、さらに共通した特徴を持つエンティティごとに分類して作成した集合が“サブクラス”である。サブクラスは、クラスの部分集合となる。サブクラスの“親”としての役割を果たすクラスをスーパークラスと呼ぶ。サブクラスは、さらにそのサブクラスを構成するエンティティを、共通した特徴を持つエンティティごとに分類して、サブクラスを作成することもできる。

図2-1では、社員クラスのスーパークラスは人間クラスである。社員クラスのサブクラスは管理職クラスである。

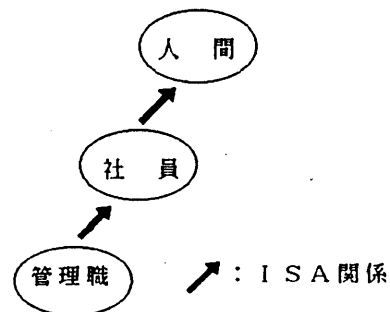


図2-1 汎化階層構造

スーパークラスーサブクラスは、上位・下位の関連を表しており、ISA関係である。スーパークラスーサブクラス・・・と集合を作ることによって、階層構造を作成できる。スーパークラス、サブクラスの階層構造は、下位の階層ほど

そのサブクラスが表現するものが限定され、具体的なものとなる。

2. 2 属性

エンティティを表現するために”属性”を用いる。属性は、それぞれの属性に対応した値とクラスで一意となる名前をもつ。属性に対応した値が存在しないときには、”値を持たない”という値（ナル値）を持つ。

SIMでは、クラスの構成要素すなわち表示可能なデータ値を値とする属性とエンティティを値とする属性を導入している。

(1) データ値属性

表示可能な値をデータ値として持つ属性。データ値属性はエンティティの特徴を表す。

ある人を表現する場合、その人の氏名や生年月日がデータ値属性である。

(2) エンティティ値属性

別クラスや同じクラスのエンティティを値としてもつ属性。エンティティ値属性はエンティティとエンティティとの関連を表す。エンティティとエンティティの関連の方向性も表す。

ある人をその人の”親”を使って表現する場合、その人の”親”を人と人との関連を表すエンティティ値属性にできる。”親”を親の氏名や生年月日であらわせば、”親”はデータ値属性となる。別の人への人間関係であらわせば、エンティティ値属性となる。

2. 3 特性

属性の特徴をあらわすのが”特性”である。特性には、データ型、複数値／単一値、その他オプションがある。

2. 3. 1 データ型

データ型は、特定のデータ値属性がとりうる値の集合を定義する。SIMでは、データ型によってデータ値属性のとりうる値に意味をもたせている。

データ型には、SIMがあらかじめ用意している型とデータベース利用者がみずから定義する型がある。

(1) SIMが用意しているデータ型

整数型 整数。-1、187。

実数型 浮動小数点。5.14X10⁸。

文字型 EBCDIC文字あるいは漢字（2バイト文字）。A、月。

論理型 TRUE（真）またはFALSE（偽）のいずれかの値。

時刻型 HH:MM:SSの値。ここでHHは時間、MMは分、SSは秒をあらわす。10:23:57。

日付型 MM/DD/YYまたはMM/DD/YYYYの日付。ここでMMは月、DDは日、YYとYYYYは年をあらわす。10/07/1989。

数字型 固定小数点。48.7。

文字列型 EBCDIC文字あるいは漢字（2バイト文字）の集合。BIRTHDAY、東京都。

複合型 一つ以上のデータ型で構成される型。複合型を構成するデータ型をなににするかは、データベース利用者が決定する。例えば、氏名という複合型の属性は、文字型属性の姓名と文字型属性の名前で構成すると定義できる。

記号型 記号を値とする。どのような記号を値とするかは、データベースの利用者が定義する。たとえば、属性性別を記号型属性と定義し、その値として男性と女性を定義する。すると、性別は、男性と女性のみを値としてとる。

(2) 利用者が定義するデータ型

利用者定義型 利用者が必要に応じて定義するデータ型。たとえば、データ型として、体重や身長を定義できる。体重や身長をデータ型とする属性は、どちらも数字を値とするが、型が異なるために、比較や演算の対象にはならない。

データ型に、値の”部分範囲”を指定することもできる。部分範囲指定により、データ値属性の値にさらに意味をもたせることが可能である。

2. 3. 2 複数值属性と単一値属性

属性（データ値属性あるいはエンティティ値属性）は、同時にとる値の数により複数值属性と単一値属性にわかれる。

(1) 複数值属性

同じデータ型の値を同時に複数個とる。

(2) 単一値属性

一つだけの値を持つ。

2. 3. 3 オプション

属性の特性にはこれまでに述べたデータ型、複数值属性と単一値属性に加えて次のオプションも定義できる。

- (1) 属性がとる値がユニークであること。
- (2) 属性は必ず値を持つこと。（すなわち属性はナル値を持たないこと。）
- (3) 複数值属性の値は、あるエンティティにおいては、お互いに異なっていること。
- (4) 複数值属性がとることができる値の数の最大数。
- (5) データ値属性の省略時値。
- (6) エンティティ値属性の逆関係

2. 4 属性の継承

一般にISA関係では、上位概念の性質を下位概念が継承する。SIMでは、サブクラスを構成するエンティティは、全てのスーパークラスを構成するエンティティの属性を全部継承する。さらに、サブクラスの詳細情報として、サブクラスの特徴を表す属性を定義できる。すなわち、サブクラスを構成するエンティティの特徴は継承した属性とそのサブクラスで定義した属性で表現される。

図2-2で、管理職クラスのエンティティがもつ属性は、次の5つである。管理職クラスの詳細情報である職階。スーパークラスから継承した属性の氏名、生年月日、入社年月日、社員コード。

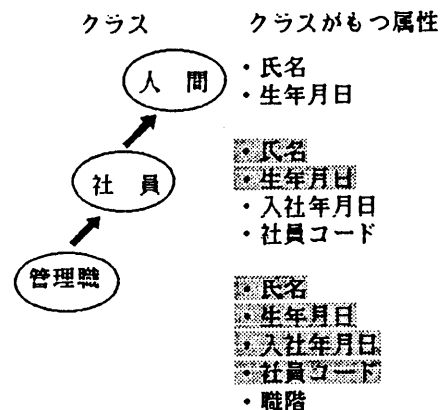


図2-2 属性の継承

3 意味の表現と管理

3. 1 視点

SIMでは、データベースを検索・更新するときに視点をもつことができる。視点をおいたサブクラス（あるいはクラス）から参照できる属性は、次の三種類となる。

- (1) スーパークラスから継承した属性。これを継承属性と呼んでいる。
- (2) 視点をおいたサブクラス（あるいはクラス）で定義した属性。これを直接属性と呼んでいる。
- (3) 視点をおいたサブクラス（あるいはクラス）を構成するエンティティとエンティティ値属性によって関連づけられた別のエンティティがもつ属性。これを拡張属性とよんでいる。

図3-1で示すように、視点を社員クラスにおいた場合、参照できる属性は、継承属性である氏名と生年月日、直接属性である入社年月日と社員コード、拡張属性である部署名と部署コードである。

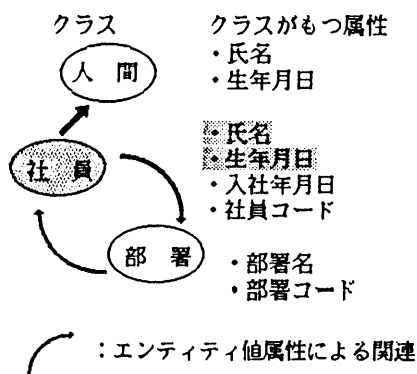


図3-1 参照可能な属性

3. 2 データ間の関連

エンティティ値属性によって、エンティティ間の関連をつけることができることを述べたが、エンティティ値属性に、特性を記述することで、さらに詳しく関連を表現できる。

エンティティ値属性を複数值属性にすると、1対1の関連だけではなく、1対多、多対多の関連も表現できる。

また、エンティティ値属性に”逆関係”の特性をもたせると、エンティティ値属性で関連づけられたエンティティは、相互に関連していることを表現できる。たとえば、エンティティ値属性の上司と部下を逆関係にすると、それぞれのエンティティ間には逆の関係が成り立つことになる。すなわち、ワトソン君の上司はシャーロック・ホームズであり、シャーロック・ホームズの部下はワトソン君である。

3. 3 データの意味

利用者は、データの持つ意味を表現するために、2. 2で述べたデータ型や部分範囲等の指定を使うことができる。

たとえば、利用者定義型のデータ型として体重と身長を定義したとすると、同じ数字の100でもデータ型体重とデータ型身長では意味が異なる。データ型が体重の場合には、重さが100キログラムを意味する。データ型が身長の場合には、背の高さ100センチメートルを意味する。

データ値属性に未成年という意味をもたせたい場合には、まず、データ型年齢を定義する。データ型年齢は整数を値とすると定義する。次にデータ型未成年を定義する。データ型未成年は、年齢を値としてとり、その範囲は、0から19までと定義する。最後に、未成年という意味をもたせたいデータ値属性のデータ型を未成年と定義する。データ型未成年のデータ値属性は、0から19までの整数のみを値とする。これは、未成年が年齢がゼロ才から19才までを意味するのと同じである。

3. 4 データの整合性

属性の値に、”特性”を指定することで、データベースに意味が定義される。SIMはこの意味に従ってデータの整合性を検査し保守している。

(1) データ間の関連にもとづく整合性

SIMはデータの更新時に、エンティティ間の関連を保証する。エンティティ値属性である上司と部下を”逆関係”であると定義しておく。すると、新入社員が配属になり、データベース利用者が、新入社員から配属先の部長へ上司の関連をつけると、SIMは部長から新入社員へ部下の関連をつける。社員が退職したため、データベース利用者がその社員のエンティティを削除すると、SIMは部長からその社員への部下の関連を解消する。

(2) データの意味にもとづく整合性

利用者が、3.3で述べたデータ型未成年の属性の値を20に変更しようとする、SIMは誤りとする。また、利用者が新たにエンティティを追加する際に、必須の属性がナル値であれば、SIMは誤りとする。これらは、データベースに意味が定義されているために、できることである。データベース管理システムが整合性の保守を行わないならば、これらの整合性検査は、適用業務プログラムの中で行わなければならない。

(3) 汎化階層構造にもとづく整合性

スーパークラス-サブクラスの汎化階層構造をつくっている場合、上位クラスからエンティティを削除すると、SIMは下位のクラスからも削除する。たとえば、人間-社員-管理職のクラスが汎化階層構造を作っているとする。不幸なことにある社員が亡くなったので、人間クラスから削除すると、SIMは社員クラスと管理職クラスからも削除する。また、エンティティの属するサブクラスを変更すると、そのサブクラスより下位のクラスからエンティティを削除して、データベースの整合性を保守する。社員が退職したため、社員クラスの構成要素からはずすと、SIMはその社員が管理職であれば、管理職クラスの構成要素からはずす。ただし、人間クラスの構成要素からは、はずさない（会社をやめても人間ですから）。

4 おわりに

SIMでは、データの意味やデータ間の関連を定義できる。これらの定義をもとにデータの整合性の保守もおこなっている。すなわち、SIMはデータの静的な意味を管理している。

<参考文献>

[1]M.Hammer and D.Mcleod, Database Description with SDM: A Semantic Data Model, ACM Trans. Database Systems, Vol.6 No.3 Septemb

er 1981, Pages 351-386

[2]Unisys Corporation, A Series infoExec™ semantic information Manager (SIM) Technical Overview

[3]Joseph & Cogan Associates Inc., INTRODUCTION TO INFOEXEC