

オブジェクトモデル化のためのメタファ構造化

— JDMF / BMの適用について—

堀内 一
東京国際大学

森田勝弘
日本銀行

JDMF及びその拡張としてのJDMF/BMは、オブジェクトモデル化基準として、他にない、いくつかの特徴をもっている。

その一つは、原子オブジェクトとそれに基づくオブジェクト集約階層に関する枠組みを持つことである。その枠組みは、実世界における概念をオブジェクトやその関連で表現するだけでなく、オブジェクトそのものの組成 (ingredient) を明らかにし、オブジェクトの共有や実現を容易とするものとなる。

つまり、実世界の関心領域を表現することだけに捕らわれ、関心領域の偏りそのものを取り除く過程をもたないモデル化は、オブジェクトモデルの必要条件を満たしても、オブジェクトの共有性や重複排除、あるいは安定性、などの十分条件を満たさない。オブジェクトモデル化における、十分条件検討の欠如は実システムにおける再利用やオブジェクト実現に大きな支障を来す。

本稿では、JDMFの特徴をモデル化のメタファ (暗喩) と結びつけて、オブジェクト再利用、あるいはモデル一意性を目的としたモデル化の指針とすべくその構造と構成概念を論じる。また、例題として、ISO/IEC/SC21/WG3 CSMFプロジェクトで共通例題として提案された「ホテル予約問題」を採り上げる。

Structuring metaphor for Object Modeling

Hajime HORIUCHI,
Tokyo International University

Katsuhiro MORITA,
Bank of Japan

JDMF(JSA Data modeling Facility) proposed by INSTAC Information Resource Schema Research Project, has several advantages in the object modeling capability. One of which is having an object aggregation mechanism based upon atomic object and typical modeling constructs. This mechanism allows the sophisticated sharing and easy implementation of object rather than explaining the real world concept using object and relationship, by providing the explicit definition capability of ingredients in each objects.

In this paper, the identification of modeling metaphors and its hierarchical structure which can be used in the object modeling efforts, are investigated. Also, a prototype system which was submitted to the ISO/IEC JTC1 SC21 CSMF project as a common example, is discussed in the term of dynamic behavior modeling.

1 はじめに

JDMF及びその拡張としてのJDMF/BMは、オブジェクトモデル化基準として、他にない、いくつかの特徴をもっている。

その一つは、原子オブジェクトとそれに基づくオブジェクト集約階層に関する枠組みを持つことである。その枠組みは、実世界における概念をオブジェクトやその関連で表現するだけでなく、オブジェクトそのものの組成(ingredient)を明らかにし、オブジェクトの共有や実現を容易とするものとなる。

つまり、実世界の関心領域を表現することだけに捕らわれ、関心領域の偏りそのものを取り除く過程をもたないモデル化は、オブジェクトモデルの必要条件を満たしても、オブジェクトの共有性や重複排除、あるいは安定性、などの十分条件を満たさない。オブジェクトモデル化における、十分条件検討の欠如は実システムにおける再利用やオブジェクト実現に大きな支障を来す。

本稿では、JDMFの特徴をモデル化のメタファ(暗喩)と結びつけて、オブジェクト再利用、あるいはモデル一意性を目的としたモデル化の指針とすべくその構造と構成概念を論じる。また、例題として、ISO/IEC/SC21/WG3 CSMFプロジェクトで共通例題として提案された「ホテル予約問題」を採り上げる。

2 オブジェクトモデル化のメタファ

2.1 モデル化とメタファ

メタファ(metaphor: 隠喩、暗喩)とは、物事を認識する上で、認識する主体が予めもつ仮説的構造、類似構造、あるいは、その構造を構成する要素(記号)に抱く概念を指す。

そもそも、モデル化とは対象を何らかのメタファで捉えて表現することであり、その結果としてのモデルは作成者のメタファに依存する。また、作成者のメタファも作成者自身が抱く問題意識によって大きく変化する性格を持つ。

オブジェクトモデル化における多くの混乱は、メタファ自体の多様性に無関心であるか、依拠するメタファを特定せずにモデル化を進めるか、あるいはモデル化目的とメタファの不一致、などに基づいている。

したがって、本来、何らかの目的を与えられているモデル化作業では、その目的に合致したメタファの特定と共有が不可欠となる。

そこで、モデル化のメタファ特定を容易とするために、メタファ自体のタクソノミとその共有が重要となる。

2.2 モデル化メタファに関する論議

オブジェクトモデル化のメタファに関する論議の多くは、これまでデータモデル化技法、あるいはソフトウェアエンジニアリングの構造化手法として論じられてきたものと、ほぼ同じである。

これまで用いられてきたオブジェクトモデル化メタファの代表的なものは次のようなものである。

(1) ER(実体関連)

実世界(UOD)を実体とその関連で捉えようとするメタファ。古くからデータモデルのメタファとして使われてきた。

(2) MVC(ModelViewControl)

Smalltalkで用いられたユーザインタフェースに関するメタファ。

(3) デザインパターン

E. Gamma[1]らが提案するオブジェクト設計用のオブジェクトモデルひな形。酒井、Wirfs-Brockらがオブジェクトフレームワーク[2][3]と呼ぶものと同じ。

しかしながら、これまでのモデル化メタファに共通することは、対象領域(UOD)または問題意識に直接貢献するモデルを重視していることである。その結果、対象を表現できても、その実現に関する規定やガイドがなかったり、共有性の吟味が欠如する傾向にあった。

3 モデル化メタファの構造

3.1 メタファの相互関係

オブジェクトモデル化は、その目的、意図、あるいは支援対象によって異なる。それらの相違を明らかにしながら、それぞれのモデルの位置づけを行うためのモデル化メタファは、図1に示すような多次元のものとなる。

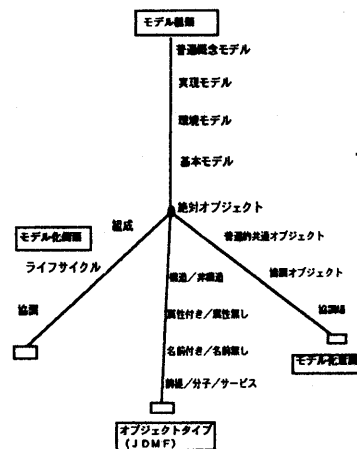


図1 モデル化メタファの軸

3.2 モデル化目的

モデル化は、本来、明確な目的と意図をもつ。モデル化の目的を「実世界の忠実な表現」に置いても、実世界の全てを表現することは不可能である。実世界の何を表現するのかはモデル化の意図に依存するからである。

問題意識によってモデル化で用いられるメ

タファは異なり、捉えられるモデルも異なる。したがって、先ずメタファはモデル化目的によって分類されるべきである。

(1) モデルの種類

- オブジェクト指向は、図2に示すように、
- 異なるコミュニティ間での情報共有
- ソフトウェア開発、保守の効率化
- 分散環境の維持、コモディティソフトウェアの連携確保

などに大きな意義を持つものと言える。それぞれの目的を達成する上で、つぎのような固有のモデルが必要となる。

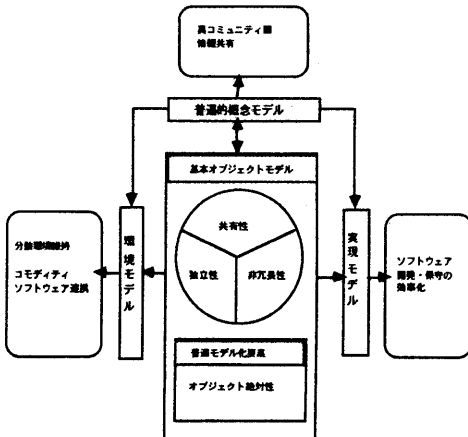


図2 オブジェクト指向の目的とモデル種類

● 普遍的概念モデル

異なる情報システムによって支えられている独立なコミュニティの間で、情報を交換するために必要となるモデル。EDIなどがその代表例である。可能な限り、システムの実現方式や、特定なシステム環境から独立なモデルである。

● 実現モデル

特定なアプリケーション、特定なドメインを対象としたモデルで、そこにおける情報システムを開発、保守する上で求められるモデル。開発や保守に関する要求、あるいはシステム構成に関する仕様も含まれる。

● 環境モデル

異なるアーキテクチャをもつコンピュータシステムやソフトウェアシステムによる分散環境間で、商用のコモディティソフトウェアを連携させる上で求められるモデル。

● 基本オブジェクトモデル

上記の目的とは独立に、オブジェクトを構成させる上で求められるモデル。目的別に求められるモデルの要素(オブジェクト)を規定するもので、オブジェクト構成原則と普遍的なモデル化要素から成り立つ。

システムの開発、活用などの目的に依存したモデルと、それらの目的から独立なモデルとの相違は、かつて、データモデルを、情報論理モデル(Info-logical Model)とデータ論理モデル(Data-logical Model)とに分けた考え方と似ている。しかしながら、すべてのモデルは、明確な目的概念を持つ。図3は各モデルがもつ目的概念を示すものである。

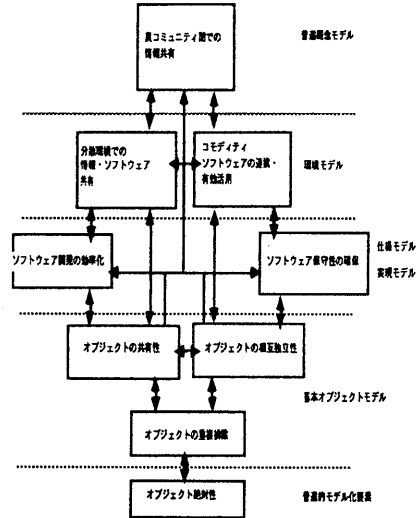


図3 モデル化と目標概念

3.3 モデル化側面

モデル化側面とは、オブジェクトのいかなる側面、あるいは対象のいかなる側面を記述するかを示すものである。それぞれの側面は、次のような意図をもつ。

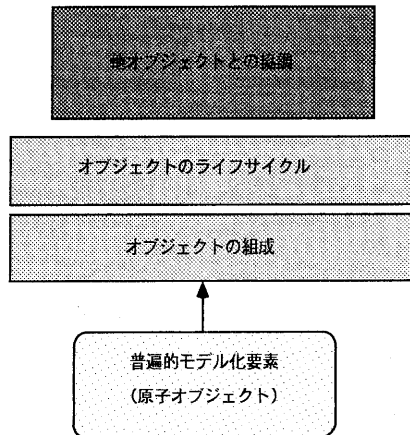


図4 モデル化側面の階層

- (1) 原子オブジェクトと絶対性
オブジェクトの絶対性とは、そのオブジェ

クトの存在がすでに広く認められており、論議する必要がなく、その特性も時間と共に変化しないことを指す。

しかし、オブジェクトの多くは、特定の目的をもつモデルを構成し、その目的からの偏りを受ける。偏りをうけるからこそ、その領域に特化した意味を表現できる。

「社員」という特化したオブジェクトも「人間」としてのオブジェクトの存在が先でその制約や振る舞いを継承する。さらに「人間」というオブジェクト概念を明らかにするために、それを記述するより小さな概念を使用せざるを得ない。例えば、氏名、生年月日、性別、血液型、身長、などである。それらの小さな概念は、最後には、「人名」、「日付」、「長さ」など、それ以上説明を要さず、永続性のあるものに到達する。そのような「絶対的」オブジェクトを「普遍的モデル化要素」と呼ぶことにする。JDMFで原子オブジェクトと呼ぶものに該当する。

ここで原子オブジェクトは次のような特性をもつ。

- その存在は公知のものであり、永続的である。
- 値は変化しない。
- 「制約」をもつが、「ライフサイクル操作」をもたない（関知しない）。

原子オブジェクトの数は経験的に、極めて少ない（100～200）のものであり、その事前列挙は実務的にも可能である。列挙された原子オブジェクトに基づき、上位の集約オブジェクトを構成させることは、その集約オブジェクトについて明示的に記述できない側面（暗示領域）を補うことができる。

（2）オブジェクトの組成

原子オブジェクトを組み合わせる表現される集約オブジェクトは、絶対的な原子オブジェクトに比して何らかの意図をもつ相対的な存在と言える。

オブジェクトモデル化の使命の一つは、豊富な意味をもつ様々な相対的なオブジェクトを、そのような絶対的な原子オブジェクトを用いての組成を明らかにすることで、その相互比較可能性、共有性、非冗長性、あるいはオブジェクトの相互独立性の確保を可能とするにある。

また、オブジェクトの基本組成を導くメタファは次のようなものである。

一般形：

- 核（データまたはオブジェクト）
- 制約
- ライフサイクル操作

図5は原子オブジェクトから集約されるオブジェクトの組成に関するメタファを示すものである。

各種のオブジェクト型は、JDMFによっ

てその概念が与えられた。それに従って、それぞれのオブジェクトタイプを決定するのが、何らかの意図を持つモデル化であり、そのモデル化いとはによってオブジェクト型を導くものを「ビュー」と考える。

「属性ビュー」とは、構造オブジェクトのサブタイプである「属性つきオブジェクト」を構成させるときの手段で、モデル化の意図に従って、どのような属性を付与するかを決めるものである。

「集約ビュー」とは、JDMF/BMで拡張された「前提オブジェクト」、「分子オブジェクト」、「サービスオブジェクト」をモデル化の意図に従って使い分ける手段である。

同じ名前付きオブジェクトであっても、あるものは、そのライフサイクル操作に着目する必要はない。

（3）オブジェクトライフサイクル

オブジェクトの発生から消滅に至る全ての過程をライフサイクルという。すべてのオブジェクトは基本的にライフサイクル操作をもつが、「名前付きオブジェクト」と「協調オブジェクト」だけにライフサイクル操作を考えればよい。

ライフサイクル操作に似たものに、協調操作がある。ライフサイクル操作は、あくまでも当該オブジェクトの発生、消滅、変更に関する操作をさす。協調操作は、二つ以上のオブジェクトの相互作用によって引き起こされるアクションを指す。ただし、そのアクションの一つとしてライフサイクル操作が存在するので、明確な分離は難しい。

しかしながら、オブジェクトの相互作用を分析する前に、当該オブジェクトのライフサイクル操作を洗い出すことは困難ではない。

（3）他オブジェクトとの協調

UODの意味、概念は、先ず、オブジェクトの関連で捉えられる。しかし、問題意識（モデル化意図）によって、関連するオブジェクトが異なる。

モデル化意図によって、複数オブジェクトから構成される「オブジェクト協調場」と「集約ビュー」が形成される。

複数オブジェクトの関連は、オブジェクト「相互の参照」とオブジェクト「相互の動的な作用」によって発生する。

JDMF/BMでは、どちらもオブジェクトの集約化で取り扱われる。

「オブジェクト協調場」はオブジェクトフレームワーク [2] [3] で論じられているものと同じといえる。協調場に参加するオブジェクトをパートナーオブジェクトと呼ぶこともある。オブジェクト相互の作用を、「契約と委託」関係と見なす、あるいは「シナリオ」とよぶメタファも提案されている。

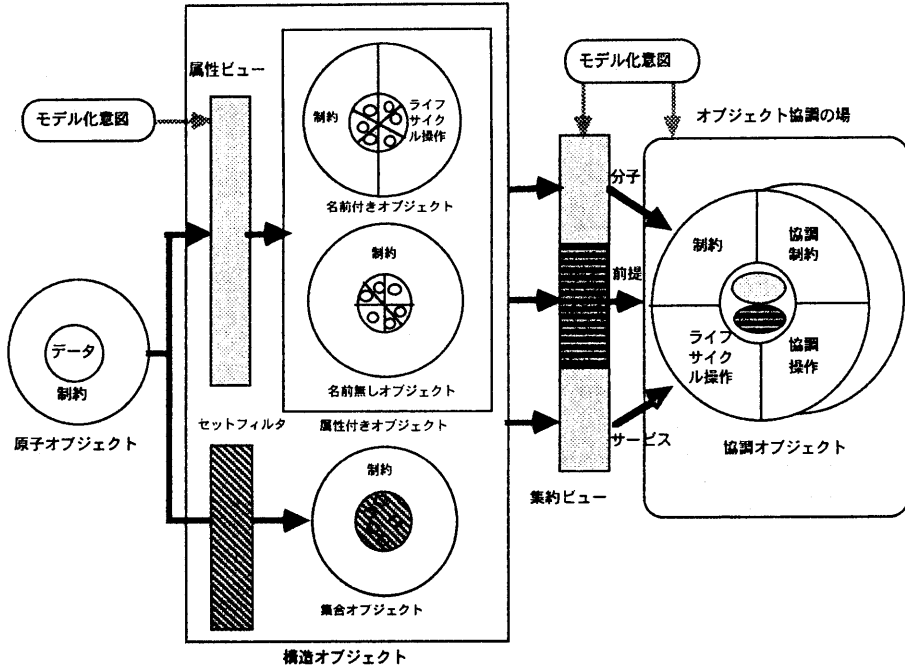


図5 オブジェクト組成のメタファ

3.4 モデル化意図と集約階層

オブジェクトの集約は、モデル化意図によって異なるものが求められる。集約オブジェクトがさらに上位のオブジェクトに集約されることも多い。

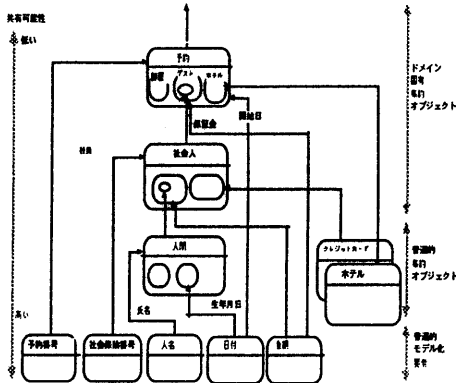


図6 オブジェクト集約の例

図6に示すように、実世界の概念を捉えるためには多階層の集約が求められる。このとき、集約の階層が高くなるほど、特化が進み再利用可能性は低くなる。

集約階層の低位のものほど再利用の可能性

が高いことは、そのプロジェクトが多目的に使われることを意味し、そのオブジェクトの存在を、上位の集約オブジェクトからは決定できない。

例示における「予約」オブジェクトは「協調オブジェクト」であり、「社会人」オブジェクトは「予約」オブジェクトを構成する「前提オブジェクト」となる。「予約」オブジェクト側から「社会人」オブジェクトを見ると、「ホテル予約」という協調場では、その社会人のライフサイクルを決定できないし、その必要もない。

しかし、「社会人」オブジェクトについて、別な協調場「社会保険」などでは、ライフサイクルが重要となる。

JDMF/BMにおける「前提オブジェクト」、「分子オブジェクト」、「サービスオブジェクト」などのメタファ「オブジェクトタイプ」は、それらの階層をガイドする有効な手段となる。

4 オブジェクトモデル化手順とメタファ

オブジェクトモデル化に当たって、低位のオブジェクトほど先に定義されている必要があり、原子オブジェクトだけでなく、共通に利用される「普遍的集約オブジェクト」の列挙とライブラリ化が重要となることは言うまでもない。

そこで分析作業としては、

- (1) 既存データ分析に基づく原子オブジェ

クト列挙

- (2) 普遍的集約オブジェクトの事前洗い出し
- (3) 協調オブジェクト定義段階における下位オブジェクトへのフィードバックが重要となる。

ここで、特に「普遍的集約オブジェクト」は独立した「名前付きオブジェクト」となり、その洗い出しを特定なモデル化メタファとして、例えば、PLAN-DB手法[6]などで用いられている、「リソース型」「イベント型」などの「オブジェクト性格」に関するメタファも有効である。

図7は、オブジェクトモデル化作業と各種メタファの関連を示したものである。これらのメタファを整備して、標準化することでオブジェクトモデル化、オブジェクトの設計は分析、設計プロセスに関するいたずらに詳細な規定を用いることなくガイドすることができる。

オブジェクトフレームワークの標準化として期待されていることの一つである。

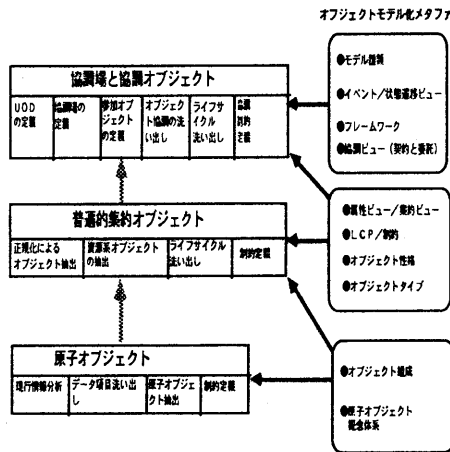


図7 モデル化作業とメタファ

5. 例題 (ホテル問題)

ここでは、米国から提案の「国立公園サービスのためのホテル予約システム」を対象に、業務要件の整理、静的モデルの記述、動的モデルの記述の順でJDMFによるモデリング例を紹介する。

5.1 業務要件の整理

(1) 業務仕様

国立公園サービスにおけるホテル予約のプロトタイプ・システムに関するものであり、以下の要件からなる。

S01: この予約システムは、のOld Faithful Inn Hotel (イエローストーン国立公園)、El Tovar Hotel (グランドキャニオン国立公園)、およびAhwahnee Hotel (ヨセミテ国立公園) を取扱い

対象とする。

S02: それぞれのホテルは、そのホテル名により一意識別できる。

S03: 各ホテルとも、通常の所在地(番地、町丁目、市町村、州、郵便番号)のほかに、電話番号とファックス番号を情報として持つ。

S04: モービル・トラベル・サービスによる格付ランク(1~5個までの星数で表される)を持つ。

S05: 各ホテルは、それぞれ毎年所定の期間だけ営業する。

S06: 各ホテルとも、需要動向に応じた季節料金を適用する。つまり、季節別部屋タイプ別に定額の料金体系を定めている。

S07: 予約は13ヶ月前から受け付ける(たとえば、1995年の1月宿泊分は1994年1月から予約を受け付ける)。

S08: ただし、創業予定日前、あるいは廃業予定日後の予約は受け付けない。

S09: ホテルの部屋は、幾つかのタイプに分かれ、かつそれぞれの部屋タイプごとに禁煙と喫煙の区分がある。

S10: 部屋タイプや喫煙区分の変更は、一定の日から有効となる。

S11: 個人予約は、個人が直接申込みのものほかに、旅行代理店を通すものがある。

S12: 予約は、特定の一期間、特定の部屋タイプと喫煙区分(複数可)を指定して行う。

S13: 個人の場合は、予約時に氏名、住所、電話番号、およびもしあればファックス番号を伝える。

S14: 旅行代理店経由の場合には、それらの情報のほかに、代理店名と代理店識別番号を伝える。

S15: この代理店識別番号は、予約申し込みをする旅行代理店と、その所在地、電話番号、およびファックス番号を識別するための番号である。

S16: 予約は、予約係が受付処理を行い、個々の予約ごとに一連番号を付与する。

S17: 予約処理を終ると、その証跡のために時刻印を付与する。

S18: 空き部屋がない場合、予約係は、顧客に待ちリストに登録するかどうかをたずねる。

S19: 待ちリストの登録にあたっては、特定のホテル、あるいは同じ国立公園内にある等格ランクの(星数が同じ)ホテル・グループのいずれかを指定することができる。

S20: 待ちリストへの登録情報は、申込みの日から所定の日数だけ有効となる。

S21: その有効期限内に、希望どおりの部屋がとれた場合は、自動的に予約扱いとなる。

S22: 顧客は、待ちリストへの登録の有効期限が過ぎる前に、さらに有効期限を延長するか、あるいは空きのある部屋タイプや喫煙区分に変更することができる。

S23: 予約が成立した場合、利用予定日当日16:00までに到着するか、あるいは保証を入れておく必要がある。

S24: 16:00までに到着すれば予約は有効であるが、16:00を過ぎるとその予約は違約金なしで自動的に取消される。

S25: 16:00以降も予約を確保するためには、顧客は保証を入れる必要がある。

S26: 保証は、クレジット・カードによって行う。顧客は、保証に利用するクレジット・カードの種類と、その番号、および有効期限を伝える。

S27: 保証付予約を行った顧客が姿を見せず、ホテルにその旨の通知がなかった場合には、予約した部屋の初日分の料金がそのクレジット・カードに対し課金される。

S28: 顧客は、旅程の変更があった場合には、予約を取消し、待ちリストに登録されている他の顧客

に対し部屋を明け渡すことを励行するように仕向けられている。

S 29 : 保証付予約は、当日の16 : 00までならば取消しが可能で、取消しを行った場合には、顧客が課金されないことを証するための取消番号を受け取る。

S 30 : 保証付予約を当日16 : 00以降に取消した場合には、ホテルはその部屋を埋める努力をする。

S 31 : その部屋が埋まった場合は（取消した時間にもとづいて決まり、部屋のタイプや喫煙区分は考慮されない）、課金されない旨がその取消を行った顧客に通知される。

S 32 : 部屋が埋まらなかった場合は、ホテルの管理者の裁量により、課金するか否かが決定され、その結果が顧客に通知される。

S 33 : 予約が成立した時点で予約確認書が顧客に送付される。

S 34 : 予約確認書には、証跡のためにその予約成立時点の時刻印が付される。

S 35 : また、予約確認書には所定の予約関連情報がすべて記載される。

S 36 : 取消番号が発行された時点で、取消通知が顧客に送付される。

S 37 : 取消通知書には該当の予約と、その取消処理を行った予約係が明記される。

S 38 : また、取消が所定の期限内に行われ、違約金が不要であることを証する時刻印が付与される。

S 39 : チェックイン処理は、顧客がホテルに到着したときに行われる。

S 40 : チェックイン時刻が記録され、出発日の確認が行われる。

S 41 : 部屋番号が割当てられ鍵が顧客に渡される。

S 42 : クレジット・カードによる支払いの場合には、所定の事項が確認される。

S 43 : 顧客は、予約の保証に用いたものとは別のクレジットカードを利用することができる。

S 44 : 国立公園サービスでは、ホテルの請求額が一定の限度額を超えて、さらに顧客が滞在を続けようとする場合には、途中で清算を求め方針を採っている。

S 45 : チェックイン処理の際に、顧客にその与信限度額が伝えられる。

S 46 : 柔軟な予約方針を採っているため、16 : 00直後の振りの立寄り客は特に歓迎される。

S 47 : チェックイン時に、顧客は、特定の一部屋を指定し、他の部屋の課金をその部屋の請求書にまとめようように指示することができる。これにより、家族が複数の部屋を利用したときの課金請求を一本化することができる。

S 48 : その場合の請求書には、費目明細ごとに該当の部屋が示され、顧客が容易に明細確認ができるようになっている。

S 49 : 与信限度額に達するたびにきちんと途中清算を済ませている顧客は、通常は、出発日にチェックアウトする。

S 50 : 特定の部屋に課金請求をまとめようように指示した顧客の場合は、その指定の部屋の顧客がチェックアウトする時に、全員一緒にチェックアウトする必要がある。

S 51 : チェックアウトは、12 : 00以前に済ませる必要があるが、申し出によっては14 : 00まで延長することができる。

S 52 : 所定のチェックアウト時間を過ぎた場合には、超過料金を課金される。

S 53 : 不用意に超過料金が課金されることを防止するため、チェックアウト時の時刻印を明示する。

S 54 : 顧客は、チェックアウト時に、すべての料金明細と収支状況を記録した請求書の写しを受取る。

S 55 : 所定の与信限度額に達したとき途中清算を

行わなかった顧客は自動的にチェックアウトされる。
S 56 : 宿泊の延長は、部屋に空きがあれば、いつでも受け付けられる。

(2) 業務仕様の調整・補完

前述の業務仕様には、相互に矛盾するものや明記されていないが常識で補うことが望ましい点などが残されている。したがって、この先のモデリング作業を円滑に進めるために必要な範囲で、以下のような調整・補完を行う。

R 1 : S 0 1 では、予約対象ホテルが3つの国立公園内のそれぞれ1つのホテルに限定されているが、S 1 9 では、同じ国立公園内にある等格ランクのホテル・グループを指定できることとなっている。この点は、S 0 1 で上げているもの以外にも予約対象ホテルが存在するものとする。

R 2 : S 1 0 は、このままでは意味がはっきりしないため、「部屋タイプや喫煙区分の変更は特定の日（滞在期間内）を指定して行う」と読み替える。

R 3 : S 1 2 では、予約にあたってホテル名を特定するようにはなっていないが、S 1 9 との整合性を考えて、特定のホテル名を指定するものとする。

R 4 : S 3 1 の予約取消時の課金されない条件はあいまいなため、ここでは「取消した時点から一定の時間内に部屋が埋まったときには課金されない」扱いと考える。

R 5 : S 4 2 のクレジット・カードに関する所定の情報は、S 2 6 と同様に、カードの種類と、その番号、および有効期限とする。

5. 2 静的モデルの記述

JDMF/M-92は、オブジェクト間の静的な関連を記述することに狙いをおいた概念モデリング機能であり、この適用事例における静的なオブジェクト関連は、図8のように表される。このオブジェクト関連図では、簡単のために各オブジェクトの属性記述は省略して示してある。また、それぞれのオブジェクトや関連に対応する業務要件の番号(Snn)を記してある。ただし、この静的モデリングの例では、動的な振る舞いを的確に記述することに限界があるため、一部の業務仕様については以下のような扱いとする。

(1) 記述が困難なためモデリングに含めることを割愛

有効期間や時間帯制約のうち、それらの時点が固定ではなく、通常の属性定義による表現が困難、あるいは不自然となるものはモデリングに含めることができない。また、ある種のイベントに応じてオブジェクトの状態が変化する、あるいは、他のオブジェクトのインスタンスを発生するような動的な振る舞いについても、同じく記述が困難なため、モデリングに含めることが不適当といえる。具体例としては、以下の業務要件がこれらの条件に該当する。

S 0 7 予約は13ヶ月前から受け付ける。
S 2 1 希望通りの部屋がとれた場合は、自動的に予約扱いとなる。
S 2 3 利用予定日当日の16 : 00までに到着する必要がある。

S 2 4 16:00を過ぎるとその予約は自動的に取消しとなる。
 S 3 1 予約を取消した部屋が埋まった場合は、課金されない。
 S 3 2 部屋が埋まらなかった場合は、ホテルの裁量により課金される。
 S 5 6 宿泊の延長は、部屋に空きがあれば、いつでも受け付けられる。

(2) 不十分ではあるが表現可能な範囲内で部分的に記述する

動的な時間制約やイベント条件などの振る舞いをきちんと記述することは無理としても、その結果として生ずるオブジェクトの定常状態や関連については静的なモデリングで表現可能なものがある。そうした業務条件については、部分的ではあるがモデリング対象に含めることができる。これに該当する例としては、以下のものがある。

S 2 5 16:00以降の予約確保のためには保証を入れる必要がある。
 S 2 7 保証付予約の顧客が現れなかった場合は、ペナルティが課金される。
 S 2 9 保証付予約は、当日16:00までならば取消し可能。
 S 5 1 チェックアウトは原則12:00前で、14:00まで延長可能。
 S 5 2 所定のチェックアウト時間を過ぎると超過料金を課金される。

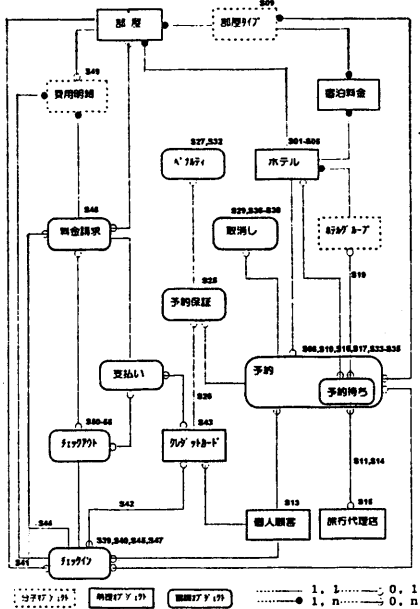


図 8 オブジェクトモデル図

5. 3 動的モデルの記述

JDMF/M-92では困難であった動的な振舞いの記述を可能とする機能拡張(JDMF/BM)が検討されている。

前述の5. 2で示した静的モデルに登場する「前提オブジェクト」を中心に、その状態遷移とイベントの関連を例示すると、図9のようになる(簡単のために、部屋の状態遷移や保証付予約に関するイベントなど一部の要求仕様を割愛している)。こうした、状態遷移とイベントとの関連を記述することにより、静的モデルでは記述が困難であった時間に関する制約条件や、予約待ちから予約確定への状態遷移なども明確に表現できるようになる。

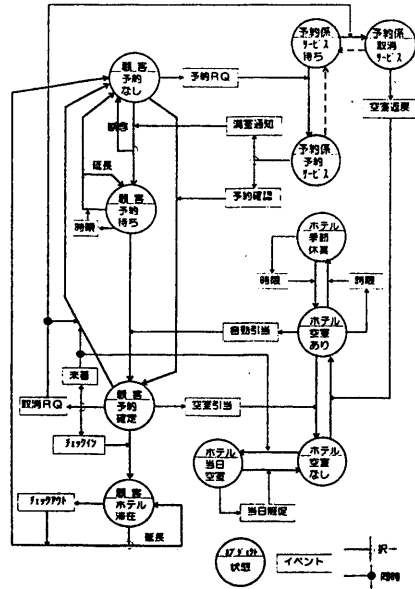


図 9 状態遷移図

【参考文献】

- [1] E. Gamma, et al, Design Patterns, Addison-wesley Profetional Computing Series, 1994.
- [2] R. Wirfs-Brock, et al, Designing Object-Oriented Software, Prentice Hall, 1990.
- [3] 酒井博敏、堀内一、オブジェクト指向設計、オーム社、1993.
- [4] 佐藤英人、JDMF/BM動的モデル拡張、情報資源スキーマ調査研究委員会資料、1995.
- [5] 情報資源スキーマ調査研究委員会、平成5年度、平成6年度報告書.
- [6] 椿 正明、DOA入門、オーム社、1994