

# 基幹情報システム設計における 概念モデルの変遷について

児玉公信<sup>†1</sup>

**概要**：情報システムの構想フェーズから実装フェーズにかけて、モデルは概念の整理状況および実装検討を反映してさまざまに変遷する。それは、建築設計における藤村龍至の超線型プロセスの一連のモックアップに似ている。本報告では、静的モデルがどのような契機でどのように変化したかを追跡し、概念形成の過程と設計者の思考の展開を探る。

**キーワード**：基幹情報システム、モデル変化、概念形成、概念モデル

## On a Series of Changes in a Conceptual Model during Enterprise Information Systems Designing

KIMINOBU KODAMA<sup>†1</sup>

**Abstract**: In developing enterprise information systems, from the planning phase to implementation phase, the models continuously change in various ways, reflecting the conceptual understanding state and implemental discussion. It is similar to a series of mockups of the super linear process of Ryuji Fujimura in architectural design. In this report, we will see how the static models were changed, and track the difference, and explore the process of concept formation and the deepening of the thought of the designer.

**Keywords**: Enterprise Information Systems, changes of a model, concept formation, conceptual model

### 1. はじめに

企業（Enterprise）の情報システム、とりわけ取引処理に関わる情報システムは、その企業の発展の歴史とともに長い時間をかけて成熟してきた、その企業のビジネスの根幹を支える基幹のシステムと言える。

10年前に著者は、そのような生産管理のための基幹情報システムの一般モデルを提案した。このモデルはその後、さまざまな業種の基幹システムに拡張されながら適用されて、時間を経て、言わば鍛えられて成長してきた[1,2,3]。本報告では、このモデルがどのように変化してきたかをレビューする。こうしたモデルの変化の意味を問うことは、モデラの思考プロセスを跡づけることで、モデルの妥当性を検証するだけでなく、取引処理以外の情報システムにおいても起こりうるモデルの変化を予測することであり、何よりも、よいモデルを追求する活動の一環でもある。

### 2. 取引処理の一般モデル

#### 2.1 取引処理

ここで取り上げる取引処理とは、企業取引において顧客から注文（Request）を受け取ってそれを適切に処理して価値を移動して、その対価を得るまでの一連の処理を指す。注文品は商品や製品だけでなく、サービスもある。注文の

内容は販売注文だけでなく、企業内部から発生する購買注文や製造注文、請求注文などもある。取引処理は、大筋業種を問わず共通であるため、概念モデルに関する議論は共有できる。

#### 2.2 取引処理の一般モデル

ここで言う一般モデルとは、これまでの業務知識やシステム設計のケース経験を基に、取引対象やビジネスプロセスの違いを超えて、取引のとらえ方、処理のとらえ方を、一般知識や基本的な概念のモデル化を通して再定義したものをいう。モデルの表現力にも依存するが、ひとたび一般モデルが定義されれば、さまざまな取引対象や業務プロセスにおいて差分適用が可能となり、短時間で一定品質の基幹情報システムの構築が期待できる。

#### 2.3 CHARM

筆者らは2008年に、少量多品種型生産管理システムの一般モデル CHARM を提案した[1]。CHARM は Cross Hierarchical Account=Resource Model の頭字語で、資源階層を越えた取引処理の概念モデル（型レベルの静的モデル）を意味する。これは、もともと会計仕訳のモデルである勘定パターン[4]を、生産資源の残高（残量）の増減記録としてとらえて拡張したものである。本報告では、CHARM のモデル表記上の変化を追跡するので、CHARM の概略を述べる。また、その基となった勘定パターンの構造と意味について触れる。

<sup>†1</sup> (株)情報システム総研  
Information Systems Institute, Ltd.

### 2.3.1 勘定パターン

勘定パターンは総勘定元帳 (General Ledger; GL) のモデルである。これは会計仕訳を記録するためのよく知られた定番のモデルであり、多くの会計パッケージなどで用いられている。

このモデル (図 1) は、1 回の取引に伴って、2 つ以上の勘定間で移動する金額を記録している。通常の会計仕訳とは異なり、借方と貸方で表現しない。貸借平均の制約は、「取引」オブジェクトにリンクする 2 つ以上の「移動」オブジェクトの合計が 0 となるという形で課している。ある時点 (たとえば現在) から見た別の時点 (たとえば前月末) のある勘定 (たとえば「現預金」) の残高は、記録日が現在より過去で取引日が前月末より過去の「取引」オブジェクトにリンクされる「移動」オブジェクトのうち「現預金: 勘定」にリンクしているものの金額を合計する。このことは、任意時点の残高をいつでも得られることを意味する。

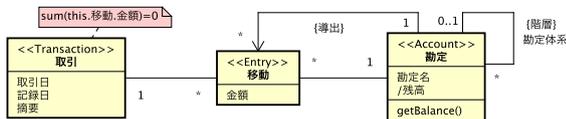


図 1 総勘定元帳のモデル (勘定パターン)

### 2.3.2 勘定パタンの在庫管理への拡張

Fowler は、Analysis Patterns[4]の中で勘定パターンを紹介した後、これを在庫管理業務に拡張して用いる例を挙げている (図 2)。これは、勘定パタンの「勘定」型を「品目」と「場所」の関連型である「保有」に置き換えた形になっている。これによって、同じ品目を異なる場所に移動する



ことが記述でき、ある品目がどの時点でどこにいくつあるかを導出できる。

図 2 在庫管理のための拡張

### 2.3.3 勘定パタンの生産管理への拡張

生産管理におけるさまざまな取引処理を行うために、在庫管理のモデルをさらに拡張したのが CHARM (図 3) である。

生産管理業務では、顧客注文および生産注文に基づいて作業計画を立て、資源の引当可能残 (在庫や処理能力) を引き当てて作業者に作業指示を出し、作業結果を実績として登録し、予実差異を反映して作業計画を調整する。この一連の機能を実現するために、上記の在庫管理のモデルに対して次の拡張を行った。①「取引」は予定オブジェクトと実績オブジェクトをもち、互いに対比可能とする (図 4)、②残を所有する「保有」は、「品目」に限定せず「設備」や「荷」などを広く「資源」として取り上げる (そのため、型の名前を「資源保有」とする)、③多様な「資源」をシン

ブルに扱うために、「資源保有」をパラメトリックに定義しておき必要になった時点で生成するである。

このほかに、「取引」あたりの資源の増減量についての貸借平均の制約を外すなどの変更を加えた。

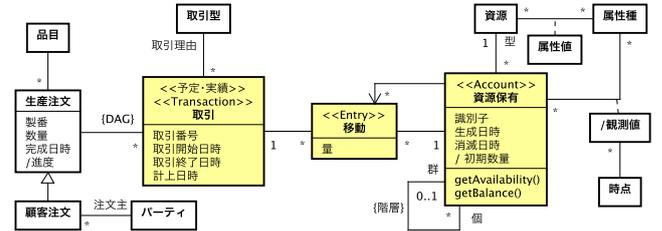


図 3 生産管理のための拡張

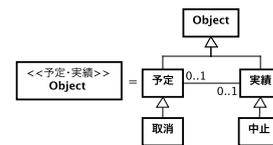


図 4 予定と実績

## 3. モデルのマクロな変化

建築家の藤村龍至は、設計過程においてモックアップを一つ作っては施主に見せ、そのフィードバックを基にすぐにモックアップを一つ作るということを 100 回以上も繰り返すことで施主とのデザインの早期の合意を目指す[5]。通常は複数案を同時に、ある程度時間をかけて作るのに対して、1 つの案を高速に変化させていくということから「超線型プロセス」と呼ばれる。

ここでケース報告するモデルの変化記録は、10 数年の間に起こった揺さぶりと改訂の足跡である。数十回を超える改訂は超線型プロセスに似ている。改訂は、本モデルを適用する複数の案件で、それぞれの施主に向けて、あるときは並行的に、要求設計、試作、実装作業からのフィードバックの結果である。

### 3.1 モデル変化の量的側面

まず、記録が残っている直近 3 年間の概念モデルの変化数を追ってみる。

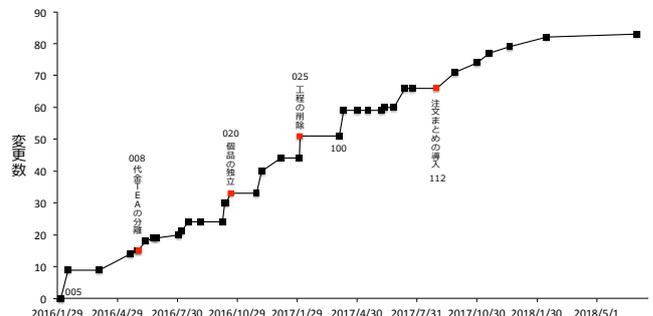


図 5 概念モデルの変更数の累積

は、2016 年 1 月から 2018 年 5 月までに概念モデルに加えられた 40 回以上の改訂における変更数の累積をプロットしたものである。ここでいう変更とは、モデル要素、すな

わち型、関連、主要な属性の追加・削除・修正を言い、これらのどれかが変われば1とカウントしている。変更の担当者は筆者である。

図中、赤色で示した点は、モデルの意味の大きな変更(以下で述べる)があった(とモデラが感じた)ことを示す。そこには改訂番号と主な変更内容を付記している。大きな変更だからと言ってモデル要素が大きく変わるわけではない。なお、改訂番号は025から100に飛んでいる。これは案件の切れ目を意味する。

### 3.1.1 改訂番号 005

案件開始直後に概念モデルは図3から図6のように変わっている。改訂番号005としているモデルである。このモデルの特徴は、「作業」「移動」「単位資源」を下位型によって細かく分類していることである。案件の初期段階では、基のモデルが想定していない要求に対して、現行モデルの構造を維持しようとして下位型を増やすことがある。

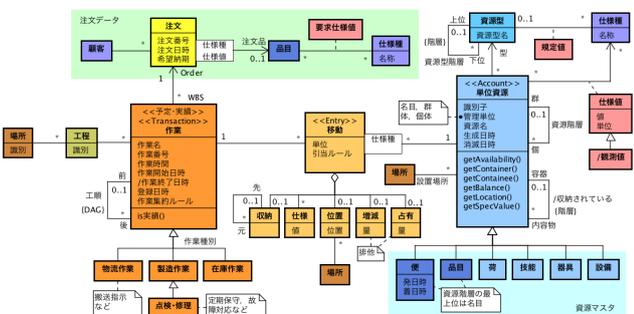


図6 改訂番号005の概念モデル全体

### 3.1.2 改訂番号 112

図7は終盤の改訂番号112としているモデルである。これまでの改訂を蓄積して、図6から大きく変わっている。

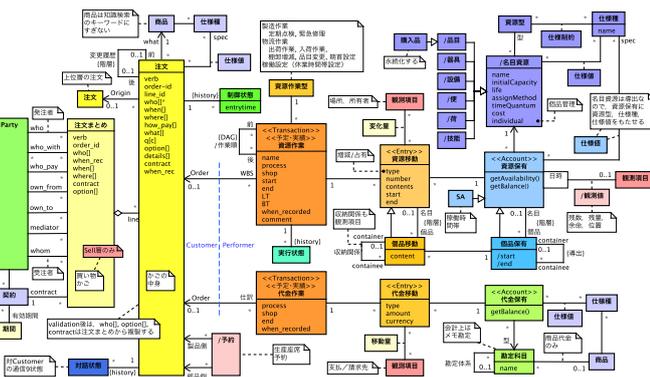


図7 改訂番号112の概念モデル全体

## 4. モデルのマイクロな変化

以下では、重要な変化のあった「注文」周辺と「資源」周辺に着目して、その足跡をもう少しマイクロに見てみる。

### 4.1 注文周辺の変化

「注文」のモデリングの開始点は改訂番号005の図8である。このモデルのCHARMとの違いは、「生産注文」と「顧客注文」が単なる「注文」に一般化されている点と、

注文品である「品目」が仕様でパラメタ化されている点である。

最初の「注文」の一般化は、顧客注文(販売注文)と生産注文(製造注文)を処理するドメインをそれぞれ分離したことによる[3]。2点目の品目のパラメタ化は、品目の多様性に対応する常套手段である(ただし、表記上は省略されている)。

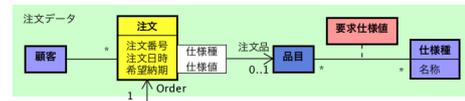


図8 改訂番号005における注文周辺のモデル

### 4.1.1 注文品とは何か

改訂番号006から011までの注文は「c注文」となっている。これはcustomer注文の意味である。上で顧客注文(本来は販売注文の意味)はドメインを分離したことにより区別をする必要がなくなった、つまり販売ドメインにおける注文と、製造ドメインにおける注文はそれぞれ別の意味を持つと述べたのに、なぜcustomer注文があるのか。これは、むしろドメイン分離によりドメイン間で注文を受け渡すことになり、ドメインの外部から受け取る注文(c注文)と外部へ送る注文(p注文)の区別が必要になったのが理由である。ただし、これらの区別は後に内部実装であることが分かってきて概念モデルからは消滅する。

この段階での論点は、注文品の変化である。006では「品目」が注文品として扱われている。注文数は明示されていないが、008では「注文数」という関連型として現れ、011ではそれが「注文品」に変わって「品目」の仕様と数、納期をもつようになる。

ほかに005との違いは、「基本契約」と「進捗」があるが大きな問題ではないので説明を省略する。

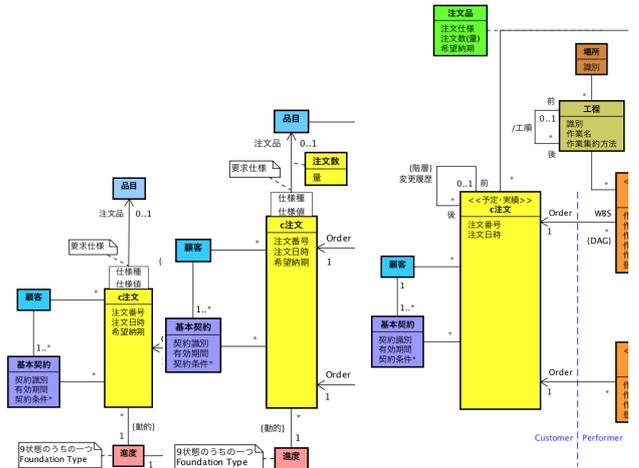


図9 改訂番号006, 008, 011における注文周辺のモデル

### 4.1.2 注文に予定と実績は必要か

改訂番号012から016までの変化の論点は「注文」の予実管理である。006の注文の「進捗」が012からは状態変

化の記録から導出するという議論から転換して、「注文」の追加データとして「注文予定」と「注文実績」をもとうとしている。これは100まで続き、最終的に「対話状態」として落ち着くまでには相当の時間を要した。

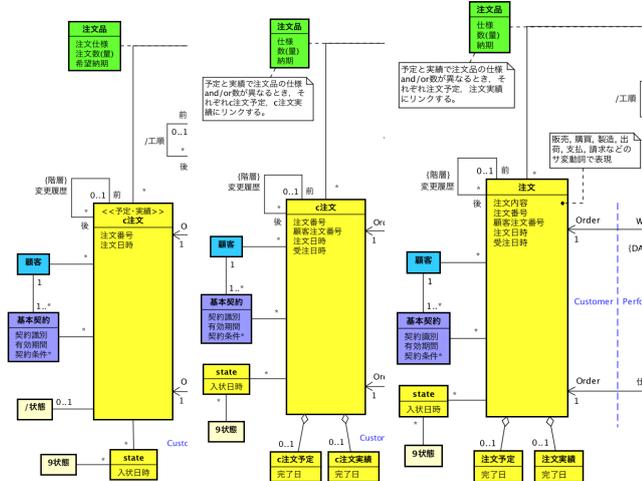


図10 改訂番号 012, 014, 016 における注文周辺のモデル

4.1.3 注文品は品目だけでない

022 では代金情報を「注文」の属性に入れてもらうという方針だったが、024 で「代金注文」として受け取るという方針が変わった。これによって従来の注文は「資源注文」として再定義された。これは大きな方針転換であったが、その意味については別の機会に報告する。

一方で、「注文品」が品目だけではないという気づきがあって、「要求資源」を、全体のモデル図では右上に存在する「品目」の上位型である「資源」に関連させたりした。

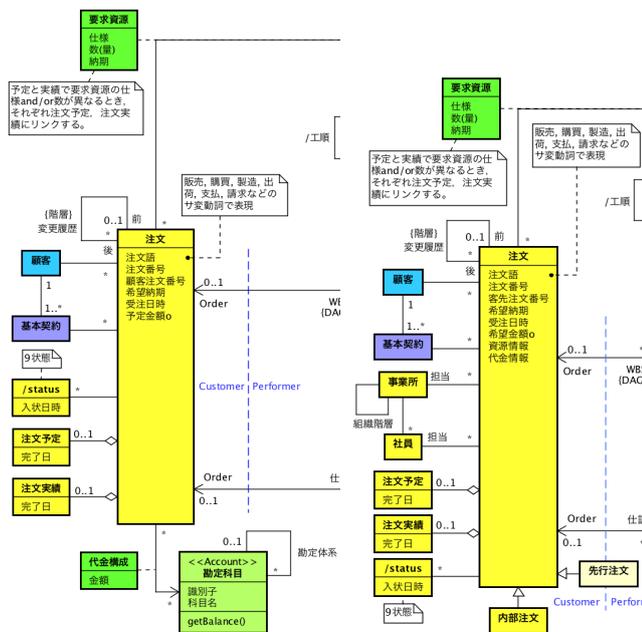


図11 改訂番号 022, 024 における注文周辺のモデル

4.1.4 商品は顧客が受けたいサービスのこと

改訂番号が100になって「商品」の意味が大きく変わる

た。製造業の案件では最終的に、商品とは発注者が受けたいサービス全体であると定義した。たとえば製造ドメインでは、製品Aを来週末までに100個入手したいとする注文の「商品」は、「生産して届けるサービス」であり、その仕様は「製品Aを来週末までに100個」で、商品の数量は常に「一式」となる。請求ドメインでは、「請求」という商品で、請求額、請求方法を仕様とする商品と言える。この概念はいまのところは(改訂番号118まで)うまく行っている。この段階では、「商品」と「資源」の関係が曖昧なままである。

それ以外の変化は、発注者(who)として「Party」を設け、その内部の「組織」を識別するようにした点である。このPartyは法人に相当するという堅い定義をしていた。「Party」の定義は次のタイミングで大きく変わる。

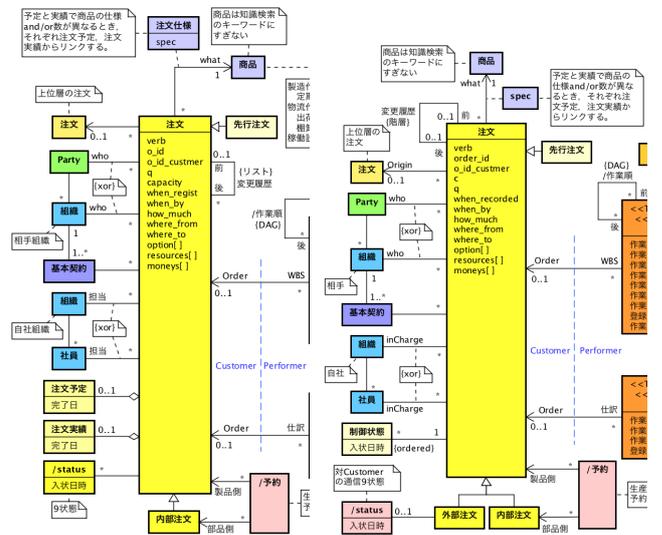


図12 改訂番号 100, 103 における注文周辺のモデル

4.1.5 注文に関わる Party

注文には多くのロールでPartyが関わる。発注者、支払者、送付先、所有者、仲介者などである。FowlerのPartyパターン[4]によると、Partyはビジネスプロセスに関わる責任ある主体である。責任はParty内の権限に関わるので、階層構造をもつ。よって、注文に関わるPartyは階層構造を意味的に内包している。

Partyを図13のように複合的な型として定義することによってPartyの「注文」との多くの関連を整理する。もちろんその実装の裏付けは必須である。こうして109では、注文に関わる可能性のある「Party」を整理した。

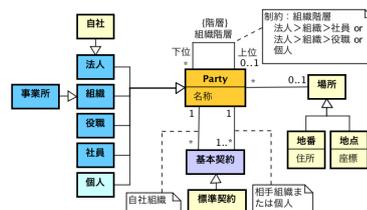


図13 Partyの再定義

112では、あらたに「注文まとめ」を導入した。生産注文を扱う分には注文まとめは必須ではなかったが、販売注文を扱う場合は、代金精算の単位となり必須となった。上位ドメインに対する「まとめ」の役割は、ドメイン内部に対しては指示の「ばらし」と結果の「まとめ」にある。単体の注文は、内部処理の進捗管理が必要で「制御状態」を設けた。「対話状態」は外部ドメインとの対話の進捗管理が目的である。109では導出としたが、これは誤りで、112では対話イベントの履歴の記録とした。「まとめ」はこうした制御状態と対話状態を調整する。

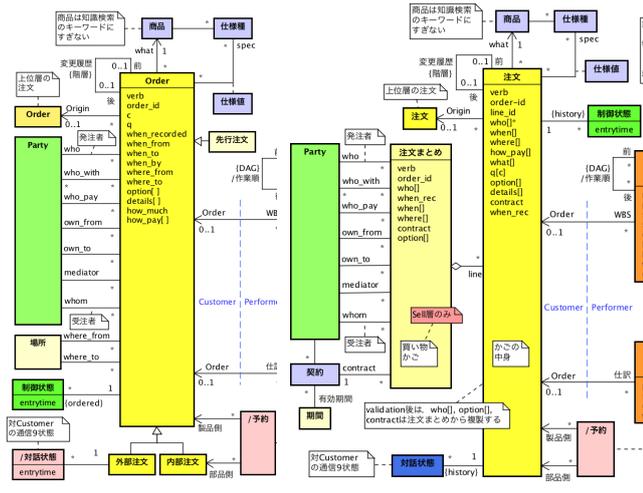


図14 改訂番号109, 112における注文周辺のモデル

4.1.6 注文まとめに迷う

114では「Project」があらたに導入された。これは「注文まとめ」に対して発生する取消や変更を束ねる概念である。最初、Partyの関わりをProjectにもたせたが、発注単位で変わることから、116では戻した。

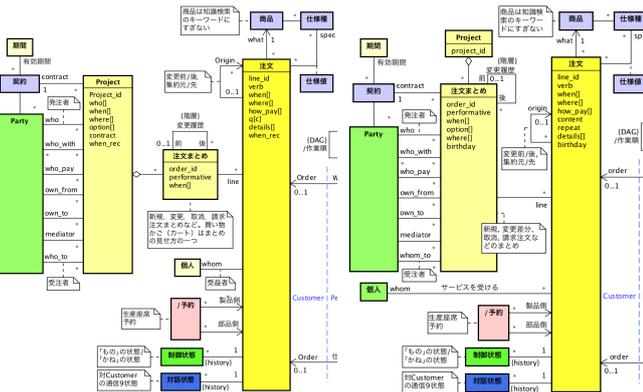


図15 改訂番号114, 116における注文周辺のモデル

4.2 注文の変化が意味すること

注文に関わるモデルの変遷を見て、論点は大きく、①注文品とは何か、②注文の単位は何かの2点である。

1点目については、注文とは、もともと誰が何をどれだけほしいかを述べるものと考えていたが、その形式や内容が業種業態によって大きく異なる。その理由は、「注文品」は「何をして欲しいか」、つまりもともとサービスを含んで

おり、サービス内容は多様だったからと考えられる。

2点目の「注文」の単位の問題は1点目と密接に関わっている。単体の注文は一つの注文品を表現するので、これが基本的な単位である。しかし、実際の注文の発行、取消、変更は、単体の注文をいくつかまとめた「注文まとめ」になり、代金の請求や取消・変更に伴う違約金の発生もまとめ単位になる。さらに、注文まとめ内の注文の組合せによって割引や割増、前提などのルールが存在する。これらの概念についてはこれからも変動するだろう。

4.3 資源周辺の変化

次に「資源」の周辺の変化について見ていく。

4.3.1 保有はインスタンス

006の005に対する違いは、「単位資源」の下位型であった「便」「品目」などを「資源型」との間に「資源」を設けて、その下位型とした点である。このとき「単位資源」を「資源保有」とした。本来、単位資源は現物(インスタンス)であり、資源マスタ(知識レベル)ではない。

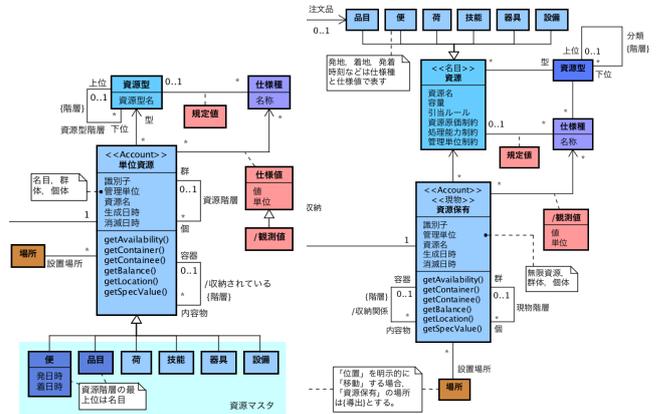


図16 改訂番号005, 006における資源周辺のモデル

4.3.2 個品保有の出現

008で資源型から仕様種への関連を削除して、020で戻している。観測パタン[4]は020の構造のほうが妥当である。020では「個品保有」が導入された。個品は008以前では「資源保有」の階層として扱っていたものが、実装検討を経て、下位型として明示したものである。「資源保有」にある観測値間の親子関連は、群体と個体間の関係であった。

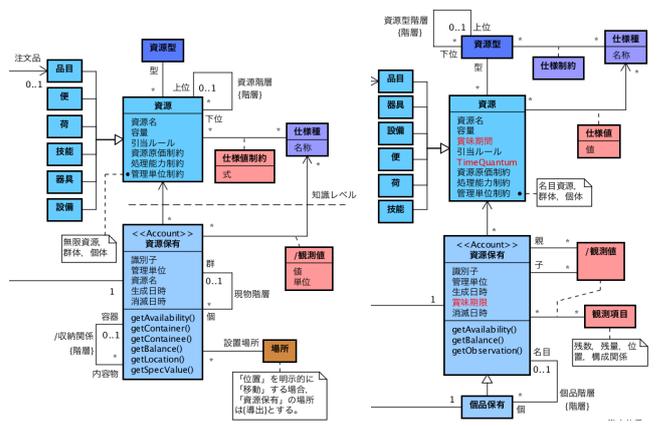


図 17 改訂番号 008, 020 における資源周辺のモデル

### 4.3.3 資源を永続化するか

これまで、「資源」は「資源型」のみをもって、仕様値が異なるものが発生するたびに「資源保有」を導出することとしていた。それは、製造業においてもともと多様な仕様値の組合せをもちうる製品を、あらかじめすべての仕様を展開して、製品マスタのように作っておくことを避けたためであった。したがって、実装は「資源」は永続化されなくてよい。

ところが購入品については、仕様ごとに、購入先、標準価格などの値を前もって作っておいて、原価シミュレーションを行いたい。そこで、「購入品」のみを購入品マスタのようにあらかじめ永続化しておく方式をとった。モデル上は、109 において「購入品」は仕様値をもち、その他の「名目資源」は「資源保有」で仕様値をもつように記述した。

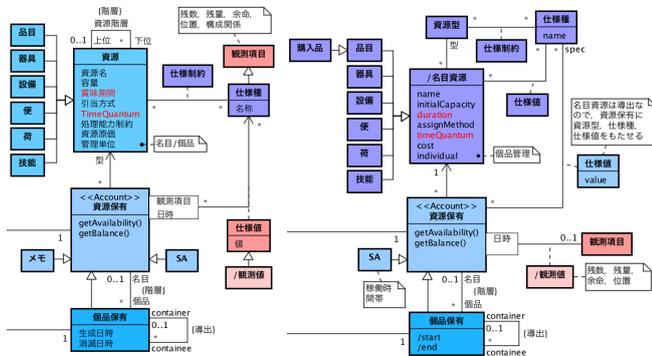


図 18 改訂番号 103, 109 における資源周辺のモデル

### 4.4 資源の変化が意味すること

資源に関わるモデルの変遷を見て、論点は、大きく①多様な仕様の組合せをもつ品目をどう扱うか、②仕様の組合せごとに設定される属性値をどう扱うかの2点だったと思う。

1 点目については、資源を仕様種と仕様値によりパラメタ化して、新しく仕様値が与えられるたびに資源保有のインスタンスを生成するという方針であった。しかし、あらかじめ属性値を保存しておきたい場合は、資源を永続化せざるを得ないという矛盾が発生する。結果として例外を作ってしまった。この揺さぶりによって、資源型、資源、資源保有の構造が明確になったが、例外をどう扱うかについては今後の事例の蓄積を待つことにする。

## 5. おわりに

モデルの変化の足跡は、実際、モデルの迷い道であった。しかし、このような迷い体験を意識しているモデルは少ないのではないかと。システム開発の案件ごとにモデルを作るのではなく、一般モデルを活用することで、すばやく、健全で、拡張性に富む実装が導かれる。このようなモデルの実践が報告され、共有されることを強く望む。

## 参考文献

- [1] 児玉公信, 水野忠則, 「少量多品種型生産管理システムの一般モデル CHARM の提案」, 情報処理学会論文誌, Vol. 49, No.2, pp.902- 909, 2008.
- [2] 児玉公信, 「計画・実行システムの一般モデル—生産管理システムと金融業務システムの共通性—」, 情報処理学会研究会報告, IS109-4, 2009/9/14.
- [3] 児玉公信, 「取引処理のための基幹情報システムの状態側面から見た一般モデル」, 情報処理学会研究会報告, IS141-4, 2017/8/25.
- [4] Fowler, M., "Analysis Patterns: Reusable Object Models," Addison-Wesley, 1997. 邦訳) 児玉公信ほか訳, 「アナリシスパターン」, ピアソンエデュケーション, 1998.
- [5] 藤村龍至, 「プロトタイプング—模型とつづやき」, LIXIL 出版 (2014)