

モデリングの心理学とシステム思考

2008年7月14日
児玉公信

モデルの意味

認知と世界
システム論

意味記憶 (semantic memory)

- Collins and Quillian (1966)

– 概念の格納

- is-a
- has-a

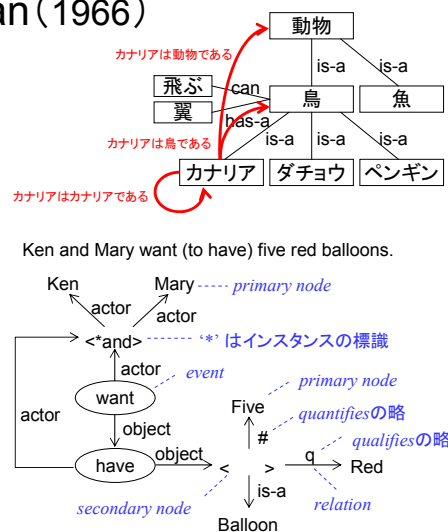
- LNR (1972)

– スロットと関連

- 属性
- 関連

- Tulving (1972)

– エピソード記憶と意味記憶



スクリプト

- Schank and Abelson (1979)

– シナリオとアクタ

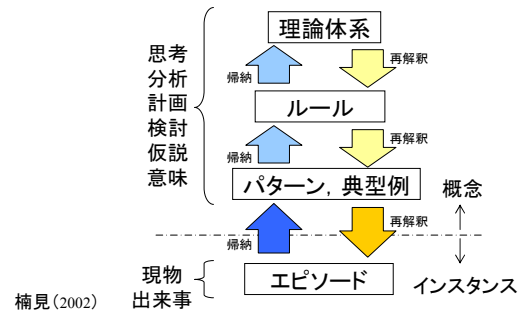
タイトル: レストラン	ロール: 客
トラック: コーヒーショップ系	給仕
小道具: テーブル	料理人
メニュー	料定係
料理 (F)	店の経営者
請求書	
金 (かね)	

初期状態:	客は空腹である結果:	客の金が減る
客は金を持っている	主人の金が増える	客は喜ぶ(かもしれない)

(a) スクリプトヘッダー

メンタルモデル

- Johnson-Laird (1983)
 - 表象と操作
 - 認知地図
 - ・ 実世界と認知のギャップ

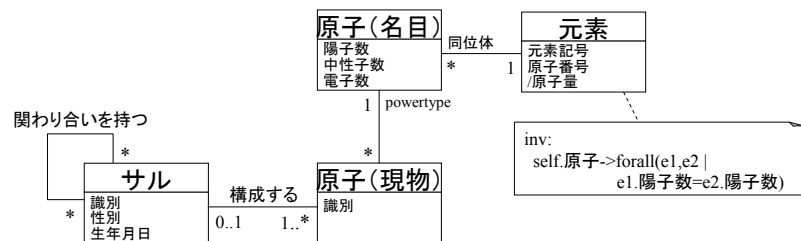


全体論

- システム
 - 分解・還元主義に対するアンチ
 - 複雑性をどう扱うか
 - 対象をシステムとして見る態度
 - ・ 全体があり, 構成要素がある
 - ・ 創発: 全体の振る舞い > 総和 (個々の振る舞い)
 - ・ 自己組織化
 - ・ 階層: システムはより大きなシステムの一部
- アブダクション
 - ・ 観測者, 操作者
 - ・ 入力, ブラックボックス, 出力
 - ・ 概念モデル, ダイナミズム

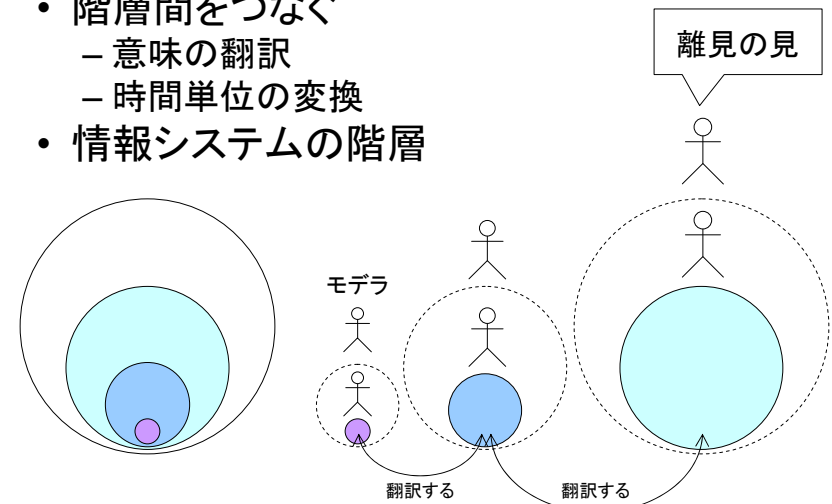
意味論

- システムの階層性
 - サルの行動を原子の振る舞いで説明 (モデル化) できるか
 - モデルは階層内の概念とそのメタでしか書けない
 - ・ 意味は階層内の文脈に依存し, 階層を越えられない
 - ・ 観測者は誰?



階層間の翻訳

- 階層間をつなぐ
 - 意味の翻訳
 - 時間単位の変換
- 情報システムの階層

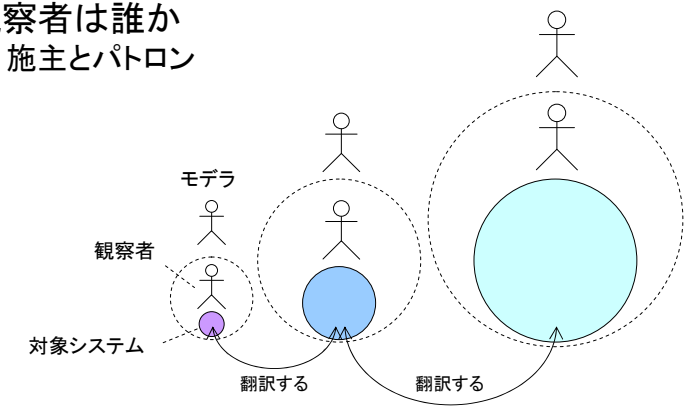


モデリング技術上の悩み

システム階層の歩き方
 正規形
 機能モデリング

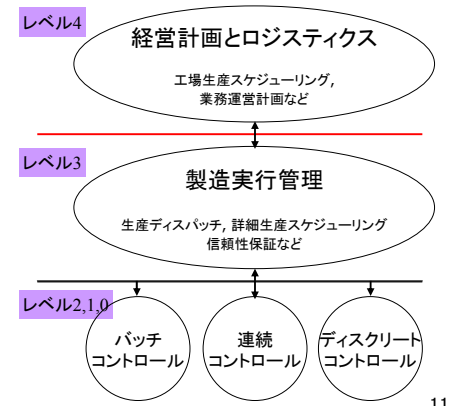
階層の中での観察者の決定

- 重層するシステムから階層を切り出す
 - 便宜的な階層と本質的な階層
 - 観察者は誰か
 - 施主とパトロン



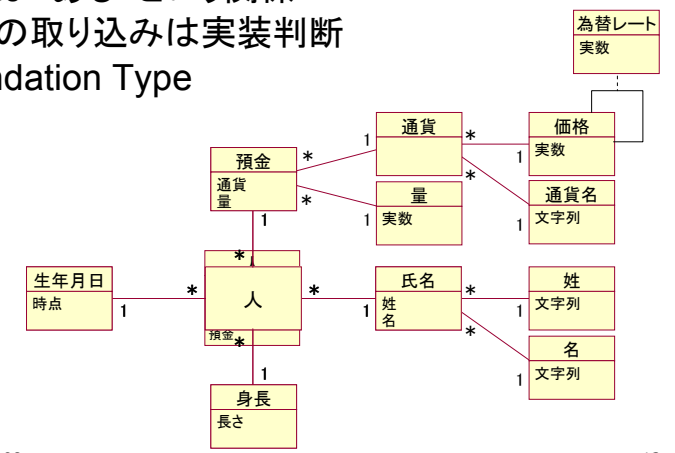
生産管理システムの例

- IEC 62264-3:経営と制御の統合
 - 情報システムとリアルワールドとの接点
 - OrderとCommand



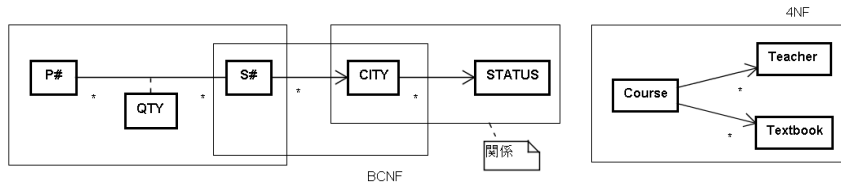
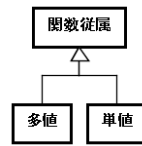
属性の外出し

- すべての属性は外に出せる
 - 本質は<ある>という関係
 - 属性の取り込みは実装判断
 - Foundation Type



正規化理論を乗り越える

- 関数従属
 - 一方向の関連
- 正規形
 - BCNFまで
 - 単値従属だけで関係を切り出す
 - 4NF
 - 多値従属も含めて関係を切り出す



ユースケースの意味

- 仕事とユースケース
- 仕事とは
 - 「行わなければならないこと」を「体や頭を使って」行うこと
- 仕事の対象と仕事のシステム
 - 「もの」-ものづくり(システム)
 - 「人」-サービス(システム)
 - 「情報」-情報システム
- 行わなければならないこと
 - 対象の「はじめの状態」を「終わりの状態」に変える
 - 考察の範囲: Universe of Discourse (UoD)
 - システムロード

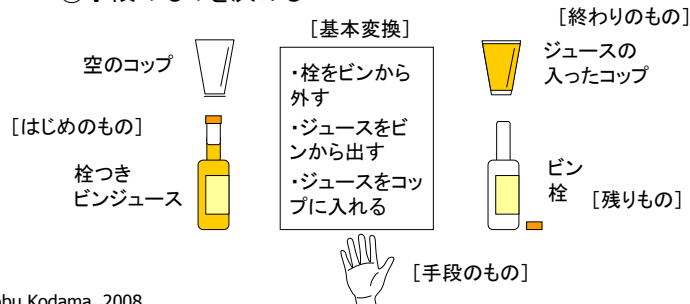
人の仕事

Dimension
を合わせる

中村善太郎:「シンプルな仕事の構想法」, 日刊工業新聞, 1992

もの・こと分析

- もの・こと図
 - はじめの「もの」を終わりの「もの」に変換する
 - ① 終わりの「もの」を決める
 - ② はじめの「もの」を決める
 - ③ 残りものを決める
 - ④ 基本変換を明確化する
 - ⑤ 手段のものを決める

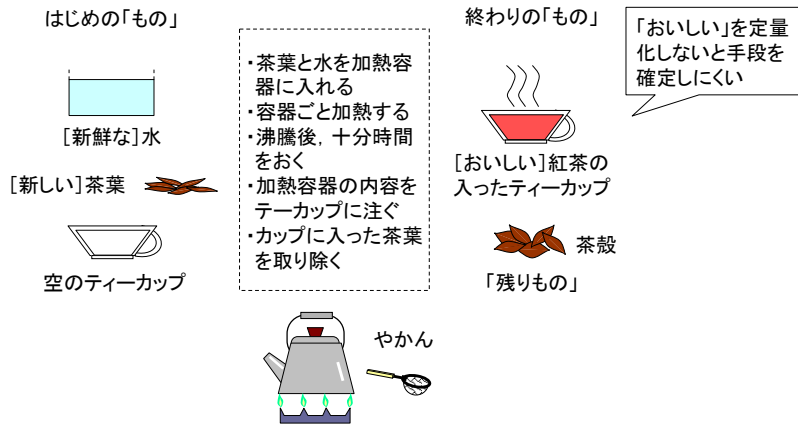


かなめ

- 要の「もの」, 要の「こと」
 - 真に得たいもの, それに適うもの
 - 目的でない余計な負荷を除去した結果
 - 終わりの「もの」の品質: 品質要件
- 要の変化
 - 必要最小限の対象の変化
 - 良品要件: すべてが解明されているわけではない
- 要の手段
 - ワークヘッド: 対象に接触して働く「もの」

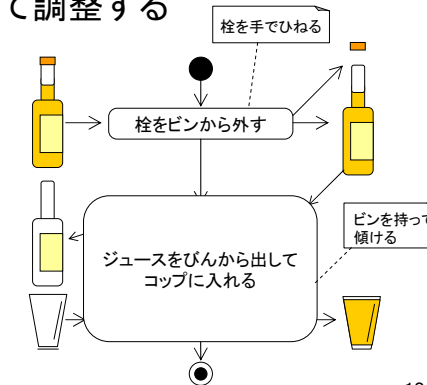
揺さぶりの
極致

最小かつ
完備



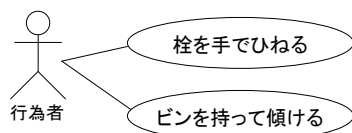
仕事の流れ

- 基本変換を業務フローとして記述する
 - 手順を具体化する
 - オブジェクトフローを追記
 - 手順を手段にあわせて調整する
 - 行為者を決める



仕事があつての機能

- 手段のものを機能として記述する
 - その機能を情報システムに割り当てたものをユースケースという
- 機能のテストとしてのシナリオ
 - ユースケースのインスタンス
 - 仕事のシミュレーション(ウォースルー)



ユースケース記述

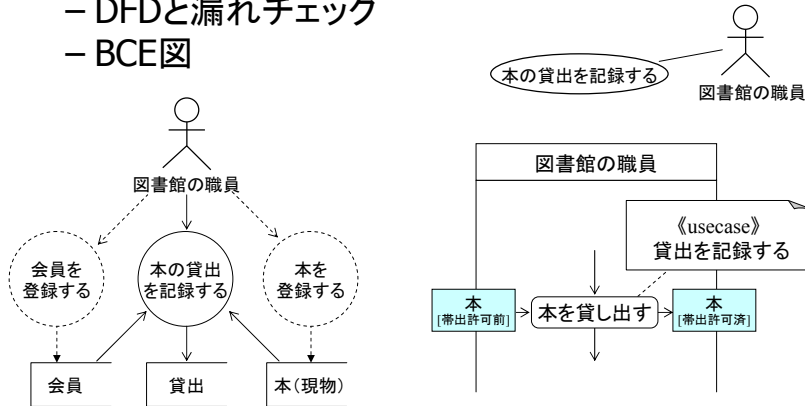
- 記述例



ユースケース名: 栓を手でひねる
 行為者: ジュースを飲みたい人
 目的: 栓をビンから外したい。
 事前条件: 栓が外れていない。
 事後条件: 栓が外れている。
 基本系列: ①行為者がこのユースケースを起動する。
 ②手はビンの口がどこにあるか行為者に聞く。
 ③行為者はその位置を提示する。
 ④手は栓を握って、外れるまで左回りにひねる。
 代替系列: A. 基本系列④で、栓が回くれひねることができない場合、...
 B. 基本系列④で、ひねってもひねっても栓が抜けない場合、...
 備考: ①栓が王冠またはコルクの場合は、別のユースケースを使う。
 シナリオ:
 ①7月7日の夕方、岡田史子は星を眺めながら、先週長野への出張の折りに購入したリンゴジュース「本生」を開けようと思った。自分の手で栓をひねろうとしたが、栓が固かったので、となりにいた青柳祐介の手を使って栓を外した。

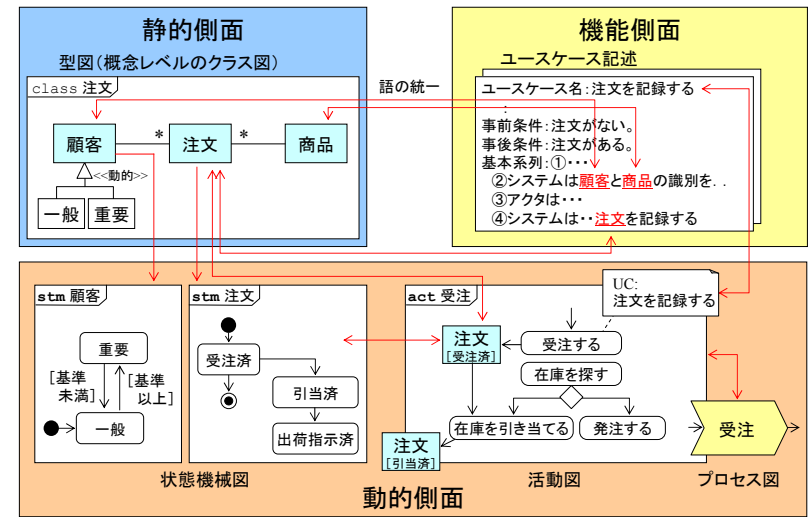
DFDの復権

- 行為と機能を分離したら、機能と型の関係が見えなくなった
 - DFDと漏れチェック
 - BCE図



(c) Kiminobu Kodama, 2008

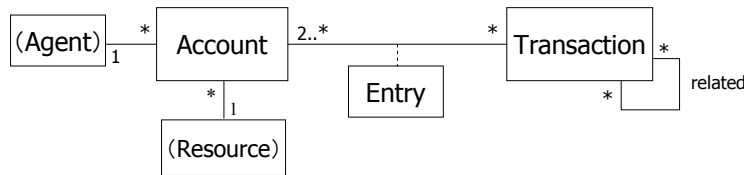
モデル間の整合



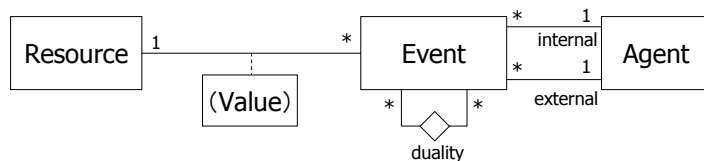
(c) Kiminobu Kodama, 2008

時間があつたら

- 関連クラスの有効性



- REA (McCarthy, 1982) との対比



(c) Kiminobu Kodama, 2008

まとめ

- モデリングを考える前提
 - 認知の働き
 - システム思考
- モデリング上の悩み
 - 世界の切り出し
 - 属性の外出しと取り込み
 - 行為者の仕事とユースケース
 - DFD

(c) Kiminobu Kodama, 2008