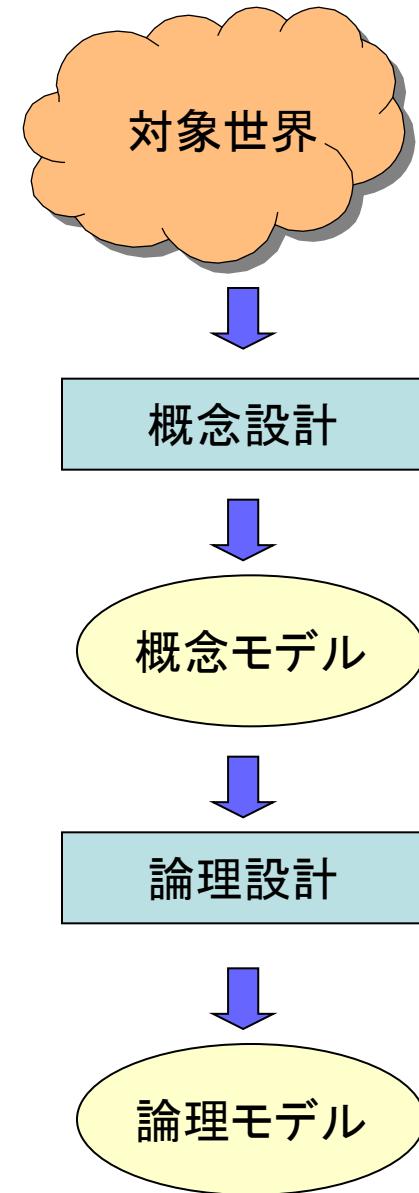


データベース 【6:リレーションナルデータベース 設計論(1)】

石川 佳治

実世界のデータモデリング(復習)

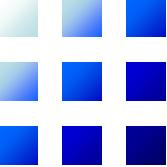
- **概念設計** (conceptual design)
 - DBMSとは独立に、対象世界を論理的に記述
 - 概念モデルを利用
 - 実体関連モデルなど
- **論理設計** (logical design)
 - DBMSが提供するデータモデルへと変換
 - 論理モデルを利用
 - リレーショナルデータモデルなど





リレーションナルデータベース設計の流れ

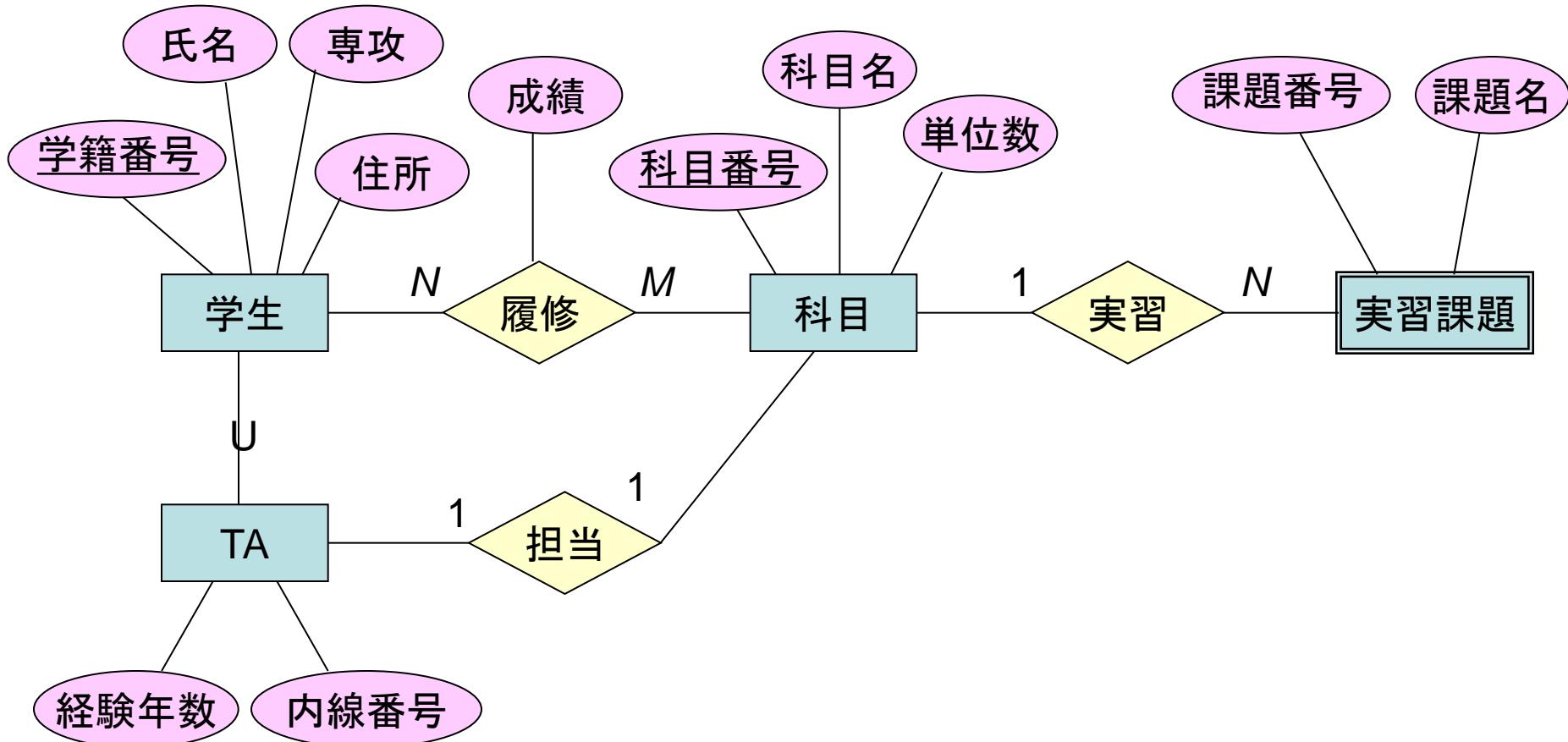
1. 実体関連モデルを用いた概念設計
 - すでに説明
2. 実体関連モデルからのリレーションナルデータベーススキーマの導出
 - 4.1節
3. リレーションナルデータベースの**正規化**
 - よりよいスキーマに洗練
 - 4.2節以降のトピック
4. さらに物理設計を行うこともある



実体関連モデルからの リレーションナルデータベース スキーマの導出



実体関連図の例(図2.8)





導出法(1): 実体集合の扱い(1)

A) 通常実体集合

- 通常実体集合 E について、リレーション R を定義
- E のすべての属性を R の属性とする
- E の主キーを R の主キーとする

B) 弱実体集合

- 実体集合 E' を識別上のオーナとする弱実体集合 E について、リレーション R を定義
- E のすべての属性を R の属性とする
- E' に対応するリレーションの主キーを R の属性に追加
- 追加した属性と E の部分キーの組合せを R の主キーとする



導出法(2) : 実体集合の扱い(2)

C) 汎化階層

- 実体集合 E に対応する汎化階層があり, E の上位の実体集合 E' が存在する場合
- E' に対応するリレーションの主キーを E に対応するリレーション R の属性に加えて R の主キーとする



導出法(3)：関連集合の扱い(1)

A) 次数2の関連集合

① R_1, R_2 が1対1対応の場合

- R_1, R_2 のいずれかにもう一方のリレーションの主キーおよび(もしあれば)関連集合自身の属性を付加

② R_1, R_2 が1対N対応の場合

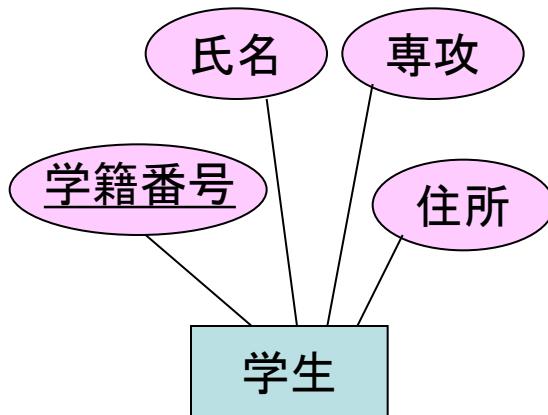
- R_2 に R_1 の主キーおよび(もしあれば)関連集合自身の属性を付加

③ R_1, R_2 がN対M対応の場合

- R_1 の主キー、 R_2 の主キー、および(もしあれば)関連集合自身の属性からなる新たなリレーションを定義
- 通常、 R_1 の主キーと R_2 の主キーの組合せをこのリレーションの主キーとする

B) 次数3以上の関連集合：③に準じてリレーションを定義

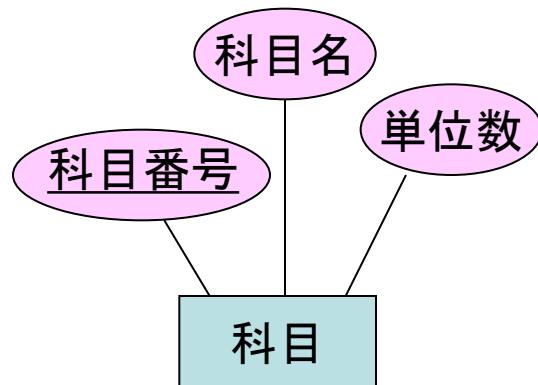
導出の例(1): 通常実体集合



通常実体集合の
ルールを適用

学生(学籍番号, 氏名, 専攻, 住所)

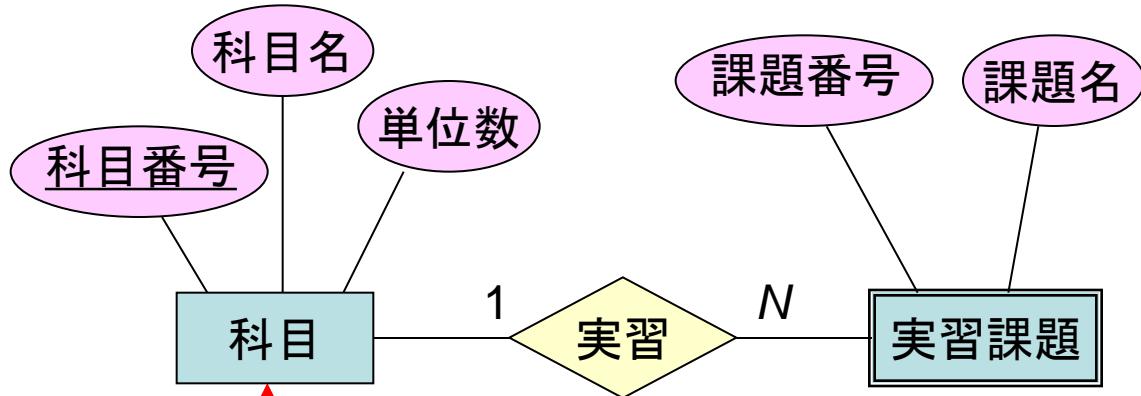
- ・リレーション「学生」を定義
- ・実体集合「学生」のすべての属性をリレーション「学生」の属性とする
- ・実体集合「学生」の主キーをリレーション「学生」の主キーとする



通常実体集合の
ルールを適用

科目(科目番号, 科目名, 単位数)

導出の例(2)：弱実体集合



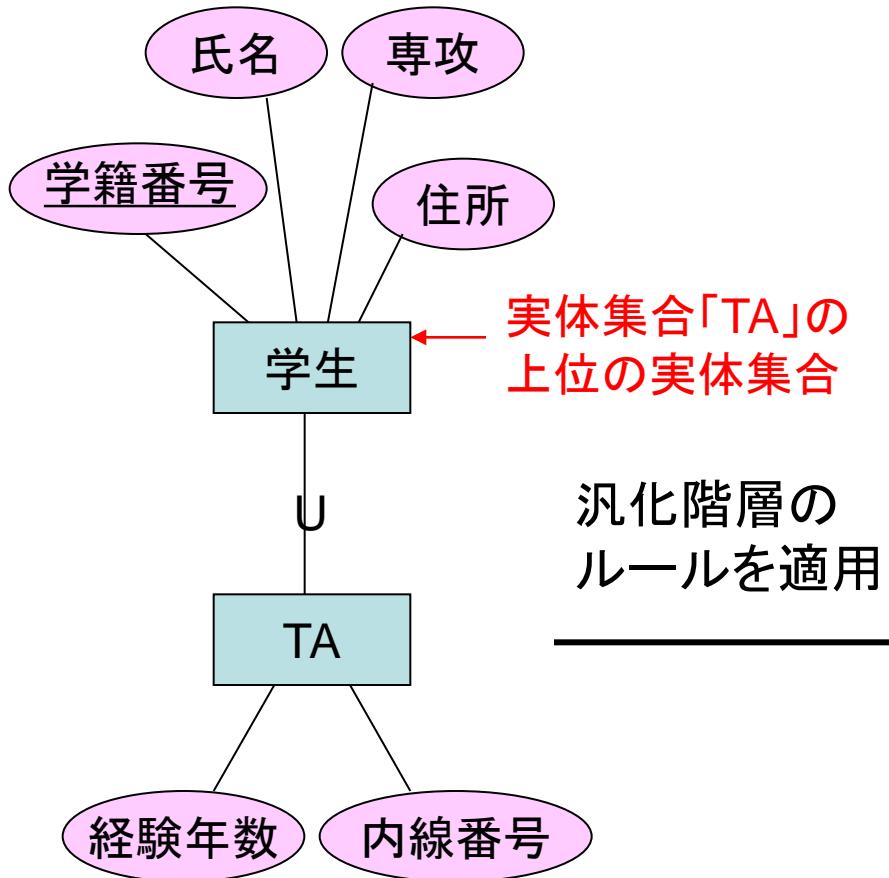
実体集合「実習課題」の
識別上のオーナ

弱実体集合の
ルールを適用

実習課題(科目番号, 課題番号, 課題名)

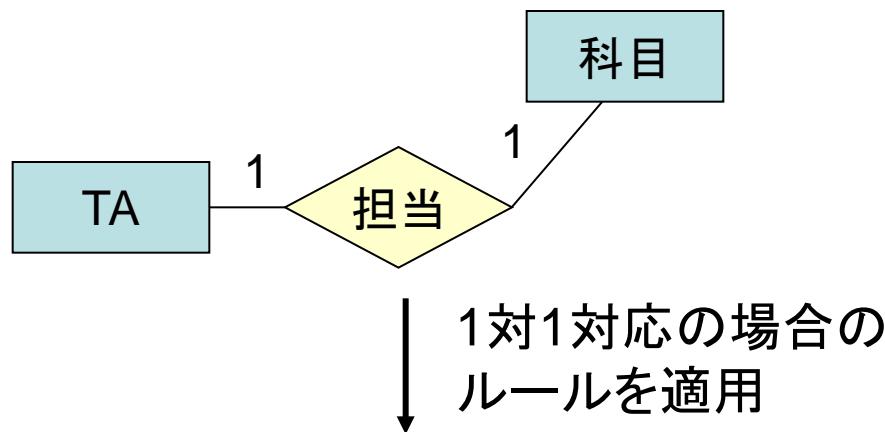
- ・リレーション「実習課題」を定義
- ・実体集合「実習課題」のすべての属性をリレーション「実習課題」の属性とする
- ・識別上のオーナである実体集合「科目」に対応するリレーション「科目」の主キーをリレーション「実習課題」の属性に追加
- ・「科目」から得た属性と「実習課題」の部分キーの組合せを「実習課題」の主キーとする

導出の例(3)：汎化階層



- ・リレーション「TA」を定義
- ・実体集合「TA」のすべての属性をリレーション「TA」の属性とする
- ・上位の実体集合「学生」に対応するリレーション「学生」の主キーをリレーション「TA」の属性に追加
- ・追加した属性を主キーとする

導出の例(4): 1対1の関連集合



1対1対応の場合の
ルールを適用

TA(学籍番号, 経験年数, 内線番号, **科目番号**)

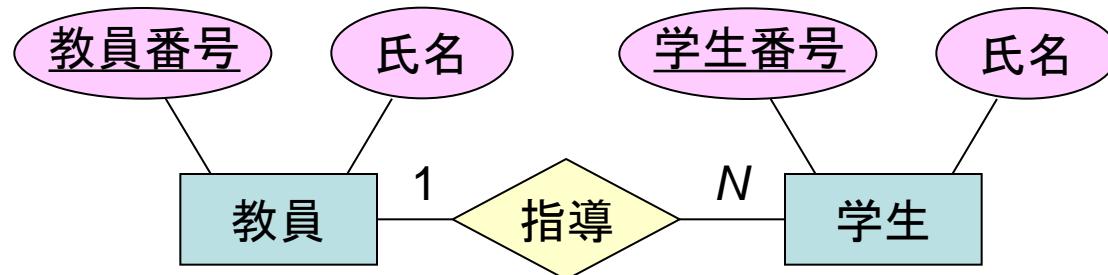
- ・リレーション「TA」あるいは「科目」の
いずれかにもう一方の主キーを追加
- ・(もしあれば)関連集合自身の属性を追加
←この場合はなし

TAがいらない科目を
許す場合には、こちらの
設計の方が適切
(2.4節の「参加制約」に
関連)

※ この場合、リレーション「TA」の主キーを「科目」に追加
するという選択も可

科目(科目番号, 科目名, 単位数, **学籍番号**)

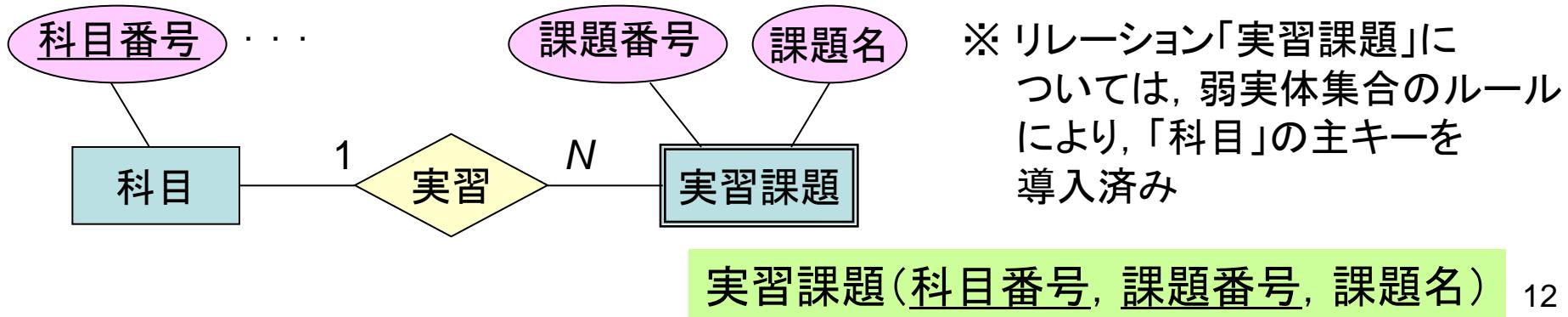
導出の例(5)：1対Nの関連集合



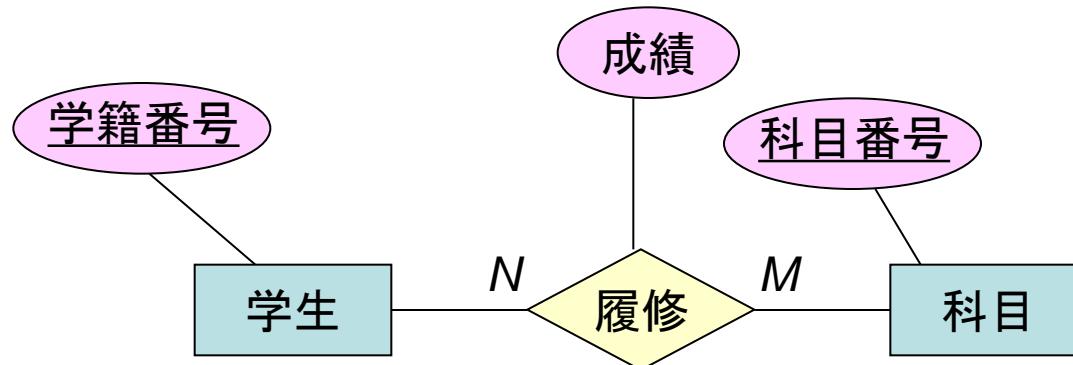
↓ 1対N対応の場合の
ルールを適用

教員(教員番号, 氏名)
学生(学生番号, 氏名, **教員番号**)

- ・Nの側のリレーション「学生」に1の側の
リレーション「教員」の主キーを追加



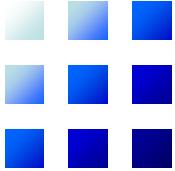
導出の例(6): N対Mの関連集合



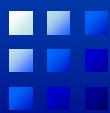
↓
N対M対応の場合の
ルールを適用

履修(科目番号, 学籍番号, 成績)

- ・リレーション「学生」, 「科目」のそれぞれの主キー, および関連集合自身の属性「成績」による新たなリレーション「履修」を生成
- ・リレーション「学生」, 「科目」の主キーを組み合わせて「履修」の主キーとする



好ましくないリレーションスキーマ



営業リレーションの例

営業

商品番号	顧客番号	社員番号	販売価格
$i1$	$c1$	$s1$	100
$i1$	$c2$	$s2$	120
$i2$	$c1$	$s1$	200
$i2$	$c2$	$s2$	210
$i3$	$c1$	$s1$	250
$i3$	$c2$	$s2$	250
$i4$	$c1$	$s1$	150

ある顧客を担当する営業マンは一人という制約がある



更新不整合(update anomaly)

A) 修正不整合

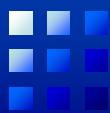
- 顧客 c が購入する品目ごとに担当営業マン s の社員番号が格納されるので冗長
- 顧客 c の担当営業マンが変更されるときは、 c が購入する商品の数だけ変更が必要

B)挿入不整合

- 顧客 c の担当を s と決めても、取引商品が決まっていないと「営業」リレーションに登録不可
- 主キーの一部である「商品番号」属性がNULLであり、キー制約に反する

C) 削除不整合

- 営業マン s が担当する顧客 c が唯一購入していた商品を取り扱い中止にしたとき、該当タプルを削除すると c の担当営業マンの情報が失われる



リレーションの分解

- 問題が生じる原因
 - 顧客と担当営業マンの関係が商品の販売情報にまぎれて表現されている
- 解決策: **リレーションの分解**
 - 販売(商品番号, 顧客番号,
販売価格)
 - 営業担当(顧客番号, 社員番号)
- 無損失結合分解
 - 自然結合をとることで, もとの「営業」リレーションの情報が復元可能

販売

商品番号	顧客番号	販売価格
$i1$	$c1$	100
$i1$	$c2$	120
$i2$	$c1$	200
$i2$	$c2$	210
$i3$	$c1$	250
$i3$	$c2$	250
$i4$	$c1$	150

営業担当

顧客番号	社員番号
$c1$	$s1$
$c2$	$s2$

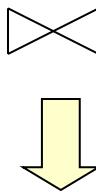


無損失結合分解でない分解

営業1	商品番号	顧客番号
	$i1$	$c1$
	$i1$	$c2$
	$i2$	$c1$
	$i2$	$c2$
	$i3$	$c1$
	$i3$	$c2$
	$i4$	$c1$

営業2	商品番号	社員番号	販売価格
	$i1$	$s1$	100
	$i1$	$s2$	120
	$i2$	$s1$	200
	$i2$	$s2$	210
	$i3$	$s1$	250
	$i3$	$s2$	250
	$i4$	$s1$	150

自然結合



	商品番号	顧客番号	社員番号	販売価格
	$i1$	$c1$	$s1$	100
	$i1$	$c1$	$s2$	120
	$i1$	$c2$	$s1$	100
	$i1$	$c2$	$s2$	120

元に
戻らない!

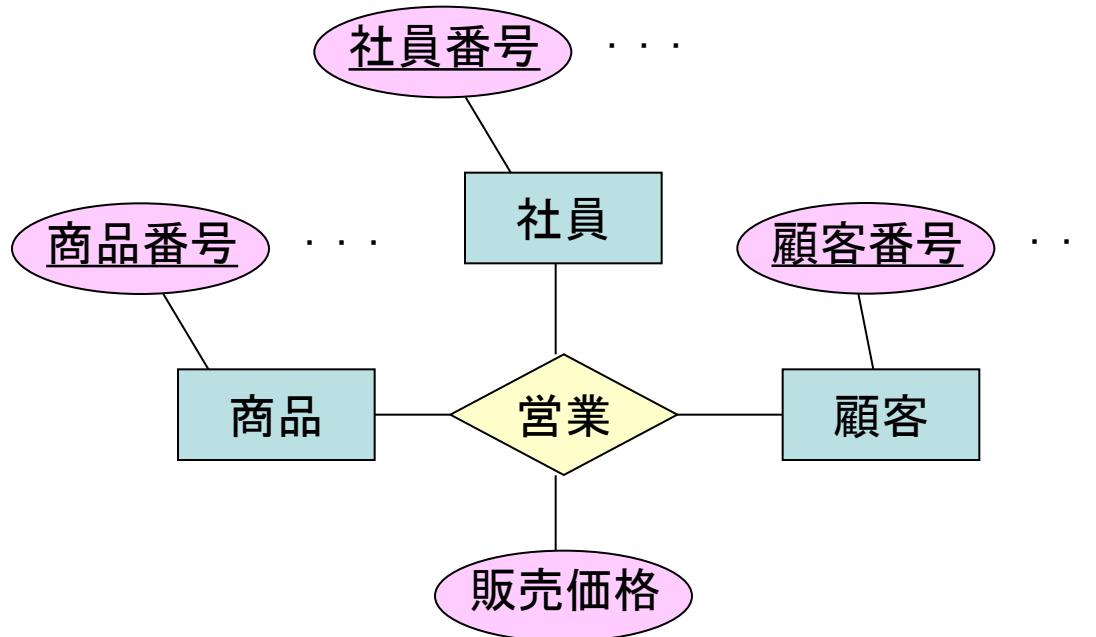


今後の展開

- 適切なスキーマ構造を得るために分解のアプローチ
 - 従属性の概念の導入
 - 正規形の概念の定義
 - 分解により正規形を得るための手法



余談: 次数3以上の関連集合



- ・商品と社員は N 対 M 対応
- ・商品と顧客も N 対 M 対応
- ・社員と顧客は1対 N 対応

- 実体関連モデルにおける関連集合での表記の問題
 - うまく対応関係を表記できない ← **実体関連モデルの限界**
- 現実的な対応策
 - まず4.1節の手続きでリレーション「営業」を導出
 - その後の正規化のステップでよりよいスキーマに分解