データベース論

朝日大学大学院
経営学研究科
奥山　徹
okuyama@alice.asahi-u.ac.jp

講義日程

- 4月17日　ガイダンスおよび集合論の基礎
- 4月24日　リレーショナルデータベースの基礎
- 5月01日　データ操作言語
- 5月08日　データベースの論理設計
- 5月15日　SQL（データベース操作言語）の基礎
- 5月22日　データベース管理システム
- 5月29日　データベースの内部スキーマ
- 6月05日　質問処理とその最適化
- 6月12日　トランザクション処理
- 6月16日　分散データベース序説
- 6月26日　定期試験

本日のお品書き

- 第一正規化（再掲）
- リレーションの高次正規形
- リレーショナルデータベースの設計

リレーションの正規性

- *R*をドメイン$D_1, D_2, \ldots, D_n$上のリレーションとしたとき、もしすべての<u>ドメインが単純</u>なら、*R*は<u>**第一正規形(1st normal form, 1NF)**</u>という
- 単純なドメイン

(1) ドメインDは他のいくつかのドメインの直積（の部分集合）ではなく、かつ

(2) Dはあるドメインのベキ集合（の部分集合）でもない

単純でないドメインの例

- 地図上の位置：位置＝{(x,y)| xは緯度、yは経度}
  →(1)に抵触
- 社員の扶養家族：一般には妻や子どもなど複数人となり、扶養家族ドメインは人名ドメインのベキ集合となる
  →(2)に抵触

都市

都市名	位置	人口
A	(X_A,Y_A)	P_A
B	(X_B,Y_B)	P_B
C	(X_C,Y_C)	P_C

扶養

社員番号	社員名	扶養家族
001	A	{H,I,J}
002	B	ϕ
003	C	{K}

正規化(normalization)

- 非第一正規形のリレーションを第一正規形にすること
- ドメインが他のドメインの直積の場合→直積を分解する
- ベキ集合の場合→タップルに分解する
- 以上の操作を、すべてのドメインが単純になるまで繰り返す

都市

都市名	緯度	経度	人口
A	X_A	Y_A	P_A
B	X_B	Y_B	P_B
C	X_C	Y_C	P_C

扶養

社員番号	社員名	扶養家族
001	A	H
001	A	I
001	A	J
002	B	-
003	C	k

第一正規形の意義

- すべてのデータを１枚のリレーションで表現できわかりやすい
- データ操作言語が簡明に定義できる
- 非正規リレーションを外部スキーマとして表現する道が残されている
- ANSI/X3/SPARCのデータベースの三層スキーマ構造と整合性がよい
- 高次正規化により、一次正規化による冗長性は解消できる

注文

顧客名	商品名	数量	単価	金額
A商店	テレビ	3	198,000	594,000
Bマート	テレビ	10	198,000	1,980,000
Bマート	洗濯機	5	59,800	299,000
C社	餅つき機	1	29,800	29,800

属性名のアンダーラインはキー属性を表す

図 4.1 リレーション 注文

リレーションの高次正規化

- 1972年のCoddの論文では第二および第三正規形について論じている
- 高次正規化の目的
  - 更新時異常の発生抑制
  - データの部品化
  - より高い平明性
  - 応用独立性

リレーショナルデータベースの前提条件

- リレーションが第一正規形であること(すべてのドメインが単純であること)
- SQLでは第一正規形であることをMUSTの条件とている

更新操作と更新時異常

- リレーションの更新
  - 挿入(insertion):リレーションに新しいタップルを入れる
  - 削除(deletion):リレーションから不要となったタップルを消す
  - 修正(modification):タップルの属性値を変更する
- <u>更新時異常</u>(update anomaly):更新時に発生するさまざまな異常

- 更新時異常(update anomaly)
  - タップル挿入時異常
  - タップル削除時異常
  - タップル修正時異常

更新時異常の例(0)

- 次のようなリレーション *卸売* を考える

卸売

商品名	卸先名	卸値	仕入値
バッグ	A社	22	15
バッグ	B社	22	15
ベルト	A社	12	8
ベルト	B社	14	8
香水	C社	30	20

更新時異常の例(1)

- タップル挿入時異常:
  - スカーフを仕入れ値10で仕入れられる業者
  - 卸し売りはまだしていない→(スカーフ、ー、ー、10)というタップルはキー制約上入れられない

卸売

商品名	卸先名	卸値	仕入値
バッグ	A社	22	15
バッグ	B社	22	15
ベルト	A社	12	8
ベルト	B社	14	8
香水	C社	30	20
スカーフ	ー	ー	10

キー制約上NGとなる

更新時異常の例(2)

- タップル削除時異常:
  - C社とは取り引きを停止した→タップル(香水、C社、30、20)というタップルを削除したい→香水を仕入れているとう情報が無くなる

卸売

<u>商品名</u>	<u>卸先名</u>	卸値	仕入値
バッグ	A社	22	15
バッグ	B社	22	15
ベルト	A社	12	8
ベルト	B社	14	8
香水	―	―	20

キー制約上NGとなる

更新時異常の例(3)

- タップル更新時異常：
  - C社との取り引きがバッグに変更になった→タップル(香水、C社、30、20）というタップルを更新したい→香水を仕入れているとう情報が無くなる

卸売

<u>商品名</u>	<u>卸先名</u>	卸値	仕入値
バッグ	A社	22	15
バッグ	B社	22	15
ベルト	A社	12	8
ベルト	B社	14	8
バッグ	C社	22	15
香水	ー	ー	20

キー制約上NGとなる

更新時異常の解消の考え方

- お互いに無関係な2つのデータが混在
- *卸売*(商品名、卸先名、卸値、仕入値)を*卸売[商品名、仕入値]*と*卸売[商品名、卸先名、卸値]*に分解する

卸売

商品名	卸先名	卸値	仕入値
バッグ	A社	22	15
バッグ	B社	22	15
ベルト	A社	12	8
ベルト	B社	14	8
香水	C社	30	20

分解

卸売[商品名、仕入値]

商品名	仕入値
バッグ	15
ベルト	8
香水	20

卸売[商品名、卸先名、卸値]

商品名	卸先名	卸値
バッグ	A社	22
バッグ	B社	22
ベルト	A社	12
ベルト	B社	14
香水	C社	30

注文

顧客名	商品名	数量	単価	金額
A商店	テレビ	3	198,000	594,000
Bマート	テレビ	10	198,000	1,980,000
Bマート	洗濯機	5	59,800	299,000
C社	餅つき機	1	29,800	29,800

属性名のアンダーラインはキー属性を表す

リレーション 注文（図5.1）

分解

{顧客名, 商品名, 数量, 金額}上の射影　　{商品名, 単価}上の射影

注文[顧客名, 商品名, 数量, 金額]

顧客名	商品名	数量	金額
A商店	テレビ	3	594,000
Bマート	テレビ	10	1,980,000
Bマート	洗濯機	5	299,000
C社	餅つき機	1	29,800

注文[商品名, 単価]

商品名	単価
テレビ	198,000
洗濯機	59,800
餅つき機	29,800

図 4.2　リレーション 注文の分解

分解についての考察

- 2つの射影リレーションに分解後、それらの自然結合をとると元のリレーションに復元できる→**情報無損失分解**
- 分解によりすべての更新時異常は解消されている(各自で確かめよ)
- 関数従属性　商品名→仕入値による情報無損失分解

情報無損失分解

- リレーション$R(A_1, A_2, \ldots\ldots, A_n)$の2つの属性集合をXとYとする
- X∪Y=Ω$_R$かつX∩Y=Z(≠φ)とする
- このとき、Rの２つの射影$R[X]$と$R[Y]$への分解が**情報無損失分解(information lossless decomposition)**であるとは、$R=R[X] * R[Y]$が成り立つときを言う

情報無損失分解の必要十分条件

- 命題4-1
  - リレーション$R(A_1, A_2, \ldots\ldots, A_n)$、$X$と$Y$を$X \cup Y = \Omega_R$かつ$X \cap Y \neq \phi$な属性集合とする。このとき常に$R \subseteq R[X] * R[Y]$が成立する。ここに⊆は集合の包含関係を表し、また自然結合＊は$R[X]$と$R[Y]$の共通属性集合$X \cap Y$上でとられる

情報無損失分解の必要十分条件

- 命題4-2
  - リレーション$R(A_1, A_2, \ldots\ldots, A_n)$、$X$と$Y$を$X \cup Y = \Omega_R$かつ$X \cap Y \neq \phi$なる属性集合とする。このとき常に$R \subseteq R[X]$と$R[Y]$が成立するための必要かつ十分な条件は$R$で多値従属性$X \cap Y \rightarrow\rightarrow X|Y$が成立することである

情報無損失分解の必要十分条件

- 定理4-1
  - リレーション$R(A_1, A_2, \ldots\ldots, A_n)$、$X$と$Y$を$X \cup Y = \Omega_R$かつ$X \cap Y \neq \phi$なる属性集合とするとき、$R$が射影$R[X]$と$R[Y]$に情報無損失分解されるための必要かつ十分な条件は$R$で多値従属性$X \cap Y \to\to X|Y$が成立することである
  - 系1
    - リレーション$R(A_1, A_2, \ldots\ldots, A_n)$、$X$と$Y$を$X \cup Y = \Omega_R$かつ$X \cap Y \neq \phi$なる属性集合とするとき、$R$が射影$R[X]$と$R[Y]$に情報無損失分解されるための十分条件は$R$で関数従属性$X \cap Y \to X - Y$、あるいは$X \cap Y \to Y - X$が成立することである

第二正規形

- リレーションスキーマ*R*が**第二正規形 (second normal form, 2NF)**であるとは、次の２つの条件を満たす時をいう
  - *R*は第一正規形である
  - *R*のすべての非キー属性は*R*の候補キーに**完全関数従属**している

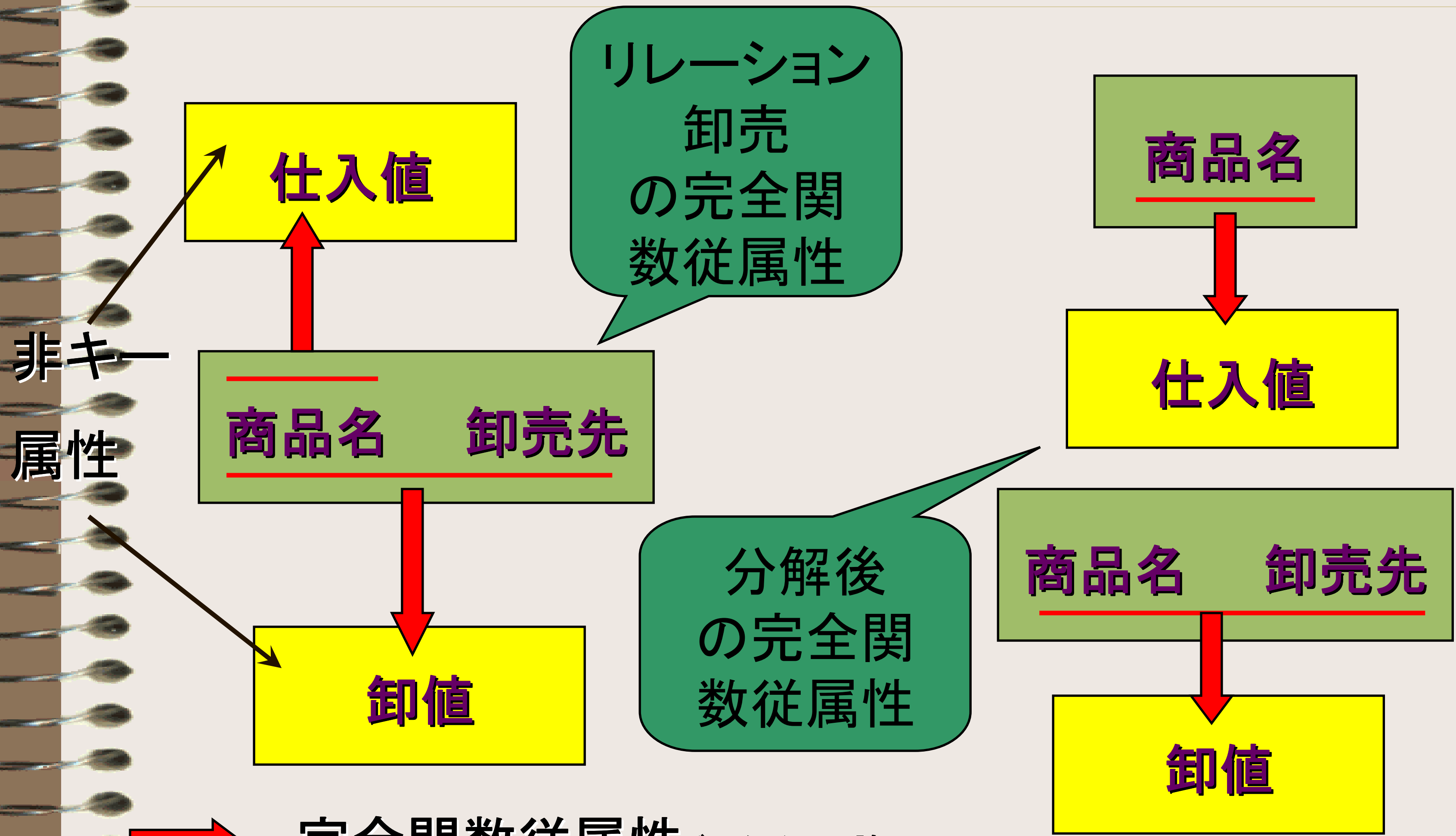
完全関数従属性

- リレーション$R(A_1, A_2, \ldots\ldots, A_n)$の2つの属性集合をXとYとする
- 関数従属性

  $(\forall t, t' \in R)(t[X]=t'[X] \Rightarrow t[Y]=t'[Y])$
- 完全関数従属性

  Xのいかなる真部分集合X'に対してもX'→Yが成立しない

完全関数従属性



完全関数従属性

［定義 1］ リレーションスキーマ $\boldsymbol{R}(A_1, A_2, \cdots, A_l, B_1, B_2, \cdots, B_m, C_1, C_2, \cdots, C_n)$ の任意のインスタンス R が，その 2 つの射影 $R[A_1A_2\cdots A_lB_1B_2\cdots B_m]$ と $R[A_1A_2\cdots A_lC_1C_2\cdots C_n]$ に情報無損失分解されるとき，およびそのときのみ

$\boldsymbol{R}$ に（自明でない）多値従属性 (multi-valued dependency，MVD)

$$\boldsymbol{A_1A_2\cdots A_l \rightarrow\!\!\!\rightarrow B_1B_2\cdots B_m}$$

が存在するという．

注文

顧客名	商品名	数量	単価	金額
A商店	テレビ	3	198,000	594,000
Bマート	テレビ	10	198,000	1,980,000
Bマート	洗濯機	5	59,800	299,000
C社	餅つき機	1	29,800	29,800

属性名のアンダーラインはキー属性を表す

リレーション 注文（図4.1）

分解

{顧客名, 商品名}上の射影

{商品名, 数量, 単価, 金額}上の射影

注文［顧客名, 商品名］

顧客名	商品名
A商店	テレビ
Bマート	テレビ
Bマート	洗濯機
C社	餅つき機

注文［商品名, 数量, 単価, 金額］

商品名	数量	単価	金額
テレビ	3	198,000	594,000
テレビ	10	198,000	1,980,000
洗濯機	5	59,800	299,000
餅つき機	1	29,800	29,800

自然結合

顧客名	商品名	数量	単価	金額
A商店	テレビ	3	198,000	594,000
A商店	テレビ	10	198,000	1,980,000
Bマート	テレビ	3	198,000	594,000
Bマート	テレビ	10	198,000	1,980,000
Bマート	洗濯機	5	59,800	299,000
C社	餅つき機	1	29,800	29,800

これらのタップルは元のリレーション注文にはなかったもので意味がない。

図 4.3　情報損失分解―無作為な分解は情報を損失すること―

	フライト番号	クルー名	乗客名
t	55	P	A
w'	55	S	A
w	55	P	B
t'	55	S	B
	55	P	C
	55	S	C

(a) リレーション フライト

フライト番号	クルー名
55	P
55	S

(b) リレーション クルー

フライト番号	乗客名
55	A
55	B
55	C

(c) リレーション 乗客

フライト番号	クルー名	乗客名
55	P	A
55	S	B
55	—	C

ここに下線は空値を表す

(d) フライト番号 $\to\!\to$ クルー名 | 乗客名 なる多値従属性が
あるので許されないリレーション フライトの例

図 4.4 多値従属性

[定義 2] リレーションスキーマ $\boldsymbol{R}(A_1, A_2, \cdots, A_l, B_1, B_2, \cdots, B_m, C_1, C_2, \cdots, C_n)$ に**関数従属性** (functional dependency, FD) $\boldsymbol{A_1 A_2 \cdots A_l \to B_1 B_2 \cdots B_m}$ が存在するとは次の条件が成立するときをいう：

R を $\boldsymbol{R}$ の任意のインスタンスとするとき

$$(\forall t,\ t' \in R)\,(t[A_1A_2\cdots A_l] = t'[A_1A_2\cdots A_l] \Rightarrow t[B_1B_2\cdots B_m] = t'[B_1B_2\cdots B_m])$$

ここに ⇨ は論理的含意 (logical implication) を表す記号である．

注文

顧客名	商品名	数量	単価	金額
A商店	テレビ	3	198,000	594,000
Bマート	テレビ	10	198,000	1,980,000
Bマート	洗濯機	5	59,800	299,000
C社	餅つき機	1	29,800	29,800

属性名のアンダーラインはキー属性を表わす

図 4.5　リレーション 注文 (図 4.1 の再掲)

—第 1 正規形であるが第 2 正規形ではない例—

第二正規形だが第三正規形でない例

- 次のようなリレーション *配属*を考える

配属

社員名	プロジェクト名	契約タイプ
e1	p1	c1
e2	p1	c1
e3	p2	c2
e4	p2	c2
e5	p3	c1

社員

<u>社員番号</u>	社員名	給与	所属	勤務地
0650	鈴木一郎	50	K55	神奈川
1508	浜崎アユ	40	K41	東　京
0231	宇田ひかる	60	K41	東　京
2034	別華武剛	40	K55	神奈川
2100	森　虎丸	40	K58	静　岡

属性名のアンダーラインはキー属性を表わす

(a) リレーション 社員 ―第二正規形ではあるが第三正規形ではない例―

社員[社員番号, 社員名, 給与, 所属]

<u>社員番号</u>	社員名	給与	所属
0650	鈴木一郎	50	K55
1508	浜崎アユ	40	K41
0231	宇田ひかる	60	K41
2034	別華武剛	40	K55
2100	森　虎丸	40	K58

社員[所属, 勤務地]

<u>所属</u>	勤務地
K55	神奈川
K41	東　京
K58	静　岡

(b) 関数従属性 所属 → 勤務地によるリレーション 社員の情報無損失分解

図 4.9 リレーション 社員 とその正規化

第三正規形

- リレーションスキーマ*R*が<u>第三正規形 (third normal form, 3NF)</u>であるとは、次の2つの条件を満たす時をいう
  - *R*は第二正規形である
  - *R*のすべての非キー属性は*R*の候補キーに<u>推移的に従属しない</u>

推移的多値従属性

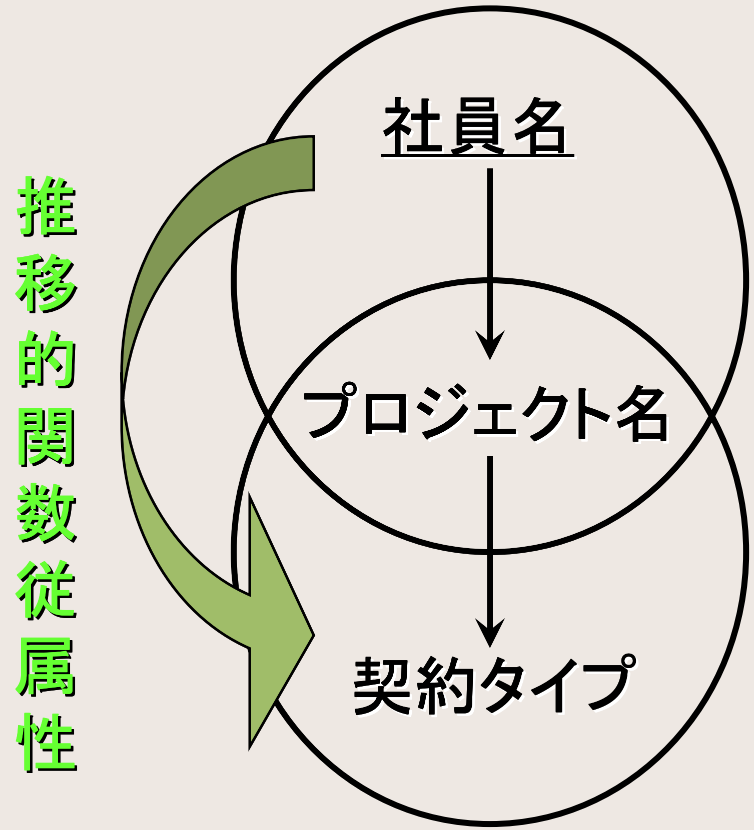
- リレーション$R(A_1, A_2,\ldots\ldots,A_n)$の2つの属性集合をXとYとする
- 多値従属性

  $(\forall t, t' \in R)(t[x]=t'[x] \Rightarrow ((t[X \cup Y], t'[Z]) \in R \wedge (t'[X \cup Y], t[Z] \in R)$ ここで、$Z=\Omega R-(X \cup Y)$
- 推移的多値従属性

  $X \rightarrow Y$、$Y \nrightarrow X$、$Y \rightarrow Z$（ただし自明でない）

  $X \rightarrow Z (Z \nrightarrow X)$を推移的多値従属制と呼ぶ

推移的関数従属性の例



関数従属性とキー

- 関数従属性を使った候補キーの定義
- 定義4-1　スーパーキー

リレーションスキーマ*R*の属性集合Hが次の条件を満たすとき*R*の**スーパーキー**と呼ばれる

*R*の任意のインスタンスRに対して次が成立する

$$(\forall t, t' \in R)(t[H]=t'[H] \Rightarrow t[\overline{H}]=t'[\overline{H}])$$

ここで、$\overline{H} = \Omega_R - H$なる差集合である

スーパーキーによる候補キーの定義

- リレーションスキーマ*R*の属性集合Kが*R*の候補キーと呼ばれるのは次の２つの条件が成立するときをいう
  - **K**は**R**のスーパーキーであり、かつ
  - **K**のいかなる真部分集合も**R**のスーパーキーとはならない

すなわち、極小のスーパーキーが候補キーである

命題3.3

- リレーションスキーマRの属性集合Kが候補キーであるのは、$\overline{K}= \Omega_R - K$として、

  (1) $K \to \overline{K}$、かつ

  (2) $(\forall H \subsetneq K)(H \not\to \overline{H})$、ここで$\overline{H}= \Omega_R - H$

が成立するとき、およびそのときのみである。

(証明)　略(ほとんど自明である。各自で考えよ)

リレーション *配属* の分解

配属

社員名	プロジェクト名	契約タイプ
e1	p1	c1
e2	p1	c1
e3	p2	c2
e4	p2	c2
e5	p3	c1

配属[社員名、プロジェクト名]

社員名	プロジェクト名
e1	p1
e2	p1
e3	p2
e4	p2
e5	p3

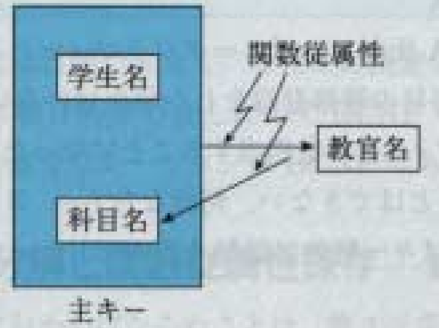
配属[プロジェクト名、契約タイプ]

プロジェクト名	契約タイプ
p1	c1
p2	c2
p3	c1

SCT

学生名	科目名	教官名
田中耕二	データベース	増永良文
田中耕二	ソフトウェア	西川博之
飯島愛子	データベース	大林弥彦
中山ゴン太	ハードウェア	喜多川優

(a) リレーションSCT
—3NFであってBCNFでない例—



(b) リレーションSCTの二つの(自明でない)関数従属性

SCT[学生名, 教官名]

学生名	教官名
田中耕二	増永良文
田中耕二	西川博之
飯島愛子	大林弥彦
中山ゴン太	喜多川優

SCT[教官名, 科目名]

教官名	科目名
増永良文	データベース
西川博之	ソフトウェア
大林弥彦	データベース
喜多川優	ハードウェア

(c) 関数従属性による情報無損失分解

図 4.11 リレーション SCT とその正規化

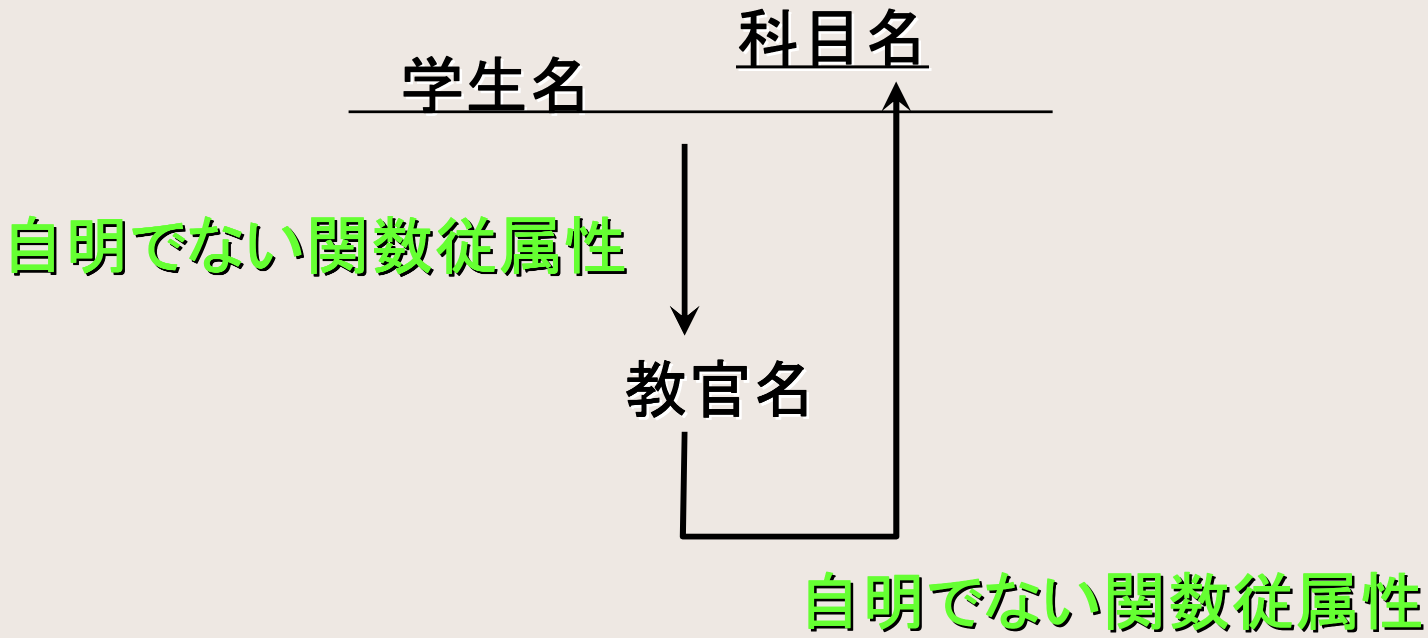
第三正規形における更新時異常の例

- 次のようなリレーション *受講を考える*

受講

<u>学生名</u>	<u>科目名</u>	教官名
s1	c1	t1
s2	p2	t2
s3	c1	t3
s4	c1	t1
s4	c3	t1

リレーション受講の2つの自明でない関数従属性



ボイス-コッド正規形

- リレーションスキーマ*R*が**<u>ボイス-コッド正規形(Boyce-Codd normal form, BCNF)</u>**であるとは、次の条件が成立する時をいう、　　X→Yを*R*の関数従属性とするとき、
  - X→Yは自明な関数従属性であるか、
  - Xは*R*のスーパーキーである

関数従属性の公理系

- $F = \{f_1, f_2, \ldots\ldots, f_p\}$をリレーションスキーマ$R$に所与（リレーションスキーマ定義者によって定義された初期）の関数従属性集合とする
- R上で成立する関数従属性はFの元だけとはかぎらない
- Fには現れない関数従属性をfとするとき、それがR上で成立するなら、fはFにより**論理内包**される、あるいは**導出**されるという

関数従属性の公理系(2)

- Fより導出されるすべての関数従属性の集合をF^+であらわし、これをFの**閉包**と呼ぶ
- F^+を求めることの意義はつぎのとおり

(1) 関数従属性は一貫性制約であるので、どれだかの関数従属性があるのか知っておく必要がある

(2) 関数従属性の集合の等価性の問題を解決できる

(3) 冗長な関数従属性を除去できる

(4) 関数従属性の理論を構築できる

関数従属性の公理系(3)

- *F*を与えてF^+を求める問題→アームストロングが公理論的アプローチで解決した
- *F*からアームストロングの諸公理を有限回適用して得られる関数従属性の集合が閉包F^+となる
- アームストロングの公理系

X,Y,Zをリレーションスキーマ*R*の属性集合とする

1. もし、Y⊆Xなら、X→Yである(反射律)
2. もしX→Yで、Z⊆ΩRなら、X∪Z→Y∪Zである(ここで、∪は和集合演算である)(添加律)
3. もし、X→YかつY→Zなら、X→Zである(推移律)

●関数従属性の公理系 (アームストロングの公理系)

(F1) X を属性集合，Y を X の部分集合とするなら $X \to Y$ である．(反射律)

(F2) $X \to Y$ かつ，Z を任意の属性集合とすると，$X \cup Z \to Y \cup Z$ である．(添加律)

(F3) $X \to Y$ かつ $Y \to Z$ なら $X \to Z$ である．(推移律)

1. 社員番号 → 住所（所与）
2. 住所 → 郵便番号（所与）
3. 社員番号 → 郵便番号（1と2と推移律）

図 4.6 アームストロングの公理系のもと 社員番号 → 郵便番号 を導出する証明

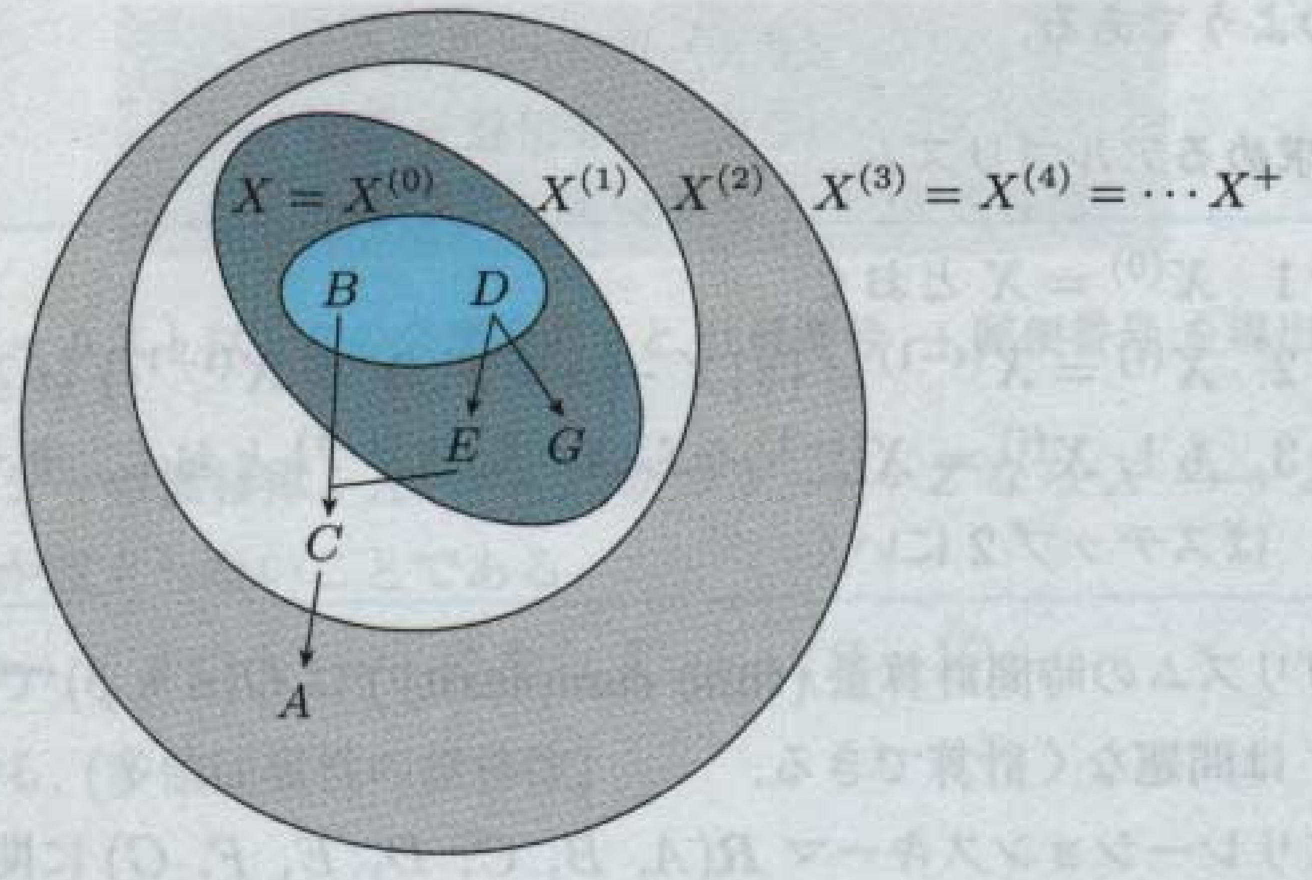


図 4.7　$X = \{B, D\}$ の F に関する閉包 X^{+} の計算

ステップ1　$G = F$ とおく．

ステップ2　G 中の各関数従属性で右辺が単一の属性でないものがあれば (これを $X \to A_1 A_2 \cdots A_n$ とする)，右辺が単一の属性である n 個の関数従属性 $X \to A_1$, $X \to A_2$, $\cdots$, X_n で置き換える．

ステップ3　G 中の各関数従属性 $X \to A$ に対して，X を構成する各属性 B に対して，$G - \{X \to A\} \cup \{\{X - B\} \to A\}$ と G が等価ならば，G の $X \to A$ を $\{X - B\} \to A$ で置き換える．

ステップ4　G の各関数従属性 $X \to A$ に対して，$G - \{X \to A\}$ が G と等価ならば，G から $X \to A$ を取り除く．

図 4.8　F の極小被覆を求めるアルゴリズム

関数従属性保存分解

- リレーションをR, $F=\{f_1, f_2,\ldots\ldots, f_p\}$を$R$に所与の関数従属性集合とする。2つの属性集合をXとYとする
- f：X→YをRの関数従属性とするとき、Rをfにより二つの射影$R[X,Y]$と$R[X,Z]$、ここで、Z＝ Ω_Rー（X∪Y）に情報無損失分解する。
- $F^+|(X\cup Y)$でF^+の元のうちX∪Y上で定義されているものの集合、$F^+|(X\cup Z)$でF^+の元のうちX∪Z上で定義されているもののみの集合とする。このとき、もし$F^+=(F^+|(X\cup Y)\cup F^+|(X\cup Z))^+$が成立するなら、Rのfによる情報無損失分解は関数従属性保存であるという。

ボイス-コッド正規形における更新時異常の例

- 次のようなリレーション *講習会を考える*

講習会

講習名	指導員名	参加者名
パソコン	山田	小林
パソコン	山田	石川
ワープロ	青木	野村
ワープロ	青木	鈴木
ワープロ	青木	森下
ワープロ	加藤	野村
ワープロ	加藤	鈴木
ワープロ	加藤	森下
データベース	山田	田中

●関数従属性保存の第 3 正規形情報無損失分解アルゴリズム

ステップ 1　F の極小被覆 G をひとつ見つける．

ステップ 2　X を決定子に持つ G の全ての関数従属性 $X \to A_1$, $X \to A_2$, $\cdots$, $X \to A_k$ に対して，リレーションスキーマ $\boldsymbol{R}[X, A_1, A_2, \cdots, A_k]$ を作る（このとき X はリレーションの主キーとなる）．この操作を G の全ての関数従属性に対して行う．その結果 $\boldsymbol{R}_1$, $\boldsymbol{R}_2$, $\cdots$, $\boldsymbol{R}_k$ なるリレーションスキーマが得られる．

ステップ 3　もし，ステップ 2 で作ったリレーションスキーマのどれもが $\boldsymbol{R}$ のいずれの候補キーも含まないならば，候補キーのひとつを K として，$\boldsymbol{R}[K]$ を作る．この場合 $\{\boldsymbol{R}_1, \boldsymbol{R}_2, \cdots, \boldsymbol{R}_k, \boldsymbol{R}[K]\}$ が，そうでなければ $\{\boldsymbol{R}_1, \boldsymbol{R}_2, \cdots, \boldsymbol{R}_k\}$ が $\boldsymbol{R}$ の情報無損失かつ関数従属性保存の第 3 正規形分解となる．

講習会の多値従属性による情報無損失分解

講習会

多値従属性
講習名→→指導員名｜参加者名
による情報無損失分解

講習名	指導員名	参加者名
パソコン	山田	小林
パソコン	山田	石川
ワープロ	青木	野村
ワープロ	青木	鈴木
ワープロ	青木	森下
ワープロ	加藤	野村
ワープロ	加藤	鈴木
ワープロ	加藤	森下
データベース	山田	田中

講習会[講習名、指導員名]

講習名	指導員名
パソコン	山田
ワープロ	青木
ワープロ	加藤
データベース	山田

講習会[講習名、参加者名]

講習名	参加者名
パソコン	小林
パソコン	石川
ワープロ	野村
ワープロ	鈴木
ワープロ	森下
データベース	田中

第四正規形

- リレーションスキーマ*R*が**第四正規形 (fourth normal form, 4NF)**であるとは、次の条件が成立する時をいう、X→→Yを*R*の多値従属性とするとき、
  - X→→Yは自明な多値従属性であるか、
  - Xは*R*のスーパーキーである

リレーションが3以上に分解される場合の更新時異常の例

- 次のようなリレーション ツアーを考える

ツアー

ツアー名	キャリア名	代理店名
ハワイ	JAL	オリエント
ハワイ	JAL	岐大観光
ヨーロッパ	JAL	オリエント
ハワイ	ANA	オリエント

3つの射影による分解

ツアー名	キャリア名
ハワイ	JAL
ヨーロッパ	JAL
ハワイ	ANA

キャリア名	代理店名
JAL	オリエント
JAL	岐大観光
ANA	オリエント

代理店名	ツアー名
オリエント	ハワイ
岐大観光	ハワイ
オリエント	ヨーロッパ

フライト

フライト番号	クルー名	乗客名
55	P	A
55	P	B
55	P	C
55	S	A
55	S	B
55	S	C
505	P'	A'
505	S'	A'

(a) リレーション フライト
— ボイス-コッド正規形ではあるが第四正規形ではない例 —

フライト[フライト番号, クルー名]

フライト番号	クルー名
55	P
55	S
505	P'
505	S'

フライト[フライト番号, 乗客名]

フライト番号	乗客名
55	A
55	B
55	C
505	A'

(b) 第四正規形への正規化

図 4.12 リレーション フライトとその正規化

第五正規形

- リレーションスキーマ*R*が**第五正規形(fifth normal form, 5NF)**であるとは、次の条件が成立する時をいう

  X_1、X_2、…、X_nを$X_1 \cup X_2 \cup \ldots \cup X_n = \Omega_R$なる任意の属性集合として、$*(X_1,X_2,\ldots,X_n)$をRの結合従属性とする
  - $*(X_1,X_2,\ldots,X_n)$は自明な結合従属性であるか、
  - 各X_iは*R*のスーパーキーである

第六正規形以上は

- 第一から第五正規形への正規化では、常に縦方向の分解を利用してきた
- そのような意味で、第六正規系は存在しない
- なぜなら、それは第五正規系において射影分解する時点で可決できるはず→第五正規系は**射影-結合正規形**とも呼ばれる
- 水平に分解するドメイン-キー正規形の理論もあるが、ここでは省略する

供給

供給元	部　品	供給先
A11	P101	K55
A11	P101	K41
A11	P102	K41
A12	P102	K41

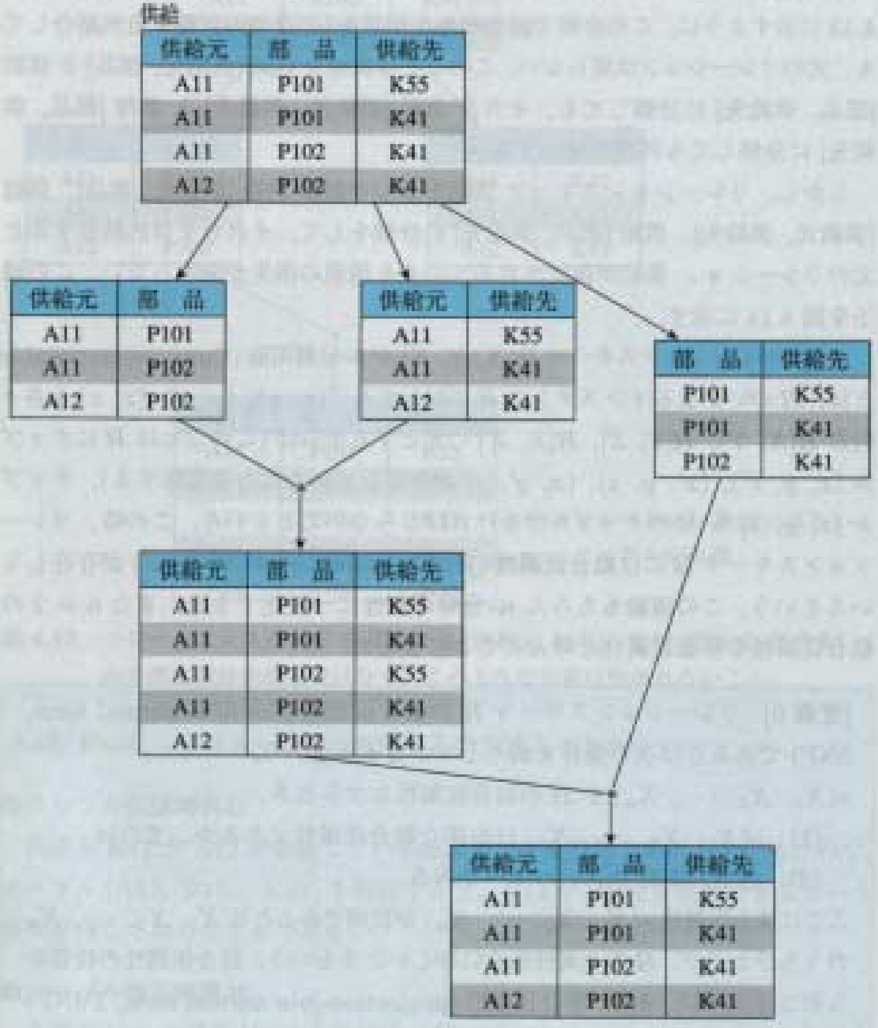
供給元	部　品
A11	P101
A11	P102
A12	P102

供給元	供給先
A11	K55
A11	K41
A12	K41

供給元	部　品	供給先
A11	P101	K55
A11	P101	K41
A11	P102	K55
A11	P102	K41
A12	P102	K41

←--- リレーション 供給にはないタップルが出現

図 4.13　リレーションスキーマ 供給の供給 [供給元, 部品]，供給 [供給元, 供給先] への分解は情報無損失ではなく，このような分解は許されないこと



供給

供給元	部　品	供給先
A11	P101	K55
A11	P101	K41
A11	P102	K41
A12	P102	K41

供給元	部　品
A11	P101
A11	P102
A12	P102

供給元	供給先
A11	K55
A11	K41
A12	K41

部　品	供給先
P101	K55
P101	K41
P102	K41

供給元	部　品	供給先
A11	P101	K55
A11	P101	K41
A11	P102	K55
A11	P102	K41
A12	P102	K41

供給元	部　品	供給先
A11	P101	K55
A11	P101	K41
A11	P102	K41
A12	P102	K41

図 4.14　リレーションスキーマ 供給の供給 [供給元, 部品], 供給 [供給元, 供給先], 供給 [部品, 供給先] の 3-分解は情報無損失分解であること

リレーショナルデータベースの設計

- 商品販売に関するデータベースを例にとり解説する
  - 販売取引→受注伝票が発生
  - 受注伝票とそれに関連するデータのデータベース化を試みる
- 手順は
  - 販売取引における制約をリストアップ
  - 要求条件を整理
  - 概念モデリングから論理モデリングへ

取引上の制約

1. 伝票番号は営業所単位でユニーク
2. 一取引の商品数は一伝票でまかなえる
3. 単価は商品ごと、営業所ごとの裁量であるていどの値引きはOK
4. 新商品の商品名は旧商品とは重複しない
5. 取引ごとに値引き(一括値引き)をする場合があるが、これは金額がマイナスの商品を販売したとして処理する
6. 一つの商品はかならず一つの商品カテゴリに属する

要求条件

1. 日付XX/XX/XXからYY/YY/YYまでの間で、一取引の合計がZZ円以上となった伝票を営業所単位で引き抜き、金額が大きい順に出力せよ
2. 日付XX/XX/XXからYY/YY/YYまでの間に、商品名Aを納入した取引先と販売した営業所名を出力せよ
3. 営業所ごと、商品カテゴリごと、月ごとの販売実績一覧表を作成せよ
4. 毎月、大口取引先（上位30社）の一覧表を作成せよ
5. 営業所別に販売実績の前年同月比一覧表をさくせいせよ

モデリング

- モデリングは、
  実世界→概念モデリング→論理モデリング
  のような2段階で進めるのが望ましい
- 概念モデリング：例えば実体-関連モデル(Entity-Relationship Model)などを使う
- 論理モデリング：リレーショナルモデル

実体-関連図と関数従属関係

- 実体関連図（次スライド）は次の関数従属関係があることを表している

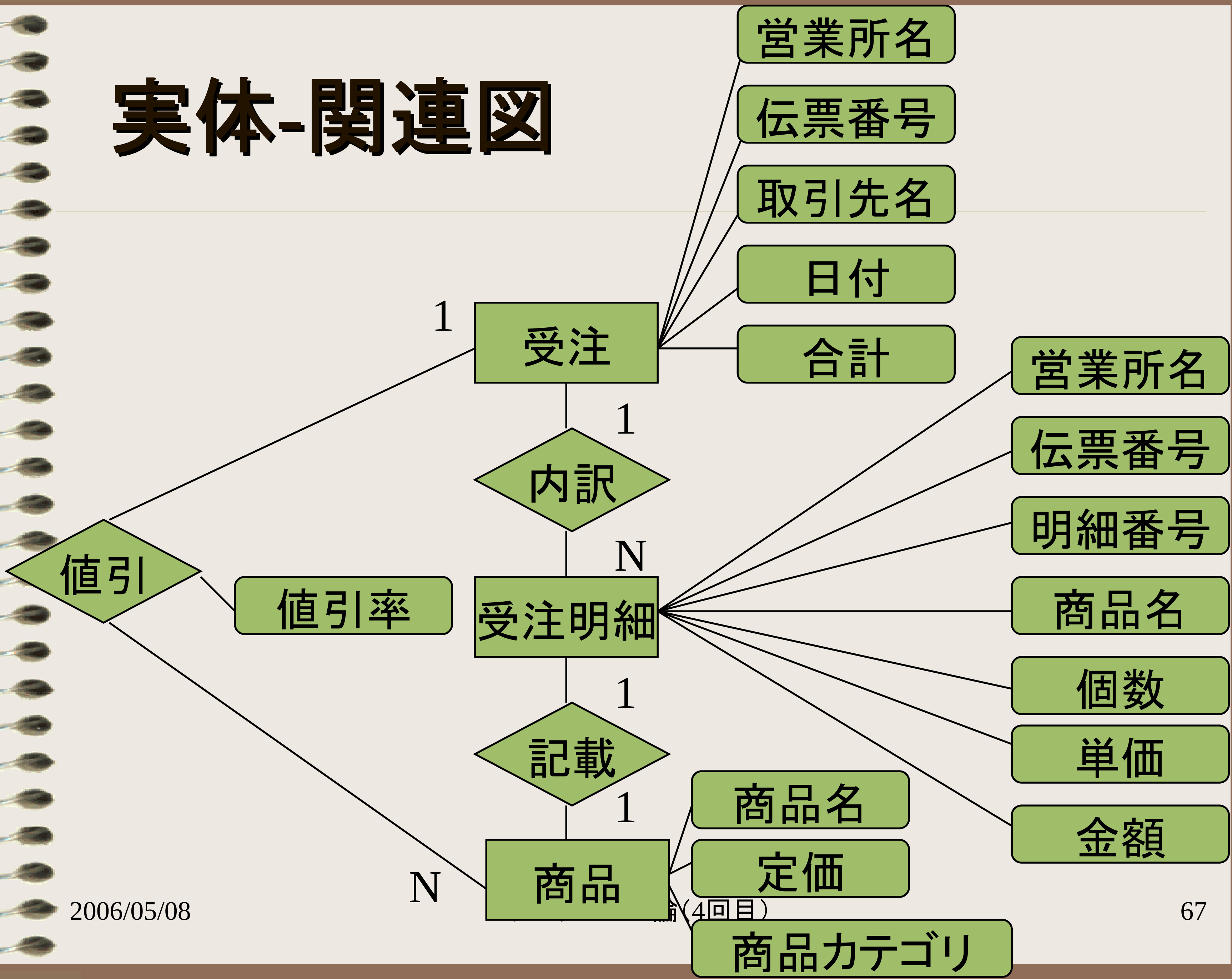
(1) {営業所名, 伝票番号}→{取引先名, 日付, 合計}

(2) {営業所名, 伝票番号, 明細番号}→{商品名, 個数, 単価, 金額}

(3) {商品名}→{定価, 商品カテゴリ}

(4) {営業所名, 伝票番号, 商品名}→{値引率}

実体-関連図



受注
営業所名
伝票番号
取引先名
日付
合計
1
内訳
1
N
受注明細
営業所名
伝票番号
明細番号
商品名
個数
単価
金額
1
記載
1
商品
商品名
定価
商品カテゴリ
値引
値引率
1
N

実表とビューおよび結果リレーション

- <u>実表</u>:スキーマにより定義され、データが入れられた実在するリレーションで、外部記憶装置に記憶される
- <u>ビュー</u>:ビュー表とも呼ぶ。任意に作成される仮想のリレーションであり、外部記憶媒体に記憶されていないが、利用者はあたかもそれが実在する表のように操作することができる
- <u>結果リレーション</u>:ユーザが各種の操作を行ったときにできる結果を保持するリレーションで、結果リレーションに対する操作ができるかどうかは実装系による

まとめ

- リレーションの高次正規形
  - 第二正規形
  - 第三正規形
  - ボイス-コッド正規形
  - 第四正規形
  - 第五正規形
- リレーショナルデータベースの設計
  - 要件の確認
  - モデリング（実体-関連モデルによるモデリング例）

レポート課題（2回目）

- レポート課題
  - リレーションを高次正規形とする意義を簡単に説明せよ
- 締め切り：5月22日タイムスタンプ有効にて電子メールで
  - メールアドレス：*okuyama@dsl.gr.jp*
  - サブジェクト：データベース論第四回課題
  - 本文の最初に必ず学籍番号、氏名を記入