

▼表4-14 表4-12の関係“部署別従業員管理者”を第4正規形にする

関係“部署別従業員”

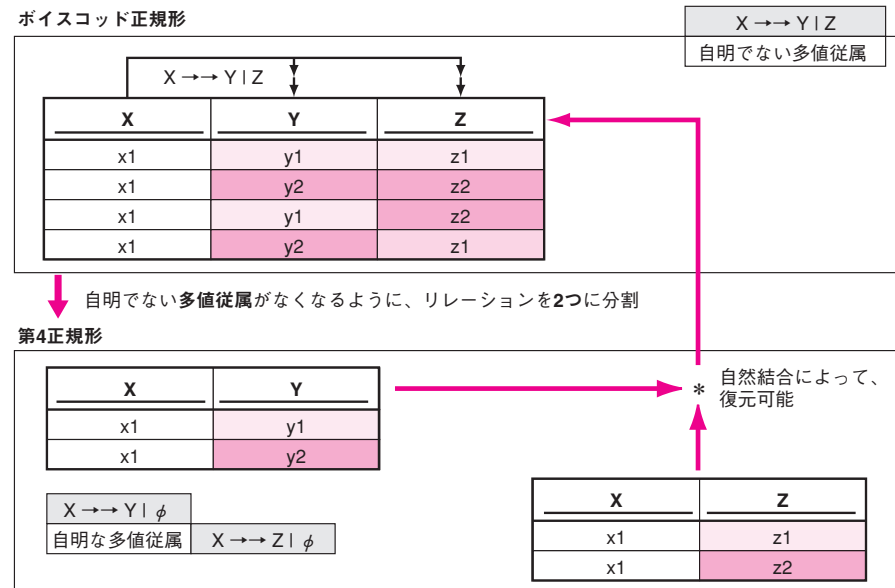
部署ID	従業員ID
0001	000000001
0001	000000009
0001	000000018
0004	000000016
0002	000000002
0002	000000009
0002	000000023
0002	000000030
0003	000000013
0003	000000029
⋮	⋮

関係“部署別管理者”

部署ID	管理者
0001	000000001
0004	000000016
0002	000000002
0002	000000009
0003	000000013
⋮	⋮

正規化後の関係“部署別従業員”と“部署別管理者”には、自明でない多値従属性がないので、これらの関係は、第4正規形です。

図4-5は、関数従属性とボイスコード正規形から第4正規形への正規化との関係を、一般化して表現したものです。



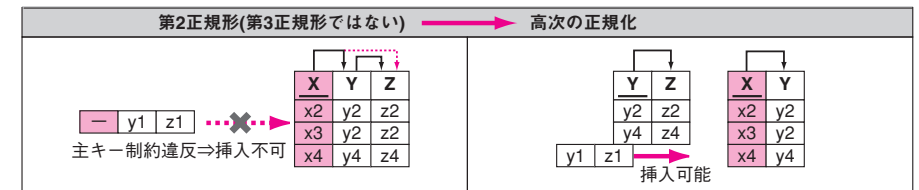
▲図4-5 ボイスコード正規形から第4正規形への変換

4.7 更新時異状

低次の正規形では、タプルの挿入・更新・削除において、データの整合性を維持することが困難になるという問題が発生する場合があります。このような問題を更新時異状といいます。

タプル挿入時異状

第3正規形でない場合には、推移的関数従属性 ($X \rightarrow Y$, $Y \rightarrow Z$ のときの $X \rightarrow Z$) が存在します。被決定子 (Z) と、それを被決定子とする推移的でない関数従属性 ($Y \rightarrow Z$) の決定子 (Y) との対応を示す事実が決まったときに、そのデータを挿入しようとすると、XとYの対応を示す事実が未決定ならば、Xの値としてナル値を挿入することになり、**主キー制約に違反**することです (図4-6)。



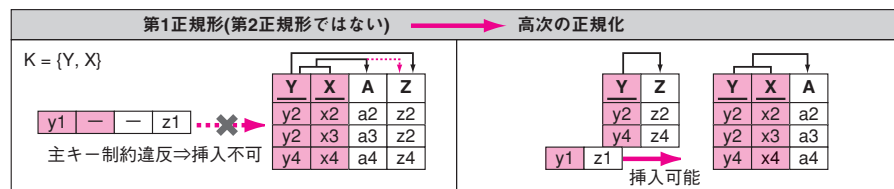
▲図4-6 第3正規形でない第2正規形でのタプル挿入時異状

たとえば、表4-12 (p.158) の関係“部署 (第2正規形)”にタプルを挿入する場合を考えてみます。この関係には、関数従属性“部署ID→部署業務”、および“部署業務→業務区分”があり、“部署ID→業務区分”という推移的関数従属性があります。直接業務でも間接業務でもない新たな「研究業務」という業務区分、その部署業務として「研究」を追加することになりました。しかし、実際に研究業務を行う部署は、未だ決定していません。このとき、関係“部署 (第2正規形)” (部署ID, 部署業務, 業務区分) にタプルを挿入しようとすると、(NULL, '研究', '研究業務') というタプルを挿入することになります。しかし、主キーである部署IDには、ナル値を挿入する必要があるため、主キー制約に違反することになり、このタプルを挿入することができません。

表4-13 (p.160) のように第3正規形に正規化されているならば、関係“業務区分” (部署業務, 業務区分) にタプル ('研究', '研究業務') を挿入することができます。

第2正規形でない場合には、候補キー ($K = \{Y, X\}$) の真部分集合 (Y) に対する非キー属性 (Z) の関数従属性 ($Y \rightarrow Z$) が存在するので、その被決定子 (Z) と決定子 (Y) の対応を示す事実が決まったときに、そのデータを挿入しようとすると、 $(K - Y) = X$ と

Yの対応Kを示す事実が未決定ならば、Xの値としてナル値を挿入することになり、**主キー制約に違反**することです(図4-7)。



▲図4-7 第2正規形でない第1正規形でのタプル挿入時異状

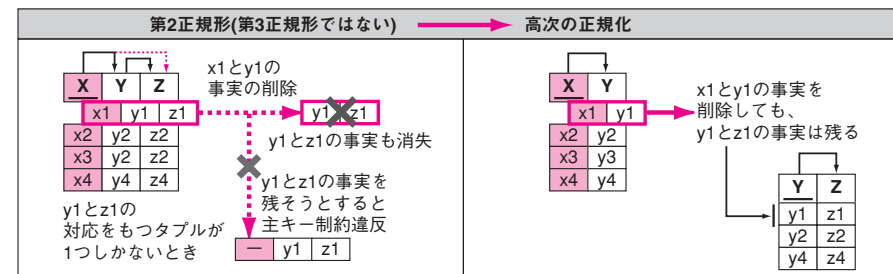
たとえば、表4-11 (p.156) の関係“部署 (第1正規形)”にタプルを挿入する場合を考えてみます。この関係には、候補キー“{部署ID, 従業員ID, 管理者}”の真部分集合である“従業員ID”を決定子とする関数従属性“従業員ID→従業員名”があります。新たな従業員“平山美幸”を採用することになり、従業員IDとして、'000000033'を割り当てましたが、どの部署に配属するかは、未だ決定していません。このとき、関係“部署 (第1正規形)” (部署ID, 部署業務, 業務区分, 従業員ID, 従業員名, 管理者) にタプルを挿入しようとする、(NULL, NULL, NULL, '000000033', '平山美幸', NULL) というタプルを挿入することになりますが、主キー (部署ID, 従業員ID, 管理者) の値は、(NULL, '000000033', NULL) になるので、主キー制約違反になり、挿入することができません。しかし、表4-12 (p.158) のように第2正規形に正規化されているならば、関係“従業員” (従業員ID, 従業員名) にタプル ('000000033', '平山美幸') を挿入することができます。

また、YとZの間のある事実情報をもつタプルがすでに存在するとき、YとZの間的事实情報の部分だけが同じである別の新しいタプルを挿入することによって、その部分に関して**冗長なデータ**をもつことになります。

タプル削除時異状

第3正規形でない場合には、推移的関数従属性 ($X \rightarrow Y$, $Y \rightarrow Z$ のときの $X \rightarrow Z$) が存在します。被決定子 (Z) と、それを被決定子とする推移的でない関数従属性 ($Y \rightarrow Z$) の決定子 (Y) との対応を示す事実をもつタプルが関係 (リレーション) 中に1つしか存在しないとき、そのタプルを削除すると、YとZのその対応関係を示す事実の情報も失われてしまいます。

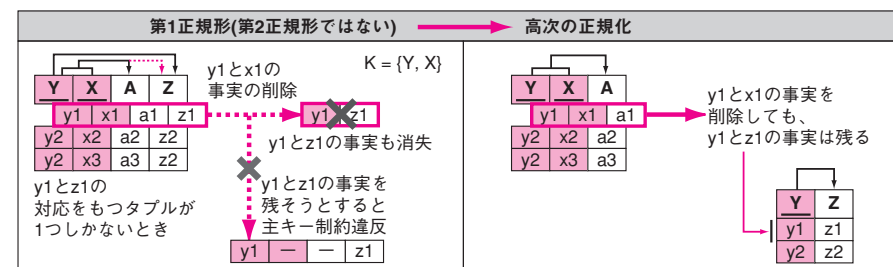
また、事実の消失を防ぐために、その事実を残そうとしても、タプル挿入時異状の場合と同様に主キー制約違反になります(図4-8)。



▲図4-8 第3正規形でない第2正規形でのタプル削除時異状

たとえば、表4-12 (p.158) の関係“部署 (第2正規形)”からタプルを削除する場合を考えてみます。この関係には、関数従属性“部署ID→部署業務”、および“部署業務→業務区分”があり、“部署ID→業務区分”という推移的関数従属性があります。部署IDが'0003'の部署を廃止するために、関係“部署 (第2正規形)” (部署ID, 部署業務, 業務区分) から ('0003', '総務', '間接業務') のタプルを削除すると、総務という部署業務の業務区分が間接業務であるという関係も削除され、この関係をもつタプルがほかになければ、この情報が失われることになります。また、このときに、この情報を残そうとすると、('0003', '総務', '間接業務') のタプルを (NULL, '総務', '間接業務') に更新する必要がありますが、主キーの部署IDがナル値になり、主キー制約に違反するため、このような更新もできません。しかし、表4-13 (p.160) のように第3正規形に正規化されているならば、関係“業務区分” (部署業務, 業務区分) のタプル ('総務', '間接業務') を残して、関係“部署 (第3正規形)” (部署ID, 部署業務) から ('0003', '総務') のタプルを削除することができます。

第2正規形でない場合には、候補キー ($K = \{Y, X\}$) の真部分集合 (Y) に対する非キー属性 (Z) の関数従属性 ($Y \rightarrow Z$) が存在します。その被決定子 (Z) と決定子 (Y) の対応を示す事実をもつタプルが関係 (リレーション) 中に1つしか存在しないとき、そのタプルを削除すると、YとZのその対応関係を示す事実の情報も失われてしまいます(図4-9)。

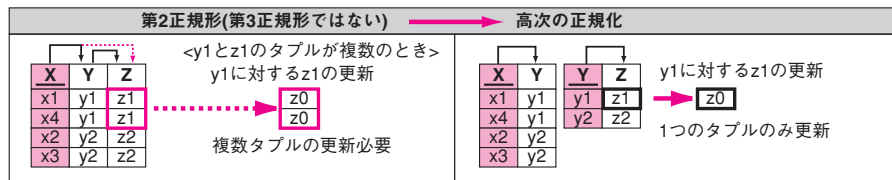


▲図4-9 第2正規形でない第1正規形でのタプル削除時異状

たとえば、表4-11 (p.156) の関係“部署 (第1正規形)”からタプルを削除する場合を考えてみます。この関係には、候補キー“{部署ID, 従業員ID, 管理者}”の真部分集合である“従業員ID”を決定子とする関数従属性“従業員ID→従業員名”があります。部署IDが'0004'の部署を廃止するために、その部署と従業員および管理者との関連を削除しようとして、関係“部署 (第1正規形)” (部署ID, 部署業務, 業務区分, 従業員ID, 従業員名, 管理者) から ('0004', '設計', '直接業務', '000000016', '鈴木一郎', '000000016') のタプルを削除すると、従業員IDが'000000016'の従業員が'鈴木一郎'であるという関係をもつタプルがほかになければ、この情報が失われます。また、この情報を残そうとすると、('0004', '設計', '直接業務', '000000016', '鈴木一郎', '000000016') のタプルを (NULL, NULL, NULL, '000000016', '鈴木一郎', NULL) に更新する必要がありますが、主キー (部署ID, 従業員ID, 管理者) の値の一部がナル値なので、主キー制約に違反するため、このような更新もできません。しかし、表4-12 (p.158) のように第2正規形に正規化されているならば、関係“従業員” (従業員ID, 従業員名) のタプル ('000000016', '鈴木一郎') を残して、関係“部署別従業員管理者” (部署ID, 従業員ID, 管理者) から ('0004', '000000016', '000000016') のタプルを削除することができます。

タプル更新時異状

第3正規形でない場合には、推移的関数従属性 ($X \rightarrow Y$, $Y \rightarrow Z$ のときの $X \rightarrow Z$) が存在するので、被決定子 (Z) と、それを被決定子とする推移的でない関数従属性 ($Y \rightarrow Z$) の決定子 (Y) との対応を示す事実情報が複数のタプルで冗長に格納されている可能性があります。このとき、このYとZの間の事実情報に関する更新を行うならば、データの整合性を維持するために、複数のタプルを更新する必要があるということです (図4-10)。

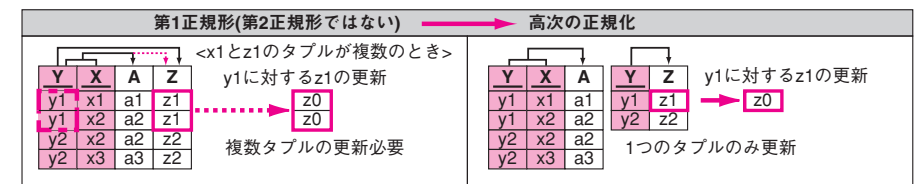


▲図4-10 第3正規形でない第2正規形でのタプル更新時異状

たとえば、表4-12 (p.158) の関係“部署 (第2正規形)”中のタプルを更新する場合を考えてみます。この関係には、関数従属性“部署ID→部署業務”、および“部署業務→業務区分”があり、“部署ID→業務区分”という推移的関数従属性があります。部署業

務が'設計'の業務区分を'直接業務'から'設計業務'に変更しようとする、関係“部署 (第2正規形)” (部署ID, 部署業務, 業務区分) 中の複数 (2つ) のタプル ('0001', '設計', '直接業務') と ('0004', '設計', '直接業務') を更新する必要があります。しかし、表4-13 (p.160) のように第3正規形に正規化されているならば、関係“業務区分” (部署業務, 業務区分) 中の1つのタプル ('設計', '直接業務') だけを更新すればよく、データの不整合を起こしにくいといえます。

第2正規形でない場合には、候補キー ($K = \{Y, X\}$) の真部分集合 (Y) に対する非キー属性 (Z) の関数従属性 ($Y \rightarrow Z$) が存在するので、被決定子 (Z) と決定子 (Y) の対応を示す事実情報が複数のタプルで冗長に格納されている可能性があります。このとき、このYとZの間の事実情報に関する更新を行うならば、データの整合性を維持するために、複数のタプルを更新する必要があるということです (図4-11)。



▲図4-11 第2正規形でない第1正規形でのタプル更新時異状

たとえば、表4-11 (p.156) の関係“部署 (第1正規形)”中のタプルを更新する場合を考えてみます。この関係には、候補キー“{部署ID, 従業員ID, 管理者}”の真部分集合である“従業員ID”を決定子とする関数従属性“従業員ID→従業員名”があります。従業員IDが'000000009'の従業員の名前を'田中夏美'から'中村夏美'に変更するためには、関係“部署 (第1正規形)” (部署ID, 部署業務, 業務区分, 従業員ID, 従業員名, 管理者) 中の複数 (3つ) のタプル ('0001', '設計', '直接業務', '000000009', '田中夏美', '000000001'), ('0002', '開発', '直接業務', '000000009', '田中夏美', '000000002') および ('0002', '開発', '直接業務', '000000009', '田中夏美', '000000009') を更新する必要があります。しかし、表4-12 (p.158) のように第2正規形に正規化されているならば、関係“従業員” (従業員ID, 従業員名) 中の1つのタプル ('000000009', '田中夏美') だけを更新すればよく、データの不整合を起こしにくいといえます。

また、タプルの更新では、ある値が削除されるので、タプル削除時異状と同様の異状も発生する可能性があります。