

テーブルの正規化

ケーススタディ

GeneXus™

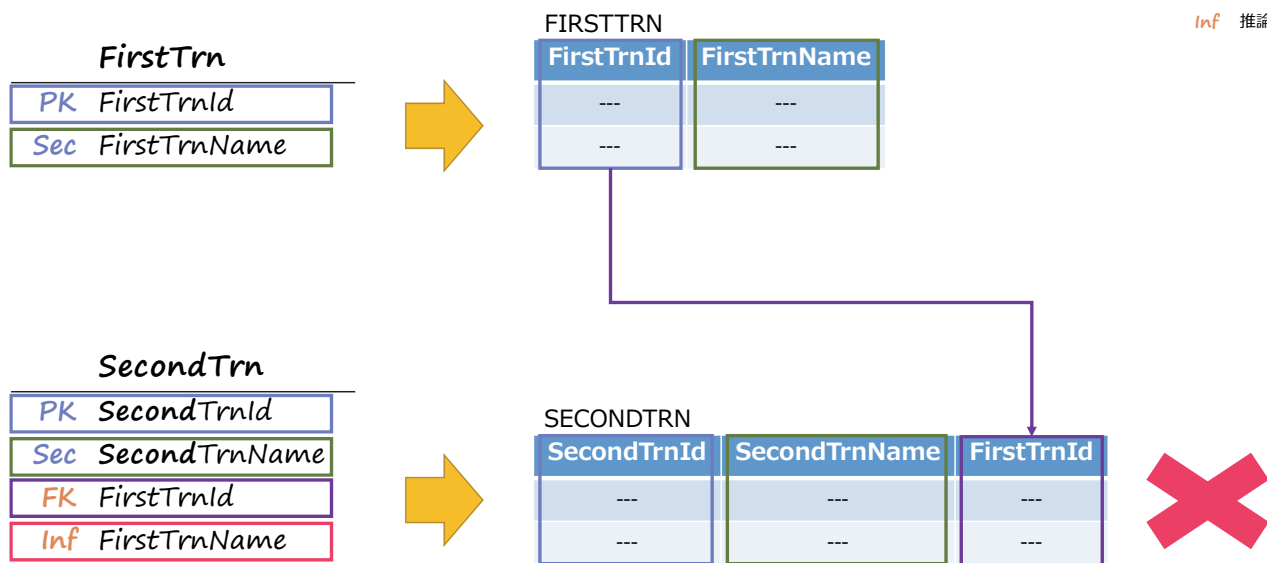
このコースでは、GeneXus がどのようにトランザクション設計からデータベース構造を判断するのかを説明してきました。

また、参照整合性チェックを自動的に行うこと、そのために各テーブルに主キーインデックスと外部キーインデックスを作成して使用することも見てきました。

ここでは、ケーススタディを使い、GeneXus がどのようにデータベースを正規化し、特定のトランザクション設計からテーブル構造を判断するのかを見ていきます。

正規化されたデータベース: 概念

PK 主キー
 Sec 従属項目
 FK 外部キー
 Inf 推論項目



正規化されたデータベースの概念から再度確認します。

正規化とは、データの重複、冗長性がないことを意味します。

これは、次の3点にまとめることができます。

1. 1つのテーブルのみに従属項目属性が存在
2. 推論される項目属性は含まない
3. 他のテーブルに含めることができる項目属性は、外部キーとして利用する主キーのみ

ケーススタディ: トランザクション設計

Country

CountryId* (PK)
CountryName (Sec)

TouristGuide

TouristGuideId* (PK)
TouristGuideName (Sec)

Trip

TripId*
TripDate
TripPrice
TouristGuideId
TouristGuideName
CountryId
CountryName

Customer

CustomerId*
CustomerName
CustomerTripsQty

Trip

(
TripId*
TripDate
TripPrice
CountryId
CountryName
CustomerTripMiles
)

それでは、ケーススタディを見ていきます。次のようなトランザクション設計を想定します。

モデルは国 (Country)、特定の国へのツアー (Trip)、ツアーを担当するツアーガイド (TouristGuide)、ツアーに参加する顧客 (Customer) から構成されます。

ここで、最初に意識していただきたい点として、テーブルの正規化について考える場合、各トランザクションだけを見て判断ではなく、全体を分析し、データモデルがどのように生成されるか考慮する必要があります。

これは、前述の通り、トランザクション同士の関係性もデータモデルへの影響を与えるためです。

初めに、Country と TouristGuide を確認すると、トランザクションは単純な単一レベルのトランザクションであり、それぞれに識別子と名前があり、外部キーはありません。

それぞれ主キーと従属する項目属性のみを持っています。

ケーススタディ: 正規化されたテーブル

COUNTRY

Country X	
Structure Indexes	
名前	タイプ
Country 構造	
CountryId	Numeric(4,0)
CountryName	VarChar(40)
Structure Indexes	
項目属性	順序
Country インデックス	
ICountry	主キー
CountryId	昇順

TOURISTGUIDE

TouristGuide X	
Structure Indexes	
名前	タイプ
TouristGuide 構造	
TouristGuideId	Numeric(4,0)
TouristGuideName	VarChar(40)
Structure Indexes	
項目属性	順序
TouristGuide インデックス	
ITouristGuide	主キー
TouristGuideId	昇順

GeneXus により、Country トランザクションからは、COUNTRY テーブルが作成され、TouristGuide トランザクションからは、TOURISTGUIDE テーブルが作成されました。スライドにそれぞれの構造を示します。

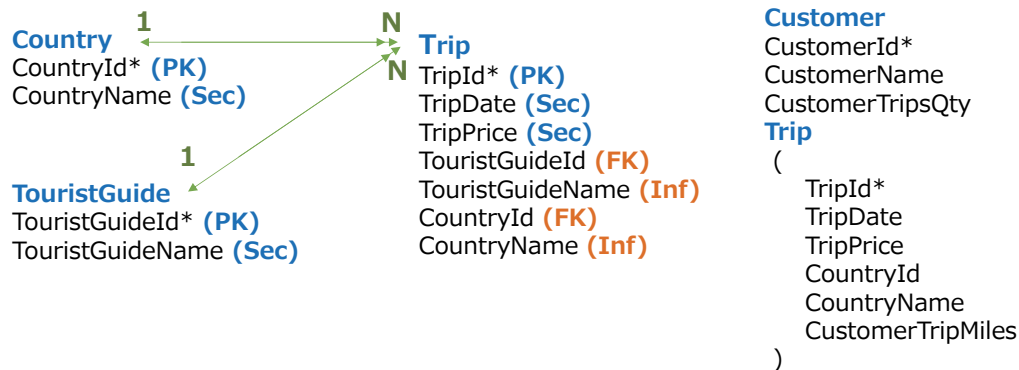
COUNTRY テーブルには主キーとして CountryId、従属する項目属性として CountryName があります。

GeneXus では、このテーブルに対して、CountryId に基づく主キーインデックスが作成されます。

ここまでに説明したとおり、主キーインデックスは、主キーの一意性を制御し、参照整合性チェックを効率的に実行するために、自動的に作成されます。

TOURISTGUIDE テーブルには主キーとして TouristGuideId、従属する項目属性として TouristGuideName があります。このテーブルには、TouristGuideId 項目属性に基づく主キーインデックスが作成されます。

ケーススタディ: トランザクション設計



続いて Trip トランザクションを確認します。TripId が主キーの項目属性、TripDate と TripPrice が従属する項目属性です。

このトランザクションでは、2 つの外部キーがあります。

1 つは、TouristGuideId です。

そして TouristGuideName はその外部キーに基づき、推論されます。

この外部キーがあることにより、TouristGuide トランザクションと Trip トランザクションは 1 対 N の関係になっています。

もう 1 つは、CountryId であり、その外部キーに基づき CountryName が推論されます。1 つ目同様に、Country トランザクションと Trip トランザクションも 1 対 N の関係になります。

ケーススタディ: 正規化されたテーブル

COUNTRY

Country X	
Structure	Indexes
名前	タイプ
Country 構造	
• CountryId	Numeric(4,0)
• CountryName	VarChar(40)
Structure	Indexes
項目属性	順序
Country インデックス	
ICountry	主キー
• CountryId	昇順

TOURISTGUIDE

TouristGuide X	
Structure	Indexes
名前	タイプ
TouristGuide 構造	
• TouristGuideId	Numeric(4,0)
• TouristGuideName	VarChar(40)
Structure	Indexes
項目属性	順序
TouristGuide インデックス	
ITouristGuide	主キー
• TouristGuideId	昇順

TRIP

Trip X	
Structure	Indexes
名前	タイプ
Trip 構造	
• TripId	Numeric(4,0)
• TripDate	Date
• TripPrice	Numeric(4,0)
• TouristGuideId	Numeric(4,0)
• CountryId	Numeric(4,0)
Structure	Indexes
項目属性	順序
Trip インデックス	
ITrip	主キー
• TripId	昇順
ITrip1	外部キー
• CountryId	昇順
ITrip2	外部キー
• TouristGuideId	昇順

GeneXus により、Trip トランザクションからは、TRIP テーブルが作成されました。スライドにその構造を追加しました。

主キーとして、TripId があり、従属する項目属性として、TripDate、TripPrice が含まれています。
その他、外部キーとなる TouristGuideId、CountryId が含まれています。

冒頭に説明した通り、値が推論できる項目属性は、物理的な列は含まれません。そのため、TouristGuideId の値から推論できる TouristGuideName、CountryId の値から推論できる CountryName は、含まれていないことが確認できます。

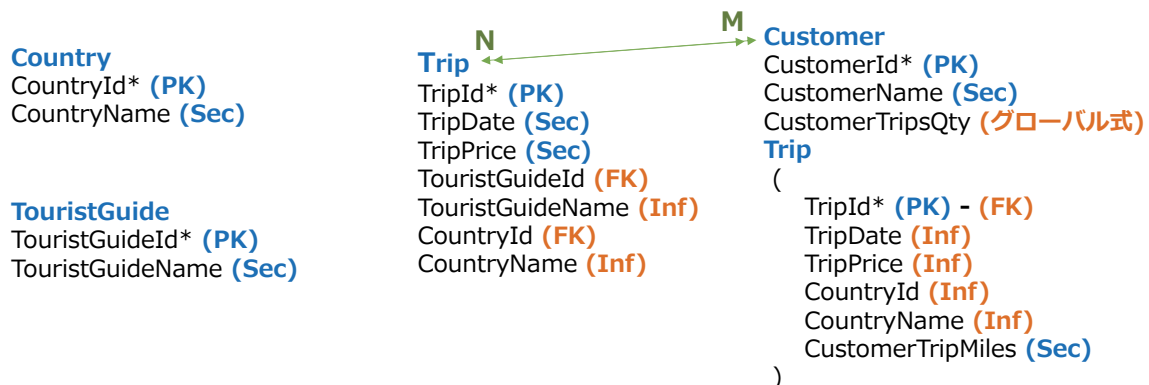
TRIP テーブルに対して作成されるインデックスは、TripId に基づく主キーインデックスと、CountryId、TouristGuideId それぞれに基づく外部キーインデックスです。主キーインデックスにより、ツアーを新規登録する場合、一意性の制御が行われ、同じ主キー値のツアーの存在は許可されません。

外部キーインデックスが利用されるケースは、Country トランザクションからデータを削除しようとした場合や、TouristGuide トランザクションからデータを削除しようとした場合です。

これは、削除しようとした主キーの値が、TRIP テーブルで利用されたレコードがないことを確認するために利用されます。

また、Trip トランザクションで、データの登録や、更新を行う場合、CountryId に入力された値が、存在する値であるかは、COUNTRY テーブルの主キーインデックスを活用し、確認し、TouristGuideId に入力された値は、TOURISTGUIDE テーブルの主キーインデックスを活用し、確認します。

ケーススタディ: トランザクション設計



最後に、Customer トランザクションを確認します。
このトランザクションは 2 つのレベルで定義されたトランザクションであり、
第 1 レベルが Customer、第 2 レベルが Trip となります。

第 1 レベルには、主キー項目属性として、CustomerId があり、従属する項目属性
として、CustomerName があります。

CustomerTripsQty は、グローバル式として定義されています。

第 2 レベルには、主キー項目属性であり、外部キーでもある TripId があり、従属する
項目属性として、CustomerTripMiles があります。

TripDate、TripPrice、CountryId、CountryName は、外部キーに基づき、推論される
項目属性です。

CountryId については、Trip トランザクションでは、外部キーでしたが、
Customer トランザクションの第 2 レベル Trip では、TripId によって推論される
項目属性となります。

TripId が第 2 レベルの主キーになっているため、Customer トランザクションと
Trip トランザクションは N 対 M の関係になります。

ケーススタディ: 正規化されたテーブル

COUNTRY

Country X	
Structure	Indexes
名前	タイプ
Country 構造	
• CountryId	Numeric(4,0)
• CountryName	VarChar(40)
Structure	Indexes
項目属性	順序
Country インデックス	
• ICountry	主キー
• CountryId	昇順

TOURISTGUIDE

TouristGuide X	
Structure	Indexes
名前	タイプ
TouristGuide 構造	
• TouristGuideId	Numeric(4,0)
• TouristGuideName	VarChar(40)
Structure	Indexes
項目属性	順序
TouristGuide インデックス	
• ITouristGuide	主キー
• TouristGuideId	昇順

TRIP

Trip X	
Structure	Indexes
名前	タイプ
Trip 構造	
• TripId	Numeric(4,0)
• TripDate	Date
• TripPrice	Numeric(4,0)
• TouristGuideId	Numeric(4,0)
• CountryId	Numeric(4,0)
Structure	Indexes
項目属性	順序
Trip インデックス	
• ITrip	主キー
• TripId	昇順
• ITrip1	外部キー
• CountryId	昇順
• ITrip2	外部キー
• TouristGuideId	昇順

CUSTOMER

Customer X	
Structure	Indexes
名前	タイプ
Customer 構造	
• CustomerId	Numeric(4,0)
• CustomerName	VarChar(40)
• 論理項目属性	
• CustomerTripsQty	Numeric(4,0)
Structure	Indexes
項目属性	順序
Customer インデックス	
• ICustomer	主キー
• CustomerId	昇順

CUSTOMERTRIP

CustomerTrip X	
Structure	Indexes
名前	タイプ
CustomerTrip 構造	
• CustomerId	Numeric(4,0)
• TripId	Numeric(4,0)
• CustomerTripMiles	Numeric(4,0)
Structure	Indexes
項目属性	順序
CustomerTrip インデックス	
• ICustomerTrip	主キー
• CustomerId	昇順
• TripId	昇順
• ICustomerTrip1	外部キー
• TripId	昇順
• ICustomerTrip2	外部キー
• CustomerId	昇順

GeneXus により、Customer トランザクションから、CUSTOMER テーブル、CUSTOMERTRIP テーブルが作成されました。

これは、レベルごとに 1 つテーブルが作成されるためです。

スライドにその構造を追加しました。

CUSTOMER テーブルが、第 1 レベルに対応しています。

主キーとして、CustomerId があり、従属する項目属性として、CustomerName が含まれています。

「論理項目属性」というノードがあり、CustomerTripQty が含まれています。

これは、CustomerTripQty が自動で値を算出する項目属性であるため、論理的にテーブルに属していることを表し、物理的な列は作成されないことを表しています。

CUSTOMERTRIP テーブルが、第 2 レベルに対応しています。

テーブル名は、第 1 レベル + 第 2 レベルの名前に基づきます。

主キーは、CustomerId と TripId の複合キーとなり、CustomerTripMiles が従属する項目属性として含まれています。

主キーが複合キーとなる理由としては、第 1 レベルの特定のレコードに紐づき、複数のレコードを登録できるようにするためです。

また、TripId は、外部キーでもあるため、この値に基づき、推論できる TripDate、TripPrice、CountryId、CountryName は含まれません。

では、インデックスはどうなるでしょうか。

CUSTOMER テーブルに対して作成されるインデックスは、CustomerId に基づく主キーインデックスのみです。

CUSTOMERTRIP テーブルに対して作成されるインデックスは、CustomerId、TripId に基づく主キーインデックスと、TripId、CustomerId それぞれに基づく外部キーインデックスです。

トランザクションの定義でも説明していましたが、Customer トランザクションの第 2 レベルである Trip に対する CUSTOMERTRIP テーブルには、CountryId は

物理的な列として作成されず、外部キーインデックスも作成されませんでした。
これは、TripId の値に基づき、推論される項目であるためです。

トランザクション設計とテーブル構造の対比

トランザクション設計

Country
CountryId*
CountryName

TouristGuide
TouristGuideId*
TouristGuideName

Trip
TripId*
TripDate
TripPrice
TouristGuideId
TouristGuideName
CountryId
CountryName

Customer
CustomerId*
CustomerName
CustomerTripsQty
Trip
(
 TripId*
 TripDate
 TripPrice
 CountryId
 CountryName
 CustomerTripMiles
)

Country X

Structure | Indexes

名前	タイプ
Country 構造	
CountryId	Numeric(4.0)
CountryName	VarChar(40)

Trip X

Structure | Indexes

名前	タイプ
Trip 構造	
TripId	Numeric(4.0)
TripDate	Date
TripPrice	Numeric(4.0)
TouristGuideId	Numeric(4.0)
CountryId	Numeric(4.0)

Customer X

Structure | Indexes

名前	タイプ
Customer 構造	
CustomerId	Numeric(4.0)
CustomerName	VarChar(40)
CustomerTripsQty	Numeric(4.0)

CustomerTrip X

Structure | Indexes

名前	タイプ
CustomerTrip 構造	
CustomerId	Numeric(4.0)
TripId	Numeric(4.0)
CustomerTripMiles	Numeric(4.0)

TouristGuide X

Structure | Indexes

名前	タイプ
TouristGuide 構造	
TouristGuideId	Numeric(4.0)
TouristGuideName	VarChar(40)

正規化されたテーブル

正規化されたテーブル

ここまでの内容の通り、GeneXus では、上に示すトランザクション設計から、下に示す正規化された構造がデータベースに作成されることを確認しました。

*GeneXus*TM

training.genexus.com
wiki.genexus.com