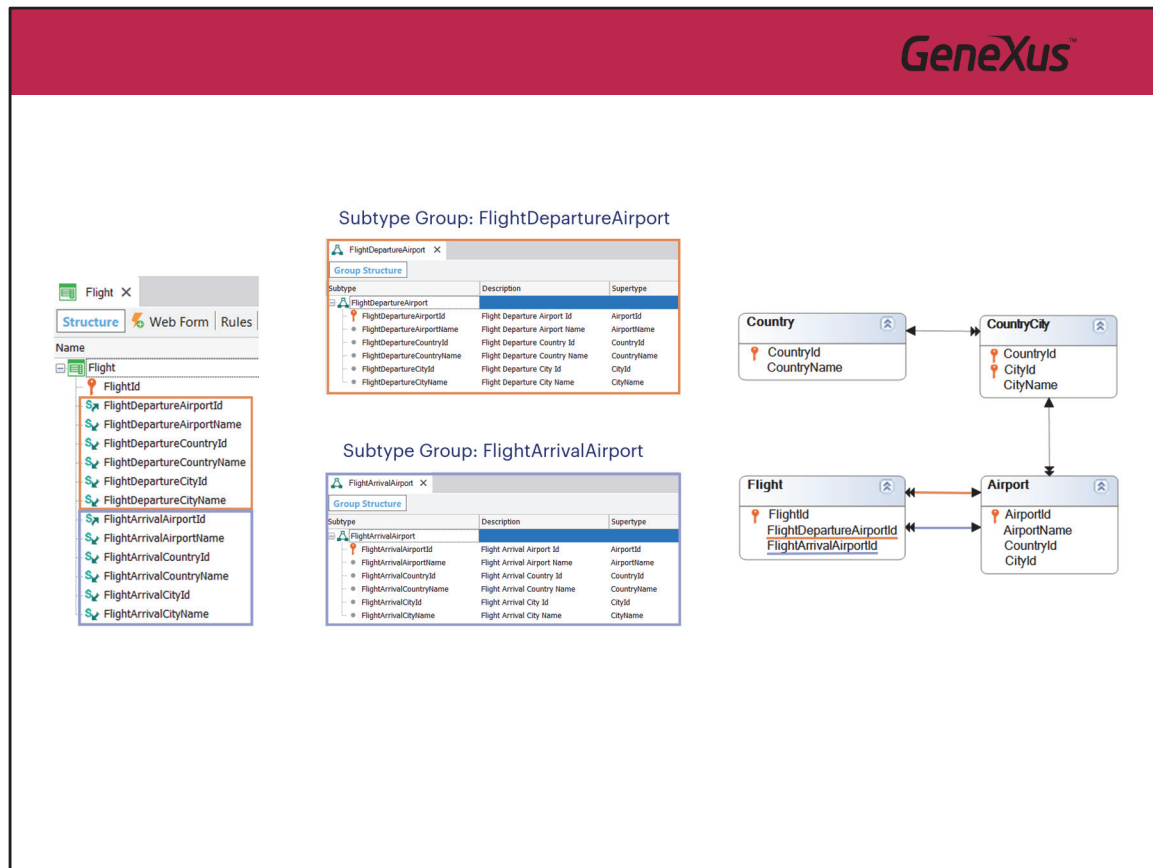


サブタイプ

複数参照とスペシャライゼーション

GeneXus[™]



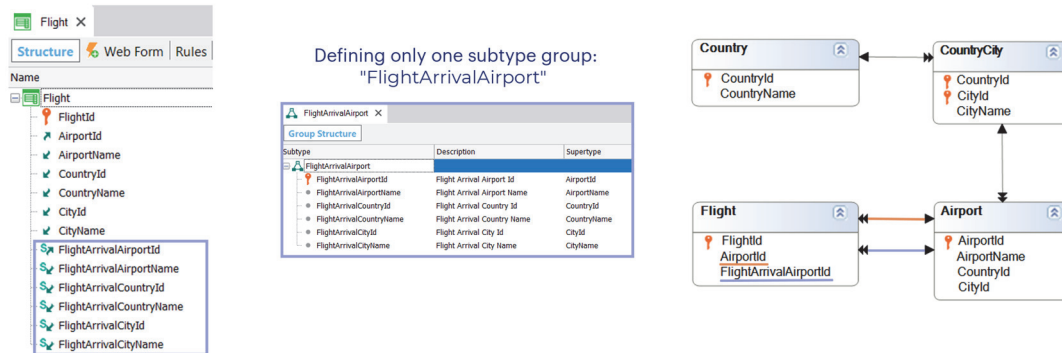
前の章で見たとおり、同じ現実のエンティティへの二重参照がトランザクションにある場合はサブタイプのグループを定義する必要があります。

Flight トランザクションの場合、フライトごとに出発空港と到着空港があります。トランザクション構造に同じ項目属性 AirportId を 2 回含めることはできず、GeneXus で重複する項目属性名の追加に対してエラーが表示されました。このため、サブタイプのグループを 2 つ定義することにしました：出発空港を指定する「FlightDepartureAirport」と、到着空港を指定する「FlightArrivalAirport」です。

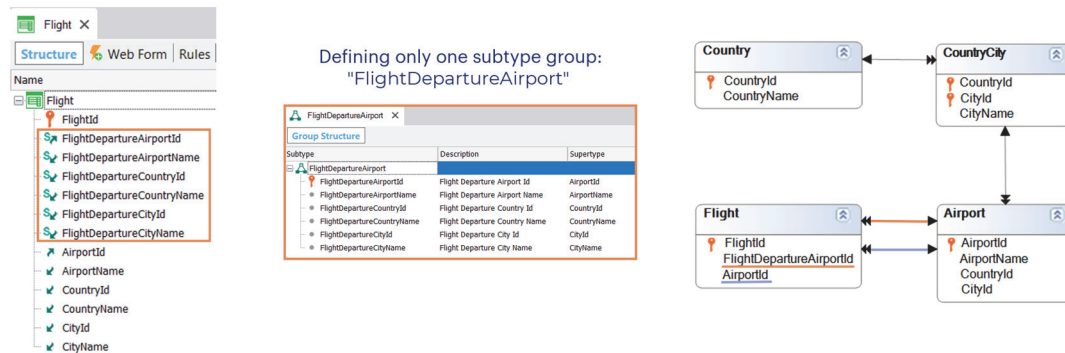
各空港の国と都市を推論するため、各グループ内では、空港名に加え、国と都市に対応する項目属性のサブタイプも定義しました。

このため、Flight トランザクションで FlightDepartureCountryName を指定すると、出発空港 FlightDepartureAirportId から推論される CountryName になります。これらのサブタイプは同じグループ内で定義されているため、その関連付けと関係が確立されています。

Flight トランザクションで FlightArrivalCountryName を指定すると、FlightArrivalAirportId 項目属性から推論されることが分かっています。つまり、ここにあいまいさはありません。Flight から Country に到達するには、まったく異なる 2 とおりの方法があります。



ほかに、AirportId 項目属性を出発空港に使用し、到着空港を指定するサブタイプのグループを定義する方法があります。



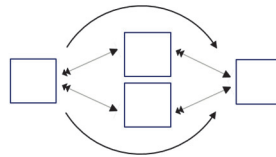
または、AirportId 項目属性を到着空港に使用し、出発空港を指定するサブタイプのグループを定義することもできます。
いずれの場合も、データモデルには前述の方法と同じ関係が反映されます。

MULTIPLE REFERENCES

Direct



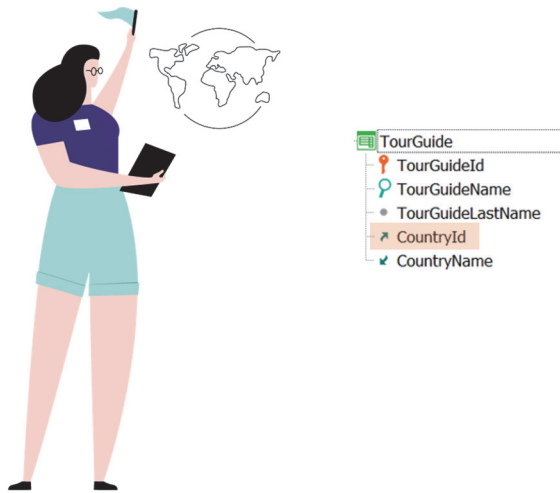
Indirect



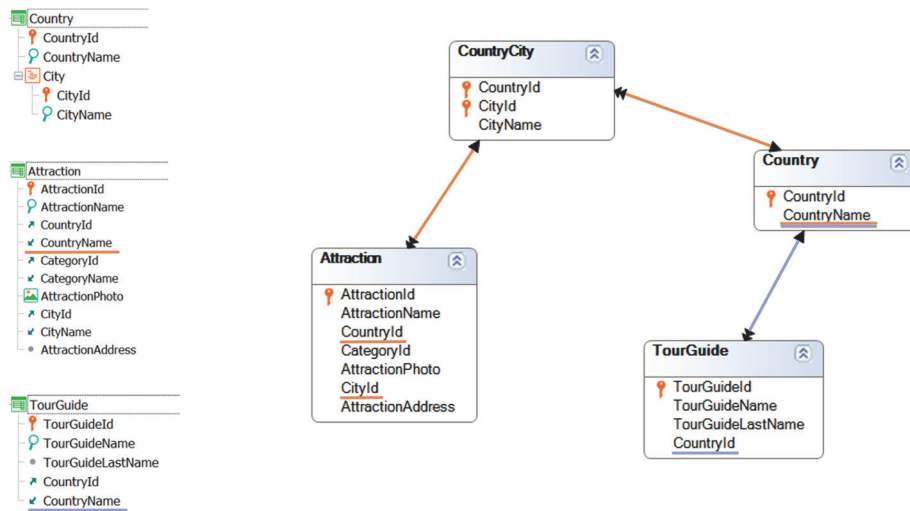
前の章では、1 つのテーブルから、直接関係がある別のテーブルへの複数参照のケースも確認しました。

しかし、こうした参照は直接的である必要はありません。

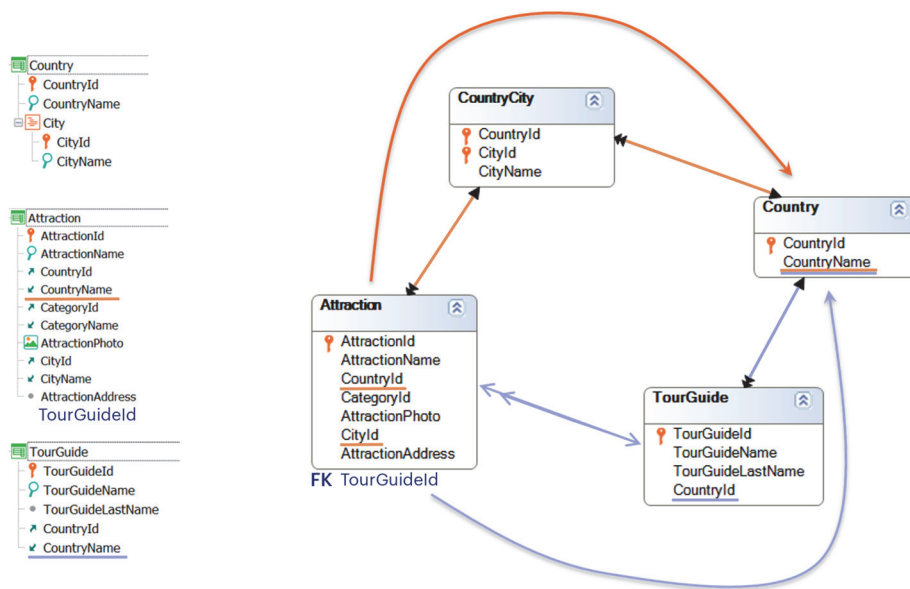
1 つのテーブルから別のテーブルへ 2 とおりのパスがある場合もあります。この場合、間接的な関係となるため、サブタイプでパスを区別する必要があります。



ツアーガイドの情報を記録するトランザクションを追加するとします。各ツアーガイドには国籍があるため、トランザクション構造に CountryId 項目属性を追加します。



テーブルダイアグラムを見ると、Attraction トランザクションから、その観光名所に対応する CountryName を推論できます。この情報は拡張テーブルにあります。同様に、TourGuide から CountryName、つまりツアーガイドの国を推論できます。



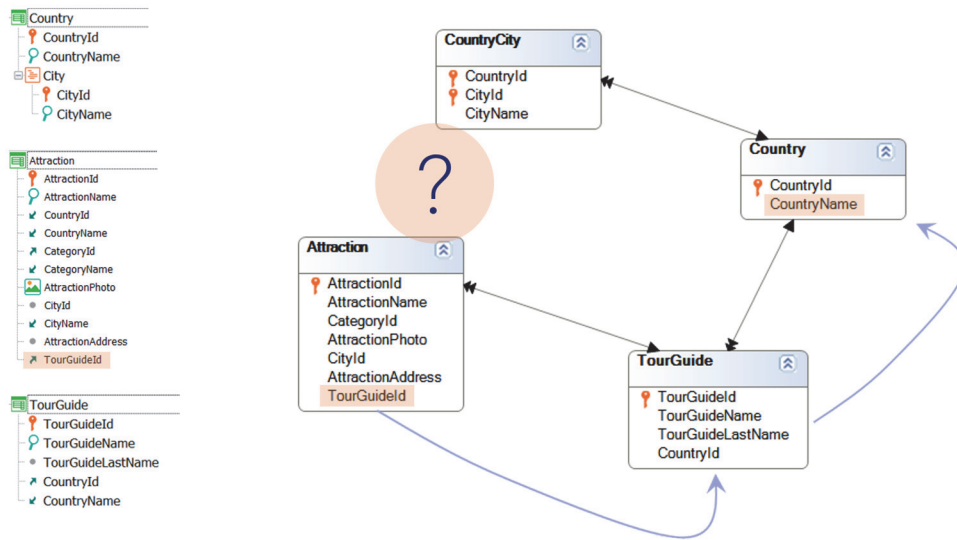
Attraction と TourGuide のエンティティをリンクするために、Attraction に、ツアーガイドを識別する項目属性 TourGuideId を追加するとします。観光名所にはそれぞれ 1 人のツアーガイドが割り振られていると想定します。この場合、テーブル間の関係はどのようになるでしょうか。

TourGuideId は Attraction 内で TourGuide テーブルへの外部キーとなります。この関係を水色の矢印で示します。この場合、Attraction から 2 とおりの方法で CountryName を推論できることになり、あいまいさが生じることが分かります。

Attraction の CountryName 項目属性はどこから推論されるのでしょうか。観光名所の都市と、観光名所のツアーガイドのどちらでしょうか。いずれの値も同じだった場合は問題になりませんが、この例では一致するとは限りません。観光名所がある国が、ツアーガイドの母国と一致する必要はありません。

これらの CountryName ロールを区別するためには、サブタイプを使用する必要があります。

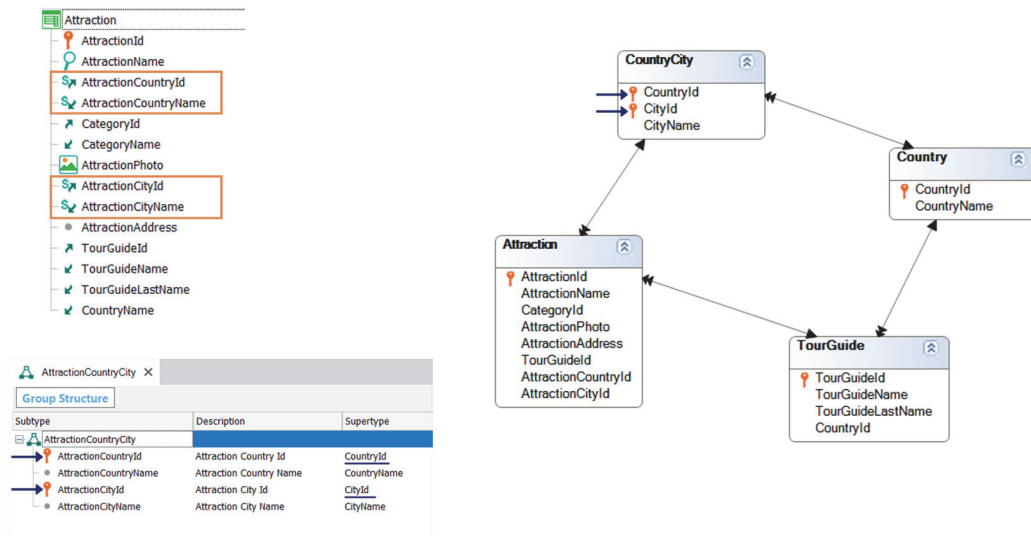
ただし、この場合、必要なのはロールの区別だけではありません。



Attraction トランザクションに TourGuideId 項目属性を追加した後のテーブルダイアグラムを見てみます。CountryId が Attraction の外部キーではなくなるため、Attraction と CountryCity の関係がなくなっています。テーブルからなくなっており、推論される項目属性となります。それでは、何から推論されるのでしょうか。それは TourGuideId です。

GeneXus では最初にテーブルが正規化されます。つまり、項目属性名と識別子に基づき、各テーブルに入れる項目属性とその関係が判断されます。TourGuide の識別子である TourGuideId が Attraction 内にあり、それと同時に Country の識別子である CountryId が TourGuide 内にあるため、Attraction の TourGuideId から TourGuide テーブルを経由して CountryName を推論できると理解されます。

このトランザクション設計では、観光名所の国を指定できず、CountryName はツアーガイドの国になります。ツアーガイドの国とは別に Attraction の国を指定するには、サブタイプを使用するしか方法はありません。



問題が発生するのは Attraction であるため、観光名所の国と都市を表すサブタイプのグループを作成します。

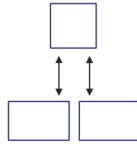
今度は、テーブルダイアグラムで関係が正しく表されています。CountryName 項目属性は TourGuideId から推論され、あいまいさはありません。観光名所の国を表し、推論に使用される項目属性は AttractionCountryName となります。これは AttractionCountryCity グループに属する CountryName のサブタイプです。

また、このグループには 2 つの主項目属性があります: AttractionCountryId と AttractionCityId です。これらは、スーパータイプ CountryId と CityId に従い、CountryCity テーブルの主キーに対応しています。

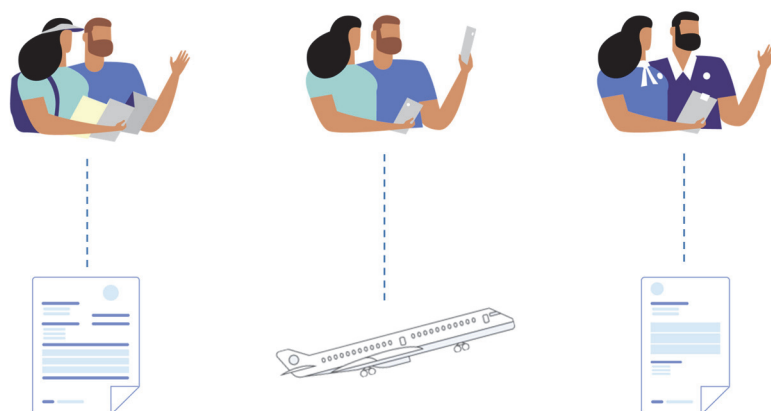
この方法で問題は解決できますが、元の項目属性 CountryId と CountryName を使用して観光名所の国を検索するオブジェクトがほかにある場合は、サブタイプの項目属性 AttractionCountryId と AttractionCountryName をそれぞれ使用するようオブジェクトを修正する必要があることに注意してください。

ほかにも考えられる解決策はありますが、このコースでは取り上げません。

SPECIALIZATION

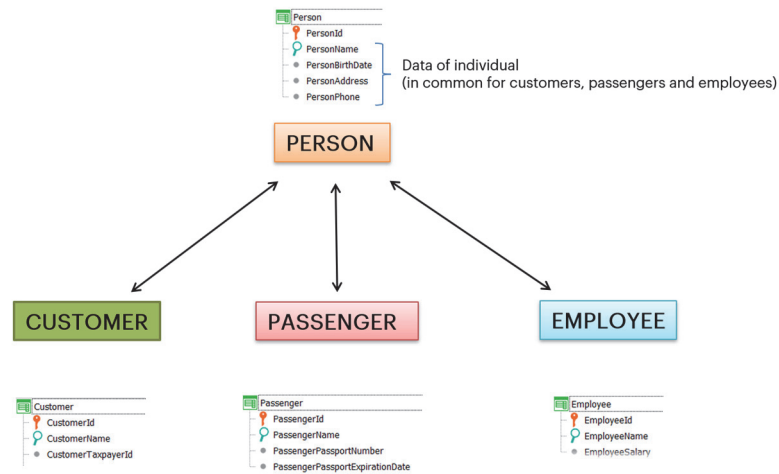


ここで、サブタイプを使用する別の例として、スペシャライゼーションについて説明します。全般的な情報を記録するトランザクションと、具体的な情報を記録するトランザクションがある場合、後者は前者を特化 (スペシャライゼーション) したものととなります。



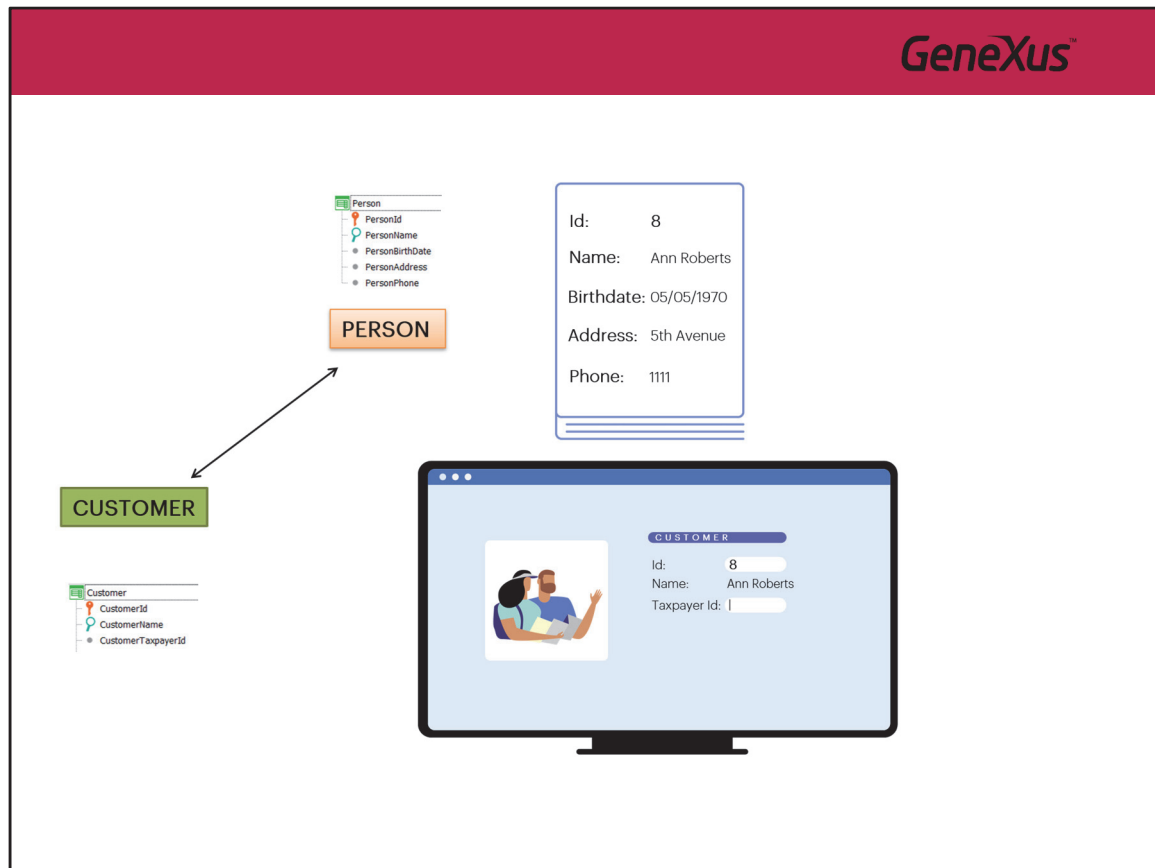
旅行代理店が、航空券やツアーパッケージを販売する顧客に関する詳細情報（納税者番号など）、搭乗客に関する詳細情報（パスポート番号や有効期限など）、代理店の従業員に関する詳細情報（給与など）を扱う必要があるとします。

代理店は顧客に料金を請求し、搭乗客に座席を予約し、従業員に給与明細書を発行します。

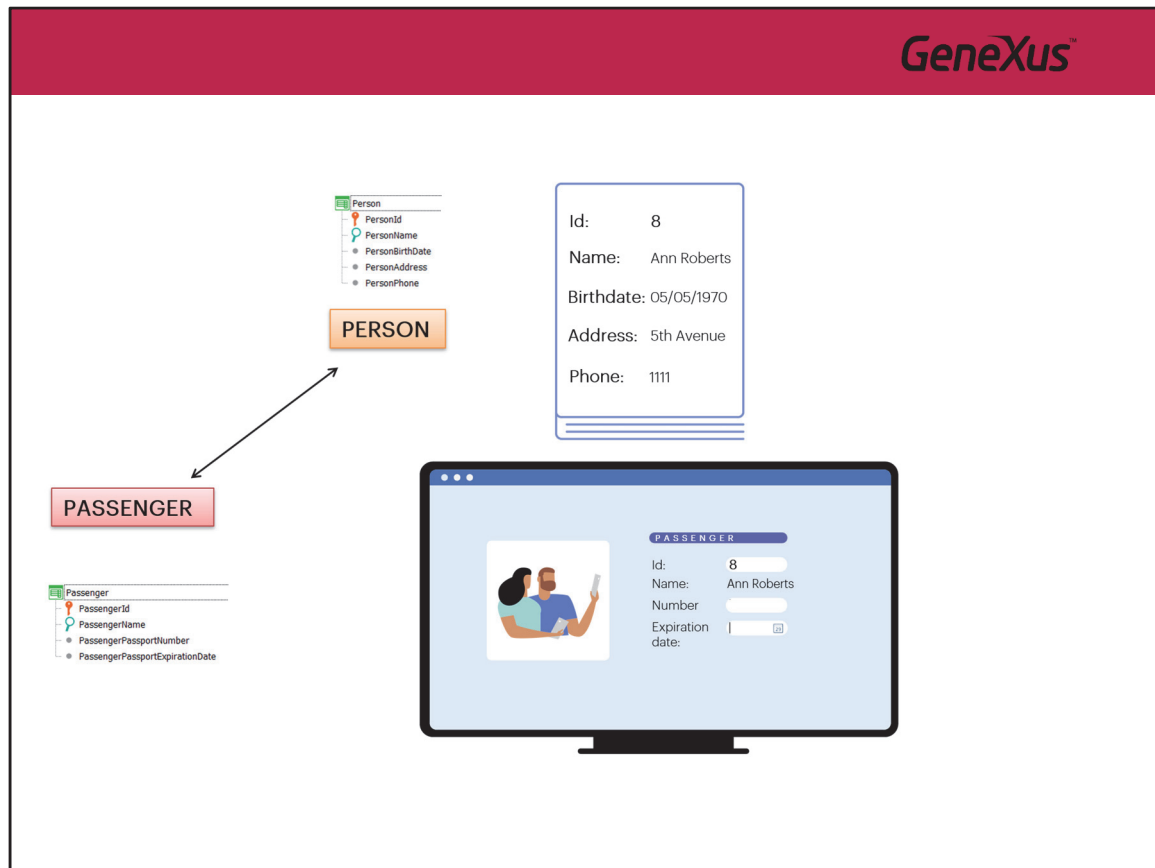


この場合、Customer トランザクションのみを定義するのではなく、すべての人々に共通する情報 (名前、生年月日、住所、電話番号など) を扱う Person というトランザクションと、Person のスペシャライゼーションとなるトランザクションを定義できます。顧客、搭乗客、従業員はすべて人 (Person) だからです。

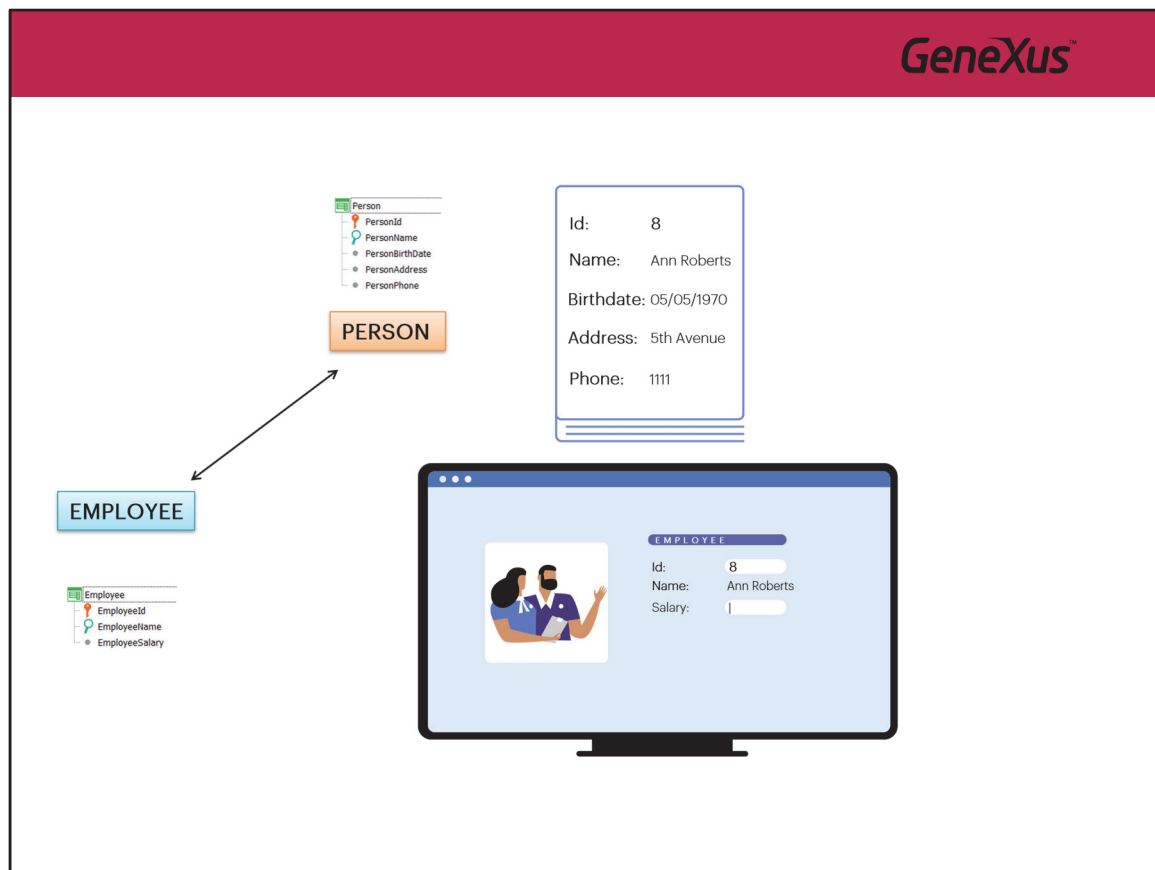
スペシャライゼーションはそれぞれ独自のデータを持つことになります (Customer は納税者番号、Passenger はパスポート番号と有効期限、Employee は給与)。



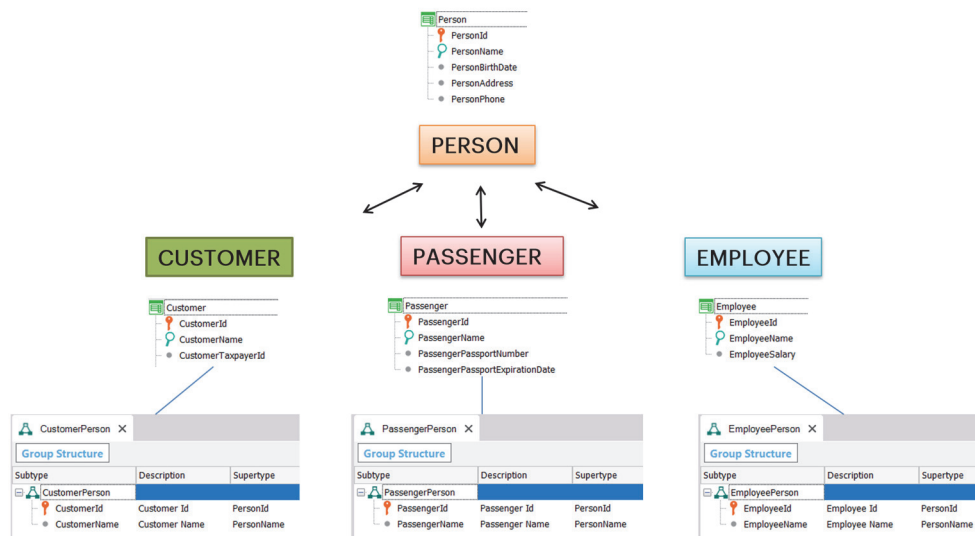
顧客が人であることを反映して、顧客 ID は人と**完全一致**するようにします。つまり、ID 8 の人が Ann Roberts で、1970 年 5 月 5 日生まれだった場合、ユーザーが顧客としてこの人の情報を入力するときに、Customer トランザクションで ID 8 を入力できるようにします。また、このフィールドから CustomerTaxpayerId 項目属性に移動し、納税者番号を入力する場合は、名前 Ann Roberts が表示されるようになります。



同様に Passenger トランザクションの実行時には、ユーザーが値 8 を PassengerId に入力すると、PassengerName に Ann Roberts が推論されるようにします。また、ユーザーが対応する項目属性 (PassengerPassportNumber と PassengerPassportExpirationDate) にパスポート番号と有効期限を入力できるようにします。



従業員についても同様です。



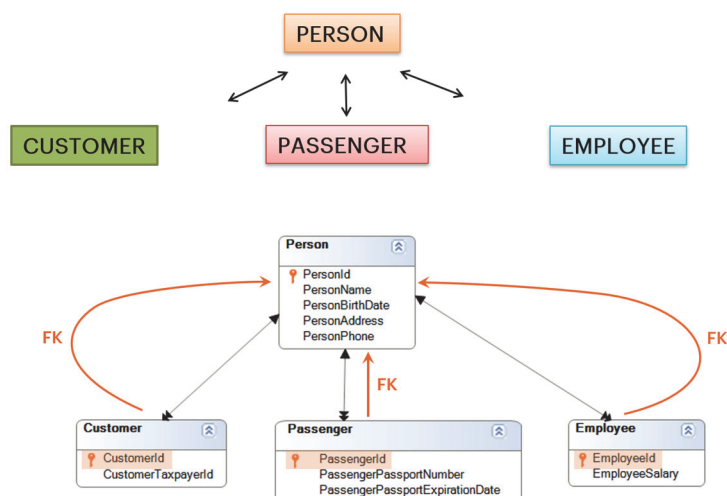
Customer、Passenger、Employee の主キーとして、PersonId に関連付けることなく CustomerId、PassengerId、EmployeeId の各項目属性を定義 (たとえば PersonName に関連付けることなく CustomerName、PassengerName、EmployeeName を定義) するだけでは想定どおりになりません。GeneXus にとって、これらは完全に独立したトランザクションとなります。

これらに関連付けるには、サブタイプのグループを定義し、顧客、搭乗客、従業員が有効な (登録済みの) 人であることを指定します。

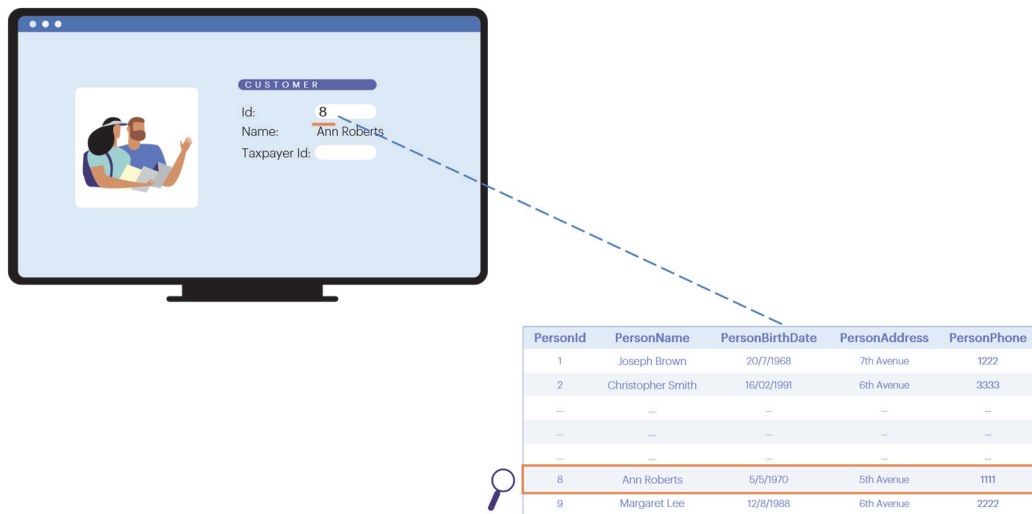
CustomerPerson グループを作成し、CustomerId と CustomerName をそれぞれ PersonId と PersonName のサブタイプとして定義します。

次に PassengerPerson グループを作成し、PassengerId と PassengerName をそれぞれ PersonId と PersonName のサブタイプとして定義します。

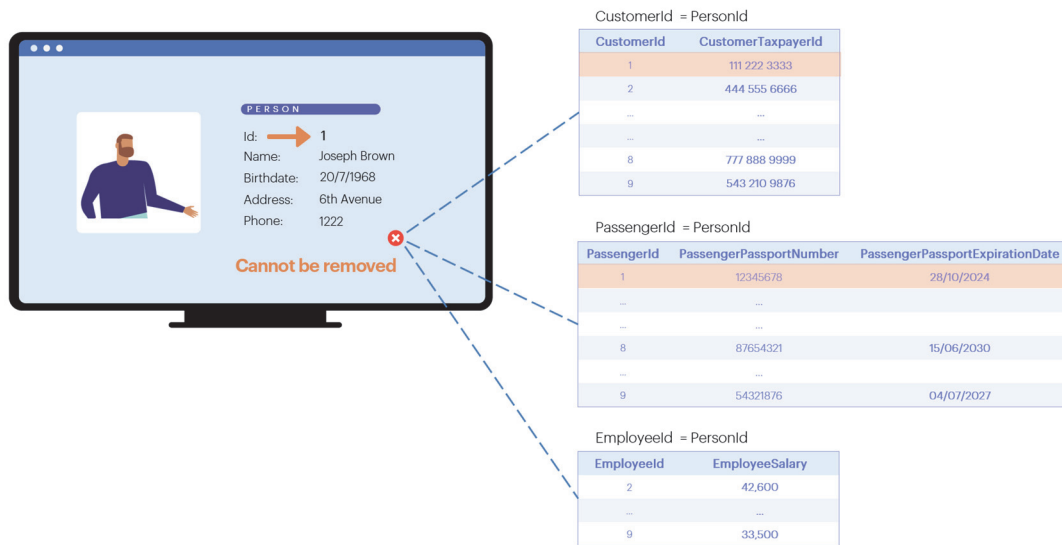
最後に EmployeePerson グループを作成し、EmployeeId と EmployeeName をそれぞれ PersonId と PersonName のサブタイプとして定義します。



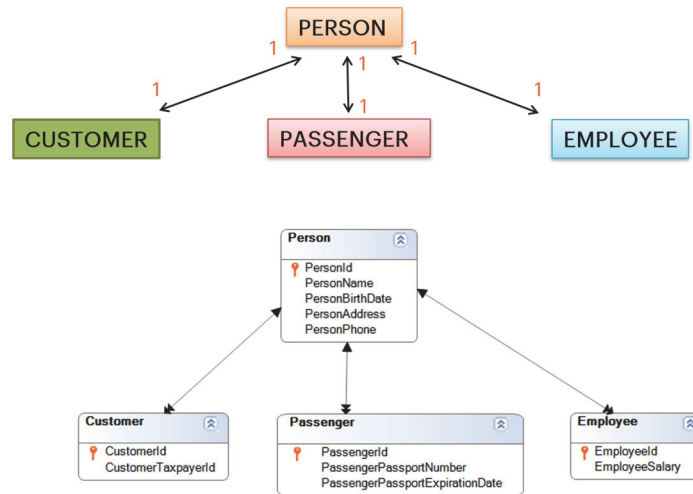
この方法で、CustomerId、PassengerId、EmployeeId の各項目属性が Customer、Passenger、Employee の各テーブルの識別子となり、結果的にその主キーとなるだけでなく、Person テーブルの外部キーとなります。この場合、データの整合性は GeneXus によって制御されます。



ユーザーが 3 つのトランザクション (Customer、Passenger、Employee) のいずれかで ID の値を入力すると、同じ ID 値のレコードが Person テーブル内で検索されます。



同様に、Person トランザクションで誰かを削除する場合は、削除しようとしている PersonId と CustomerId が一致するレコードが Customer 内にないことや、削除しようとしている PersonId と PassengerId が一致するレコードが Passenger 内にないこと、EmployeeId が PersonId と一致するレコードが Employee 内にないことが確認されます。3 つのいずれかのレコードが存在する場合は、その人を削除できません。



この設計は、全般テーブルと、それぞれのスペシャライゼーションに対応するテーブルの 1 対 1 の関係を表しています：

1 人の人物は顧客として 1 回のみ登録できます。CustomerId は有効な PersonId であり、主キーでもあるからです。同様に、搭乗客または従業員としても 1 回のみ登録可能です。

1 人の人物に 3 つのロールをすべて設定することができます。また、人としてのみ登録し、代理店の顧客、従業員、搭乗客としての追加データがなくてもかまいません。

Customer、Passenger、Employee の各トランザクションでは、CustomerName、PassengerName、EmployeeName は推論される項目属性となるため、それぞれ Customer、Passenger、Employee の各テーブルに物理的に保存されません。

GeneXus で作成されるダイアグラムでは 1 対 1 の関係は示されず、矢印は外部キーの関係のみを示します。このため、スペシャライゼーションしたテーブル名の横に二重矢印があります。

ほかにもサブタイプの使用方法是ありますが、このコースでは取り上げません。関心がある場合は、次のレベルのコースでこのトピックに関連する情報を参照してください。



動画	https://www.genexus.com/community-and-support-jp/training?ja
ドキュメント	http://wiki.genexus.jp/
認定資格	training.genexus.com/certifications