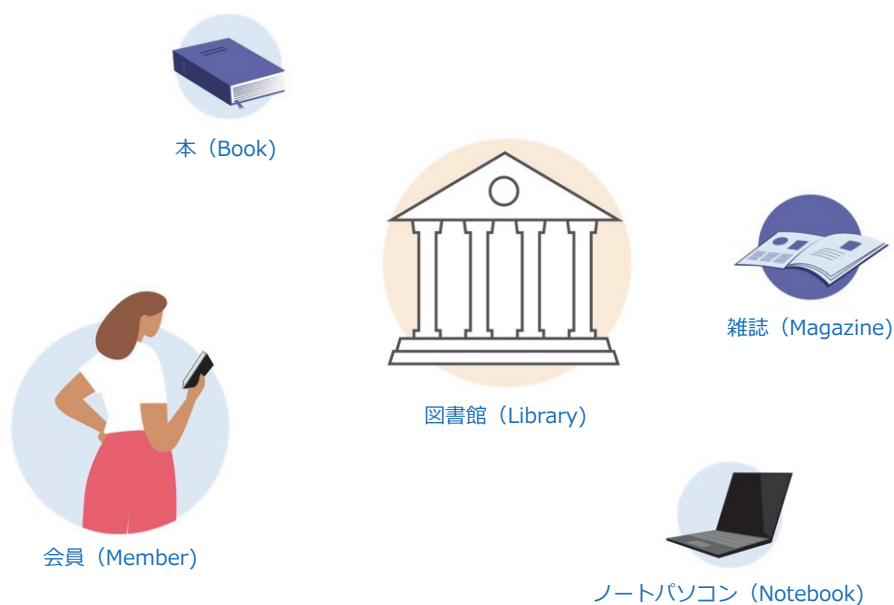


トランザクション設計モデルの分析

題材：図書館

GeneXus™

図書館の概要



ここまでの章を通じて、GeneXus で特定の現実を正しくデータモデル化するために必要な実装方法をすべて確認しました。

この章では、提示された現実の分析を通じて、必須のリソース群を特定し、一連の実際の要件を解決します。

この中で、トランザクションを設計するためのさまざまなオプションを分析します。

では、題材として、本、雑誌、ノートパソコンの貸し出しに関連する図書館のタスクを管理するために、GeneXus アプリケーションを設計する必要があるとします。

これは、本や雑誌などのさまざまな読み物や、図書館から貸し出されるノートパソコン、これらを利用できる会員と連携しています。

必要となるエンティティ①



国
(Country)

- ・管理番号
- ・名前



会員
(Member)

- ・会員番号
- ・名前
- ・住所
- ・写真
- ・電話番号
- ・20 歳以上であること



著者
(Author)

- ・管理番号
- ・名前
- ・写真
- ・出身国
- ・著者は複数の本を出版する

現時点で図書館から開示されている記録が必要な情報は以下のとおりです。

国 (Country)

それぞれの国について、一意の識別子と国名を登録します。

会員 (Member)

それぞれの会員について、会員番号、名前、住所、写真、電話番号を登録します。

ただし、会員は 20 歳以上でなければ登録できません。

著者 (Author)

それぞれの著者について、一意の識別子、名前、写真、出身国を登録します。

1 人の著者が複数の本を出版している場合があり、このような現実を表せる必要があります。

必要となるエンティティ②



国
(Country)

- ・管理番号
- ・名前



本
(Book)

- ・管理番号
- ・タイトル
- ・発行日
- ・冊数
- ・文学ジャンル
- ・著者
- ・出版社



会員
(Member)

- ・会員番号
- ・名前
- ・住所
- ・写真
- ・電話番号
- ・20 歳以上であること



雑誌
(Magazine)

- ・管理番号
- ・タイトル
- ・発行日
- ・表紙の画像
- ・冊数



著者
(Author)

- ・管理番号
- ・名前
- ・写真
- ・出身国
- ・著者は複数の本を出版する



ノートパソコン
(Notebook)

- ・管理番号
- ・写真
- ・簡易説明
- ・貸出状態

本 (Book)

それぞれの本について、一意の識別子、タイトル、発行日、図書館で利用可能な冊数を登録します。

小説、エッセイ、詩集などの文学のジャンルに本は属します。

さらに、著者と出版社についての情報も記録します。

雑誌 (Magazine)

それぞれの雑誌について、一意の識別子、タイトル、発行日、表紙の画像、利用可能な冊数を登録します。

ノートパソコン (Notebook)

図書館のサービスには、会員へのノートパソコンの貸し出しが含まれます。会員の多くは作家や研究者であるためです。それぞれのノートパソコンについて、一意の識別子、写真、簡単な説明を登録します。

さらに、そのステータス (利用可能か貸し出されているか) も記録します。

必要となるエンティティ③



国
(Country)

- ・管理番号
- ・名前



本
(Book)

- ・管理番号
- ・タイトル
- ・発行日
- ・冊数
- ・文学ジャンル
- ・著者
- ・出版社



貸出
(Loan)

- ・管理番号
- ・貸出日 (固定値)
- ・会員
- ・返却日 (自動決定)
- ・貸出物へのコメント
- ・貸出期間は固定 15 日
- ・1 度の貸し出し最大数は、本は 3 冊、雑誌は 4 冊
- ・ノートパソコンは貸し出す場合もそうでない場合もある



会員
(Member)

- ・会員番号
- ・名前
- ・住所
- ・写真
- ・電話番号
- ・20 歳以上であること



雑誌
(Magazine)

- ・管理番号
- ・タイトル
- ・発行日
- ・表紙の画像
- ・冊数



著者
(Author)

- ・管理番号
- ・名前
- ・写真
- ・出身国
- ・著者は複数の本を出版する



ノートパソコン
(Notebook)

- ・管理番号
- ・写真
- ・簡易説明
- ・貸出状態

貸出 (Loan)

図書館としてのメインタスクは、会員に本、雑誌、ノートパソコンを貸し出すことです。

それぞれの貸し出しについて、一意の識別子、貸出日、対象の会員、返却日を記録します。

貸出日は常に記録時の現在の日付となり、変更は不可、返却日は「貸出期間は一律 15 日」というルールに基づき、自動的に決定する必要があります。

1 度の貸し出しには、最大で本が 3 冊、雑誌が 4 冊含めることができ、ノートパソコンの貸し出しは最大 1 台で、含まれる場合と含まれない場合があります。本および雑誌は、一度の貸出において同じものは 1 冊だけ貸し出すことができます。各本および雑誌は、貸出において、必要と思われるコメント (表紙の損傷やページの欠落など) を入力することができます。

ドメインの定義



国
(Country)

- ・管理番号
- ・名前



会員
(Member)

- ・会員番号
- ・名前
- ・住所
- ・写真
- ・電話番号
- ・20 歳以上であること



著者
(Author)

- ・管理番号
- ・名前
- ・写真
- ・出身国
- ・著者は複数の本を出版する

名前	タイプ	モジュール	デスクリプション
ドメイン			
Id	Numeric(4.0)	Root Module	Id
Name	VarChar(20)	Root Module	Name

プロパティ	
Domain: Id	
Name	Id
Description	Id
Nulls in Forms	Empty as Null
～一部省略～	
Collection	False
Autonumber	True
Autonumber start	1
Autonumber step	1

この現実の分析し、実装を始めます。

国や、会員、著者は、単純なエンティティとして明確に区別できます。
このような構造から定義を開始していきます。

しかし、ほとんどのエンティティには、自動採番が可能な一意の識別子を記録するという点などを考慮し、準備をします。
一部の項目では、共通のデータタイプを利用できることが想定されるため、あらかじめドメインを定義します。

Id ドメインを定義し、[Autonumber] プロパティを True に設定します。

また、Name ドメインを VarChar(20) として定義します。

トランザクションの定義: Country



・管理番号
・名前

Country X				
Structure Web Layout Rules Events Variables Help Documentation Patterns				
名前	タイプ	デスクリプション	式	Null 許容
Country	Country	国		
CountryId	Id	国番号		No
CountryName	Name	国名		No

Country X	
Structure Indexes	
項目属性	順序
Country インデックス	
ICountry	主キー
CountryId	昇順
UCountry	一意
CountryName	昇順

国の情報を記録するため、Country トランザクションの定義から始めます。
 管理番号を入力する項目は、主キーとして **CountryId**、名前を入力する項目は **CountryName** のように定義しました。
 CountryId は、Id ドメインに基づき、CountryName は、Name ドメインに基づきます。
 記録できる情報について確認を行ったところ、重複した国名の登録はしたくない
 ということで、一意のインデックスも定義しています。
 同じように重複した値を登録させたくないケースで同様の定義を行うことが有効です。

都市（弱いエンティティに関する補足）



・管理番号
・名前

名前	タイプ	DESCRIPTION	式	Null 許容
Country	Country	国		
CountryId	Id	国番号		No
CountryName	Name	国名		No
City	City	都市		
CityId	Id	都市番号		No
CityName	Name	都市名		No

弱い 1 対 N

名前	タイプ	DESCRIPTION
CountryCity 構造		都市
CountryId	Id	国番号
CityId	Id	都市番号
CityName	Name	都市名

今回の要望には含まれていませんが、もし著者の出身国に加え、出身都市を必要とした場合、国に対して弱いエンティティとして都市を追加できます。弱いエンティティとなる理由としては、都市は必ず国が無ければ存在できないためです。

国に対する弱いエンティティを追加する場合、Country トランザクションへ第 2 レベルを追加します。City というレベルを追加しますが、これはトランザクションとして存在しないものです。

その理由は、CityId が単体で主キーとなるトランザクションが定義されていないためです。

著者の出身都市が必要な場合、弱いエンティティとなるため、CountryId の値と、CityIdの値が必要となります。

これは、第 2 レベル City に基づき、生成される CountryCity テーブルの主キー項目がこの 2 項目となっているためです。

前述の通り、今回都市の管理は不要なため、以降は City レベルは削除された状態で進みます。

トランザクションの定義: Member



- ・会員番号
- ・名前
- ・住所
- ・写真
- ・電話番号
- ・20 歳以上であること

名前	タイプ	デスクリプション	式	Null 許容
Member	Member	会員		
MemberDocument	Numeric(8.0)	会員番号		No
MemberName	Name	会員名		No
MemberAddress	Address, GeneXus	会員住所		No
MemberImage	Image	会員写真		No
MemberPhone	Phone, GeneXus	会員電話番号		No
MemberBirthDate	Date	会員生年月日		No
MemberAge	Numeric(3.0)	会員年齢	age(MemberBirthDate,today())	

名前	タイプ	デスクリプション	式	Null 許容
Member	Member	会員		
MemberDocument	Numeric(8.0)	会員番号		No
MemberName	Name	会員名		No
MemberAddress	Address, GeneXus	会員住所		No
MemberImage	Image	会員写真		No
MemberPhone	Phone, GeneXus	会員電話番号		No
MemberBirthDate	Date	会員生年月日		No
MemberAge	Numeric(3.0)	会員年齢	age(MemberBirthDate,today())	

会員の定義に進みます。図書館からの要望に沿って、次の項目属性を使用して Member トランザクションを定義します。

- MemberDocument : 自動採番しないため、8 桁の Numeric タイプ
- MemberName : Name ドメイン
- MemberAddress : Address セマンティックドメイン
- MemberImage : Image タイプ
- MemberPhone : Phone セマンティックドメイン

また、会員は 20 歳以上でなければならないため、生年月日と年齢の入力が必要です。年齢については、生年月日に基づき、自動的に計算されるべきです。したがって、トランザクション構造に次の項目属性も必要です。

- MemberBirthDate : Date タイプ
- MemberAge : 3 桁の Numeric タイプ

年齢の自動計算については、Age 関数を利用し、以下のグローバル式を記述することで対応できます。

age(MemberBirthDate, today())

&today 変数ではなく Today() 関数を使用する理由は、式の宣言では変数を使用できないためです。

20 歳未満の場合、登録できないようにするためには、次の Error ルールを定義します。
Error("会員は20歳以上が登録できます。") if MemberAge < 20;

この分析では、基本的なデータ入力を制御するためのルールの宣言は考慮されていない点に注意してください。

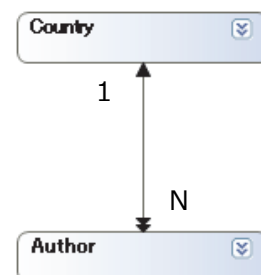
トランザクションの定義: Author



- ・管理番号
- ・名前
- ・写真
- ・出身国

・著者は複数の本を出版する

名前	タイプ	デスクリプション	式	Null 許容
Author	Author	著者		
AuthorId	Id	著者番号		No
AuthorName	Name	著者名		No
AuthorImage	Image	著者写真		No
CountryId	Id	国番号		No
CountryName	Name	国名		No



著者の定義に進みます。図書館からの要望に沿って、次の項目属性を使用して Author トランザクションを定義します。

- AuthorId : Id ドメイン
- AuthorName : Name ドメイン
- AuthorImage : Image タイプ

また、著者の出身国も記録する必要があるため、国の情報を紐づけます。
 そのため、定義済みの Country トランザクションの項目属性 CountryId と CountryName を追加します。
 CountryId は外部キーとなり、CountryName は外部キーから推論される項目です。
 この定義により、COUNTRY テーブルと、AUTHOR テーブルに 1 対 N の関係が出来上がります。

著者が複数の本を出版するという要件については、このトランザクションで満たす必要がないため、何も対応しません。

トランザクションの定義: Book



- ・管理番号
- ・タイトル
- ・発行日
- ・冊数
- ・文学ジャンル
- ・著者
- ・出版社

名前	タイプ	DESCRIPTION	式	Null 許容
Book	Book	本		
BookId	Id	本管理番号		No
BookTitle	VarChar(40)	本タイトル		No
BookEditionDate	Date	本発行日		No
BookCopies	Numeric(4,0)	本冊数		No
BookLiteraryGenre	GenreEnum	文学ジャンル		No

名前	DESCRIPTION	値
Novel	小説	Novel
Poetry	エッセイ	Poetry
Fiction	フィクション	Fiction
Article	記事	Article

本の定義に進みます。図書館からの要望に沿って、次の項目属性を使用して Book トランザクションを定義します。

- BookId : Id ドメイン
- BookTitle : VarChar タイプの 40 桁
- BookEditionDate : Date タイプ
- BookCopies : Numeric タイプの 4 桁

本に関する現実では、文学ジャンルも必要であると要望を受けていました。このジャンルには、小説やエッセイなどがありました。

文学ジャンルの概念をモデル化する場合、実装方法の一つに、列挙型ドメインの作成が思いつきます。この定義で十分でしょうか。

トランザクションの定義: Book と関連するトランザクション



- ・管理番号
- ・タイトル
- ・発行日
- ・冊数
- ・文学ジャンル
- ・著者
- ・出版社

名前	タイプ	DESCRIPTION
Book	Book	本
BookId	Id	本管理番号
BookTitle	VarChar(40)	本タイトル
BookEditionDate	Date	本発行日
BookCopies	Numeric(4.0)	本冊数
LiteraryGenreId	Id	文学ジャンル番号
LiteraryGenreName	Name	文学ジャンル名
AuthorId	Id	著者番号
AuthorName	Name	著者名
PublishingHouseId	Id	出版社番号
PublishingHouseName	Name	出版社名

名前	タイプ	DESCRIPTION
LiteraryGenre	LiteraryGenre	文学ジャンル
LiteraryGenreId	Id	文学ジャンル番号
LiteraryGenreName	Name	文学ジャンル名

項目属性	順序
LiteraryGenre インデックス	
ILiteraryGenre	主キー
ULiteraryGenre	一意
ULiteraryGenre	昇順

名前	タイプ	DESCRIPTION
PublishingHouse	PublishingHouse	出版社
PublishingHouseId	Id	出版社番号
PublishingHouseName	Name	出版社名

項目属性	順序
PublishingHouse インデックス	
IPublishingHouse	主キー
UPublishingHouse	一意
UPublishingHouse	昇順

列挙型ドメインを利用した場合、選択可能な文学ジャンルは、あらかじめ定義されたもののみです。

文学のジャンルは時代とともに変化、追加される可能性があり、この選択肢は最善ではありませんでした。

そのため、文学ジャンルを管理するため、次の項目属性を持つ LiteraryGenre トランザクションが必要と判断しました。

- LiteraryGenreId : Id ドメイン
- LiteraryGenreName : Name ドメイン

同じ名前の文学ジャンルの登録を防ぐために、ジャンル名に一意的インデックスを定義。この LiteraryGenre トランザクションで定義した 2 つの項目属性を Book トランザクションに追加します。

さらに、Book には、著者と出版社も登録する必要があります。Author は既にエンティティとして定義されているため、Author トランザクションの AuthorId と、AuthorName を追記します。

出版社も文学ジャンルと同様に固定ではなく、動的に追加、変更できる必要があると判断し、PublishingHouse のエンティティをモデル化する必要があると判断しました。

そこで、出版社を管理するため、次の項目属性を持つ PublishingHouse トランザクションを定義します。

- PublishingHouseId : Id ドメイン
- PublishingHouseName : Name ドメイン

出版社の名前にも、一意のインデックスを作成することで、同じ名前の出版社の登録を防げます。

Book トランザクションに、Author トランザクションの主キーを追加したことで、著者の要件に含まれていた著者が複数の本を出版するという要件も満たすことが出来ました。

トランザクションの定義: Magazine と Notebook



名前	タイプ	デスクリプション	式
Magazine	Magazine	雑誌	
MagazineId	Id	雑誌番号	
MagazineTitle	VarChar(40)	雑誌タイトル	
MagazinePublicationDate	Date	雑誌発行日	
MagazineImage	Image	雑誌表紙画像	
MagazineCopies	Numeric(4.0)	雑誌冊数	



名前	タイプ	デスクリプション
Notebook	Notebook	ノートパソコン
NotebookId	Id	ノートパソコン番号
NotebookImage	Image	ノートパソコン写真
NotebookDescription	LongVarChar(2M)	ノートパソコン簡易説明
NotebookStatus	Status	ノートパソコンステータス

名前	デスクリプション	値
OnLoan	貸出中	OnLoan
Available	利用可能	Available

雑誌の定義に進みます。図書館からの要望に沿って、次の項目属性を使用して Magazine トランザクションを定義します。

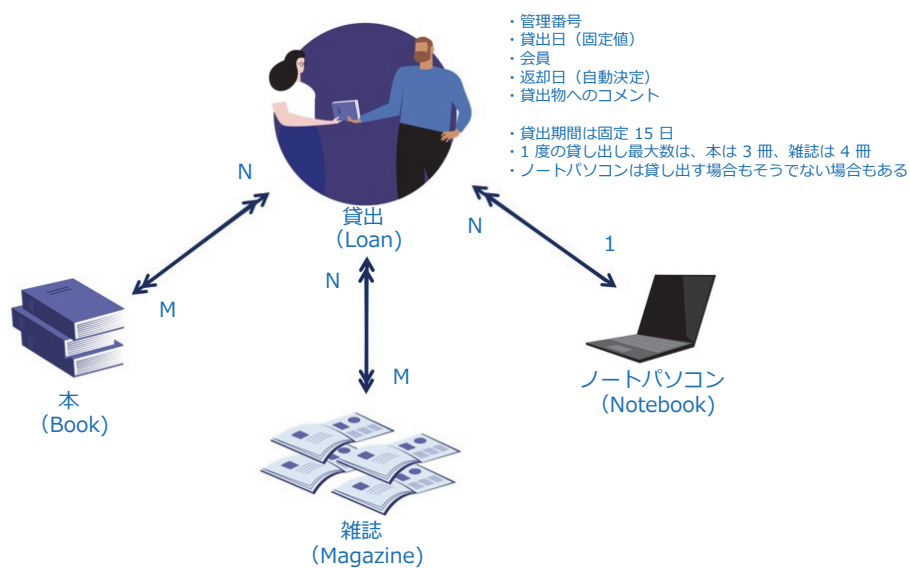
- MagazineId : Id ドメイン
- MagazineTitle : VarChar タイプの 40 桁
- MagazinePublicationDate : Date タイプ
- MagazineImage : Image タイプ
- MagazineCopies : Numeric タイプの 4 桁

ノートパソコンの定義に進みます。図書館からの要望に沿って、次の項目属性を使用して Notebook トランザクションを定義します。

- NotebookId : Id ドメイン
- NotebookImage : Image タイプ
- NotebookDescription : LongVarChar タイプの 2M 桁
- NotebookStatus : Status 列挙型ドメイン

ノートパソコンステータスは、「貸出中」か「利用可能」のみを選択肢とする必要があるため、Status という列挙型ドメインを作成し、割り当てます。

トランザクションの定義: Loan (分析)



次に、この現実において不可欠なタスクとなる貸出の概念を分析してみましょう。

この図書館では、会員は、ノートパソコン、本 3 冊、雑誌 4 冊を 15 日間借りることができます。

貸出とノートパソコンの間には 1 対 N の関係があり、貸出と本の間、および貸出と雑誌の間には N 対 M の関係があります。

これをモデル化するにはどうすればよいでしょうか。

トランザクションの定義: Loan (第1レベル)



Loan X				
Structure Web Layout Rules Events Variables Help Documentation Patterns				
名前	タイプ	デスクリプション	式	Null 許容
Loan	Loan	貸出		
LoanId	Id	貸出番号		No
LoanDate	Date	貸出日		No
MemberDocument	Numeric(8.0)	会員番号		No
MemberName	Name	会員名		
MemberAddress	Address, GeneXus	会員住所		
LoanReturnDate	Numeric(4.0)	返却日	<u>LoanDate.adddays(15)</u>	
NotebookId	Id	ノートパソコン番号		Yes
NotebookDescription	LongVarChar(2M)	ノートパソコン簡易説明		

分析結果に基づき、貸出の定義に進みます。図書館からの要望に沿って、Loan トランザクションを定義していきます。

初めに、以下の項目を定義しました。

- LoanId : Id ドメイン
- LoanDate : Date タイプ
- MemberDocument : 会員の外部キー
- MemberName : 会員の推論項目
- MemberAddress : 会員の推論項目
- LoanReturnDate : Date タイプ
- NotebookId : ノートパソコンの外部キー
- NotebookDescription : ノートパソコンの推論項目

返却日は、貸出日を基準に自動的に計算されるようにする必要があります。貸出期間は 15 日間であるため、LoanReturnDate 項目属性に関連付けられた次の計算を宣言します。
LoanDate.AddDays(15)

ノートパソコンについては、貸出に含まれる場合と含まれない場合があります。外部キーは、既定のままでは値を指定することが必須となります。そのため、[Null 許容] 列を Yes に設定します。

トランザクションの定義: Loan (Book レベル)



Loan				
名前	タイプ	デスクリプション	式	Null 許容
Loan	Loan	貸出		
LoanId	Id	貸出番号		No
LoanDate	Date	貸出日		No
MemberDocument	Numeric(8.0)	会員番号		No
MemberName	Name	会員名		
MemberAddress	Address, GeneXus	会員住所		
LoanReturnDate	Numeric(4.0)	返却日	LoanDate.adddays(15)	
NotebookId	Id	ノートパソコン番号		Yes
NotebookDescription	LongVarChar(2M)	ノートパソコン簡易説明		
LoanBooksQty	Numeric(4.0)	貸出本数	count(LoanBookComment)	
Book	Book	貸出本	Error("本は3冊まで貸出できます。") If LoanBooksQty > 3;	
BookId	Id	本管理番号		No
BookTitle	VarChar(40)	本タイトル		
LoanBookComment	VarChar(40)	貸出本コメント		No

1 度の貸出には、複数の本を含める必要があるため、Loan トランザクションに Book という第 2 レベルを追加します。

Book トランザクションで登録された本を紐づけ、同じ本を複数指定できないように、BookId を第 2 レベルの主キーにします。

これにより、外部キーであり、主キーとなるため、同じ本を複数登録できません。

他に、BookTitle も推論される項目として追加し、貸し出す本についてのコメントを記録するため、LoanBookComment を VarChar タイプの 40 桁で定義します。

1 度の貸出で本は 3 冊までしか入力できないように制御が必要です。

そのため、入力された本をカウントするため、第 1 レベルに LoanBooksQty という Numeric タイプの 4 桁の項目を追加し、グローバル式に count(LoanBookComment) と指定し、Book レベルに入力された本をカウントできるようにします。

そして、このカウント数が 3 を超えた場合にエラーメッセージが表示されるように、次のルールを定義します。

Error("本は3冊まで貸出できます。") if LoanBooksQty > 3;

トランザクションの定義: Loan (Magazine レベル)



Loan				
名前	タイプ	DESCRIPTION	式	Null 許容
Loan	Loan	貸出		
LoanId	Id	貸出番号		No
LoanDate	Date	貸出日		No
MemberDocument	Numeric(8.0)	会員番号		No
MemberName	Name	会員名		No
MemberAddress	Address, GeneXus	会員住所		No
LoanReturnDate	Numeric(4.0)	返却日	LoanDate.adddays(15)	No
NotebookId	Id	ノートパソコン番号		Yes
NotebookDescription	LongVarChar(2M)	ノートパソコン簡易説明		No
LoanBooksQty	Numeric(4.0)	貸出本数	count(LoanBookComment)	No
LoanMagazinesQty	Numeric(4.0)	貸出雑誌数	count(LoanMagazineComment)	No
Book	Book	貸出本	Error("雑誌は4冊まで貸出できます。") If LoanMagazinesQty > 4;	No
BookId	Id	本管理番号		No
BookTitle	VarChar(40)	本タイトル		No
LoanBookComment	VarChar(40)	貸出本コメント		No
Magazine	Magazine	貸出雑誌		No
MagazineId	Id	雑誌番号		No
MagazineTitle	VarChar(40)	雑誌タイトル		No
LoanMagazineComment	VarChar(40)	貸出雑誌コメント		No

1 度の貸出には、本とは別に、複数の雑誌を含める必要があるため、Loan トランザクションに第 2 レベル Book と並列になる Magazine という第 2 レベルを追加します。

Magazine トランザクションで登録された雑誌を紐づけ、同じ雑誌を複数指定できないように、MagazineId を第 2 レベル Magazine の主キーにします。

これにより、外部キーであり、主キーとなるため、同じ雑誌を複数登録できません。

他に、MagazineTitle も推論される項目として追加し、貸し出す雑誌についてのコメントを記録するため、LoanMagazineComment を VarChar タイプの 40 桁で定義します。

1 度の貸出で雑誌は 4 冊までしか入力できないように制御が必要です。

そのため、入力された雑誌をカウントするため、第 1 レベルに LoanMagazinesQty という Numeric タイプの 4 桁の項目を追加し、グローバル式に count(LoanMagazineComment) と指定し、Magazine レベルに入力された雑誌をカウントできるようにします。

そして、このカウント数が 4 を超えた場合にエラーメッセージが表示されるように、次のルールを定義します。

Error("雑誌は4冊まで貸出できます。") if LoanMagazinesQty > 3;

トランザクションの定義: Loan(不足したルール)



Loan				
名前	タイプ	デスクリプション	式	Null 許容
Loan	Loan	貸出		
LoanId	Id	貸出番号		No
LoanDate	Date	貸出日		No
MemberDocument	Numeric(8.0)	会員番号	Default(LoanDate,&Today);	No
MemberName	Name	会員名	NoAccept(LoanDate);	No
MemberAddress	Address, GeneXus	会員住所		
LoanReturnDate	Numeric(4.0)	返却日	LoanDate.adddays(15)	
NotebookId	Id	ノートパソコン番号		Yes
NotebookDescription	LongVarChar(2M)	ノートパソコン簡易説明		
LoanBooksQty	Numeric(4.0)	貸出本数	count(LoanBookComment)	
LoanMagazinesQty	Numeric(4.0)	貸出雑誌数	count(LoanMagazineComment)	
Book	Book	貸出本		
BookId	Id	本管理番号		No
BookTitle	VarChar(40)	本タイトル		
LoanBookComment	VarChar(40)	貸出本コメント		No
Magazine	Magazine	貸出雑誌		
MagazineId	Id	雑誌番号		No
MagazineTitle	VarChar(40)	雑誌タイトル		
LoanMagazineComment	VarChar(40)	貸出雑誌コメント		No

図書館の担当者から受けた要件の中に、貸出日の値は、常に現在の日付となり、変更できないというものがありました。
 こちらの対応を行っていなかったため、Rules エlementで以下のルールを宣言します。

Default(LoanDate, &Today);
NoAccept(LoanDate);

トランザクションの定義: Loan (コントロールタイプの変更)



Loan X

Structure Web Layout Rules Events Variables Help Documentation Patterns

名前	タイプ	デスクリプション	式	Null 許容
Loan	Loan	貸出		
LoanId	Id	貸出番号		No
LoanDate	Date	貸出日		No
MemberDocument	Numeric(8.0)	会員番号		No
MemberName	Name	会員名		
MemberAddress	Address, GeneXus	会員住所		
LoanReturnDate	Numeric(4.0)	返却日	LoanDate.adddays(15)	
NotebookId	Id	ノートパソコン番号		Yes
LoanBooksQty	Numeric(4.0)	貸出本数	count(
LoanMagazinesQty	Numeric(4.0)	貸出雑誌数	count(
Book	Book	貸出本		
BookId	Id	本管理番号		
BookTitle	VarChar(40)	本タイトル		
LoanBookComment	VarChar(40)	貸出本コメント		
Magazine	Magazine	貸出雑誌		
MagazineId	Id	雑誌番号		
MagazineTitle	VarChar(40)	雑誌タイトル		
LoanMagazineComment	VarChar(40)	貸出雑誌コメント		

Control Info	
Control Type	Dynamic Combo Box
Data Source From	Attributes
Item Values	NotebookId
Item Descriptions	NotebookDescription
Sort Descriptions	True
Conditions	NotebookStatus = Status.Available;
Instantiated Attrib	
Empty Item	True

この章では、データのモデリングにフォーカスしていますが、1 点入力形式についてのオプションを紹介します。

GeneXus によって生成される画面において、入力形式は、[Control Type] プロパティで設定することができます。

ノートパソコンの管理番号をユーザーに表示させないため、NotebookId の [Control Type] プロパティを Dynamic Combo Box に設定します。

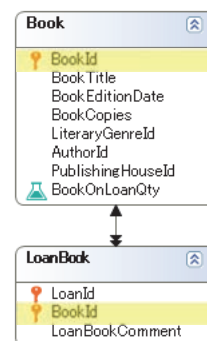
[Item Descriptions] プロパティを NotebookDescription に設定することで、画面では、登録済みのデータのみが表示されるコンボボックスとして表示されます。

さらに [Conditions] プロパティに NotebookStatus = Status.Available; と入力することで、コンボボックスに表示される選択肢は、利用可能なノートパソコンのみとなります。

最後に、NotebookId は、[Null 許容] を Yes に設定していたため、空の値を選択できるように [Empty Item] プロパティを True に設定します。

本の利用可能な冊数（貸出中の冊数）

Book				
Structure Web Layout Rules Events Variables Help Documentation Patterns				
名前	タイプ	デスクリプション	式	Null 許容
Book	Book	本		
BookId	Id	本管理番号		No
BookTitle	VarChar(40)	本タイトル		No
BookEditionDate	Date	本発行日		No
BookCopies	Numeric(4.0)	本冊数		No
LiteraryGenreId	Id	文学ジャンル番号		No
LiteraryGenreName	Name	文学ジャンル名		No
AuthorId	Id	著者番号		No
AuthorName	Name	著者名		No
PublishingHouseId	Id	出版社番号		No
PublishingHouseName	Name	出版社名		No
BookOnLoanQty	Numeric(4.0)	本貸出冊数	count(LoanBookComment)	



貸出情報を記録できるようにトランザクション設計を行いましたが、現実に基づくためには、不足した実装があります。

貸出を行う本が利用可能かどうか把握する必要があります。

図書館が保有している冊数は、登録したデータから確認できますが、そのうち、何冊貸出中で、何冊利用可能であるかわかりません。

貸出中の冊数と、利用可能な冊数をトランザクション設計に基づき、取得できるようにします。

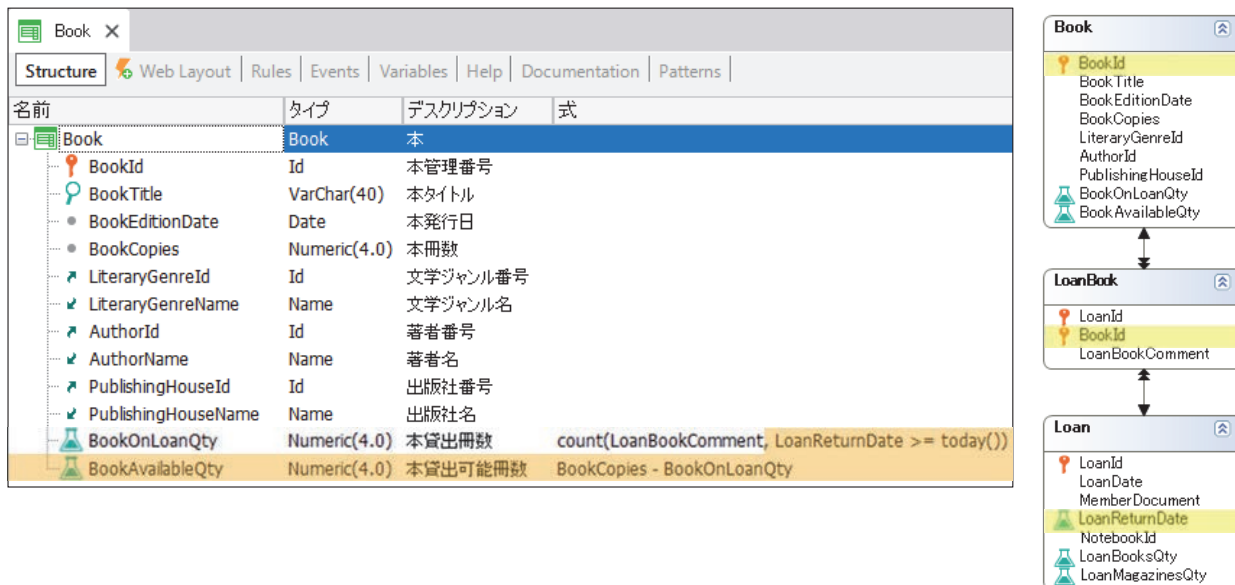
貸出中の冊数を取得する項目属性を追加します。

Book トランザクションで、BookOnLoanQty という Numeric タイプの 4 桁の項目属性を追加し、グローバル式として count(LoanBookComment) と記述します。このグローバル式によって、BookOnLoanQty には、その本の貸出中の冊数の合計が取得されます。

式が定義されているテーブルは、BOOK テーブルであり、式が集計を行うために参照するテーブルは、LOANBOOK テーブルです。

この場合、BOOK テーブルと、LOANBOOK テーブルには、BookId によって 1 対 N の関係があるため、特定の本に対する貸出中の冊数を求めることができます。

本の利用可能な冊数（利用可能冊数）



貸出中の冊数を求めることができるようになったため、利用可能な冊数を求める実装に進みたいと思いますが、貸出中の冊数に不備がありました。先ほどの定義のままでは、過去の貸出も含め、すべての本の貸出がカウントされてしまいます。

Count 関数に、第 2 引数を記述し、集計対象の本は、今日よりも未来日が返却日となっているデータのみとなるように、以下の式へ変更します。

`count(LoanBookComment, LoanReturnDate >= today())`

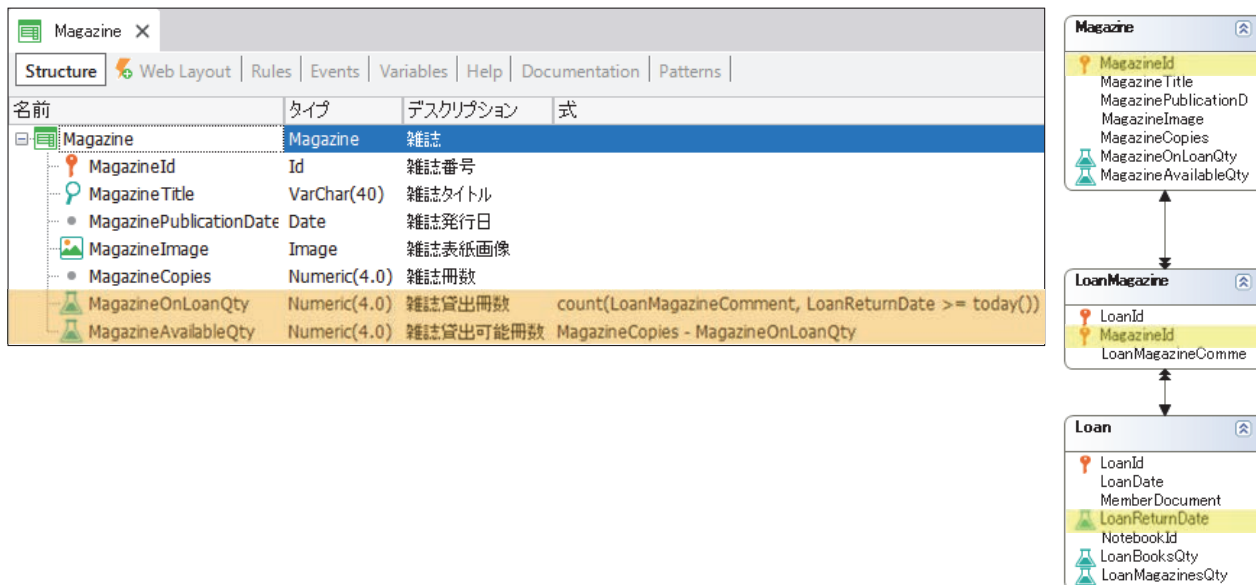
LoanReturnDate は、LOAN テーブルの項目ですが、LOANBOOK テーブルの拡張テーブルとなるため、参照可能です。

図書館が保有する冊数と、貸出中の冊数がわかるようになったため、利用可能な冊数を求める項目を追加します。

BookAvailableQty という Numeric タイプの 4 桁の項目属性を追加し、式として以下を記述します。

`BookCopies - BookOnLoanQty`

雑誌の利用可能な冊数



本と同様に、雑誌についても利用可能な冊数を取得する必要があります。そのため、現在、貸出中の冊数がわかるように Magazine トランザクションへ項目属性を追加し、グローバル式を定義しました。内容は、集計対象となる項目属性が異なるだけで、Book トランザクションで定義したものと同様です。

そして、図書館が保有する雑誌の冊数と貸出中の冊数を利用し、現在貸出可能な冊数を取得できるようにしています。

トランザクションの定義: Loan (貸出制御)

Loan				
名前	タイプ	デスクリプション	式	Null 許容
Loan	Loan	貸出		
LoanId	Id	貸出番号		No
LoanDate	Date	貸出日		No
MemberDocument	Numeric(8.0)	会員番号		No
MemberName	Name	会員名		No
MemberAddress	Address, GeneXus	会員住所		No
LoanReturnDate	Numeric(4.0)	返却日	LoanDate.adddays(15)	No
NotebookId	Id	ノートパソコン番号		Yes
LoanBooksQty	Numeric(4.0)	貸出本数	count(LoanBookComment)	No
LoanMagazinesQty	Numeric(4.0)	貸出雑誌数	count(LoanMagazineComment)	No
Book	Book	貸出本		
BookId	Id	本管理番号		No
BookTitle	VarChar(40)	本タイトル		No
LoanBookComment	VarChar(40)	貸出本コメント		No
BookAvailableQty	Numeric(4.0)	本貸出可能冊数	BookCopies - BookOnLoanQty	No
Magazine	Magazine	貸出雑誌		
MagazineId	Id	雑誌番号		No
MagazineTitle	VarChar(40)	雑誌タイトル		No
LoanMagazineComment	VarChar(40)	貸出雑誌コメント		No
MagazineAvailableQty	Numeric(4.0)	雑誌貸出可能冊数	MagazineCopies - MagazineOnLoanQty	No

```
Error("貸出可能な冊数が本にありません。") If BookAvailableQty < 0;
Error("貸出可能な冊数が雑誌にありません。") If MagazineAvailableQty < 0;
```

本や雑誌についての貸出可否を判断するために必要な情報が取得できるようになりました。
貸出を管理する Loan トランザクションへ貸出可能冊数を求めるグローバル式項目属性を本、雑誌それぞれを追加します。

そして、このグローバル式の値を条件に以下のルールを追加します。

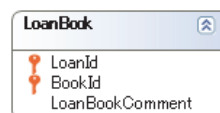
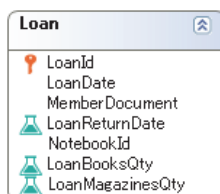
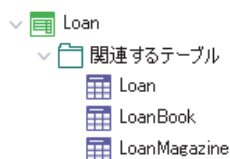
Error("貸出可能な冊数が本にありません。") If BookAvailableQty < 0;

Error("貸出可能な冊数が雑誌にありません。") If MagazineAvailableQty < 0;

このルールにより、貸出可能な冊数がない場合、エラーが発生し、登録できなくなります。

Loan トランザクション: フォームとテーブル構造

貸出番号	LoanId										
貸出日	LoanDate										
会員番号	MemberDocument										
会員名	MemberName										
会員住所	MemberAddress										
返却日	LoanReturnDate										
ノートパソコン番号説明	NotebookId										
貸出本数	LoanBooksQty										
貸出雑誌数	LoanMagazinesQty										
貸出本 <table border="1"> <tr> <th>GRID</th> </tr> <tr> <td> <table border="1"> <tr> <th>本管理番号</th> <th>本タイトル</th> <th>貸出本コメント</th> <th>本貸出可能冊数</th> </tr> <tr> <td>BookId</td> <td>BookTitle</td> <td>LoanBookComment</td> <td>BookAvailableQty</td> </tr> </table> </td> </tr> </table>		GRID	<table border="1"> <tr> <th>本管理番号</th> <th>本タイトル</th> <th>貸出本コメント</th> <th>本貸出可能冊数</th> </tr> <tr> <td>BookId</td> <td>BookTitle</td> <td>LoanBookComment</td> <td>BookAvailableQty</td> </tr> </table>	本管理番号	本タイトル	貸出本コメント	本貸出可能冊数	BookId	BookTitle	LoanBookComment	BookAvailableQty
GRID											
<table border="1"> <tr> <th>本管理番号</th> <th>本タイトル</th> <th>貸出本コメント</th> <th>本貸出可能冊数</th> </tr> <tr> <td>BookId</td> <td>BookTitle</td> <td>LoanBookComment</td> <td>BookAvailableQty</td> </tr> </table>	本管理番号	本タイトル	貸出本コメント	本貸出可能冊数	BookId	BookTitle	LoanBookComment	BookAvailableQty			
本管理番号	本タイトル	貸出本コメント	本貸出可能冊数								
BookId	BookTitle	LoanBookComment	BookAvailableQty								
貸出雑誌 <table border="1"> <tr> <th>GRID</th> </tr> <tr> <td> <table border="1"> <tr> <th>雑誌番号</th> <th>雑誌タイトル</th> <th>貸出雑誌コメント</th> <th>雑誌貸出可能冊数</th> </tr> <tr> <td>MagazineId</td> <td>MagazineTitle</td> <td>LoanMagazineComment</td> <td>MagazineAvailableQty</td> </tr> </table> </td> </tr> </table>		GRID	<table border="1"> <tr> <th>雑誌番号</th> <th>雑誌タイトル</th> <th>貸出雑誌コメント</th> <th>雑誌貸出可能冊数</th> </tr> <tr> <td>MagazineId</td> <td>MagazineTitle</td> <td>LoanMagazineComment</td> <td>MagazineAvailableQty</td> </tr> </table>	雑誌番号	雑誌タイトル	貸出雑誌コメント	雑誌貸出可能冊数	MagazineId	MagazineTitle	LoanMagazineComment	MagazineAvailableQty
GRID											
<table border="1"> <tr> <th>雑誌番号</th> <th>雑誌タイトル</th> <th>貸出雑誌コメント</th> <th>雑誌貸出可能冊数</th> </tr> <tr> <td>MagazineId</td> <td>MagazineTitle</td> <td>LoanMagazineComment</td> <td>MagazineAvailableQty</td> </tr> </table>	雑誌番号	雑誌タイトル	貸出雑誌コメント	雑誌貸出可能冊数	MagazineId	MagazineTitle	LoanMagazineComment	MagazineAvailableQty			
雑誌番号	雑誌タイトル	貸出雑誌コメント	雑誌貸出可能冊数								
MagazineId	MagazineTitle	LoanMagazineComment	MagazineAvailableQty								



定義した Loan トランザクションでは、図書館の担当者より提示あった要件をすべて満たすことができるようになりました。
最後に、今回の実装で生成された Loan トランザクションの Web Layout エlementと、生成されたテーブル構造について確認します。

Loan トランザクションには、並列に定義した 2 つの第 2 レベルがありました。
そのため、Web Layout には、2 つのグリッドコントロールがそれぞれ Book レベル、Magazine レベルに対して作成されています。

Loan トランザクションに基づき生成されるテーブルも確認します。
この定義の場合、次の 3 つのテーブルが作成されます。

- LOAN

第 1 レベルに関連付けられ、LoanId が主キーのテーブル

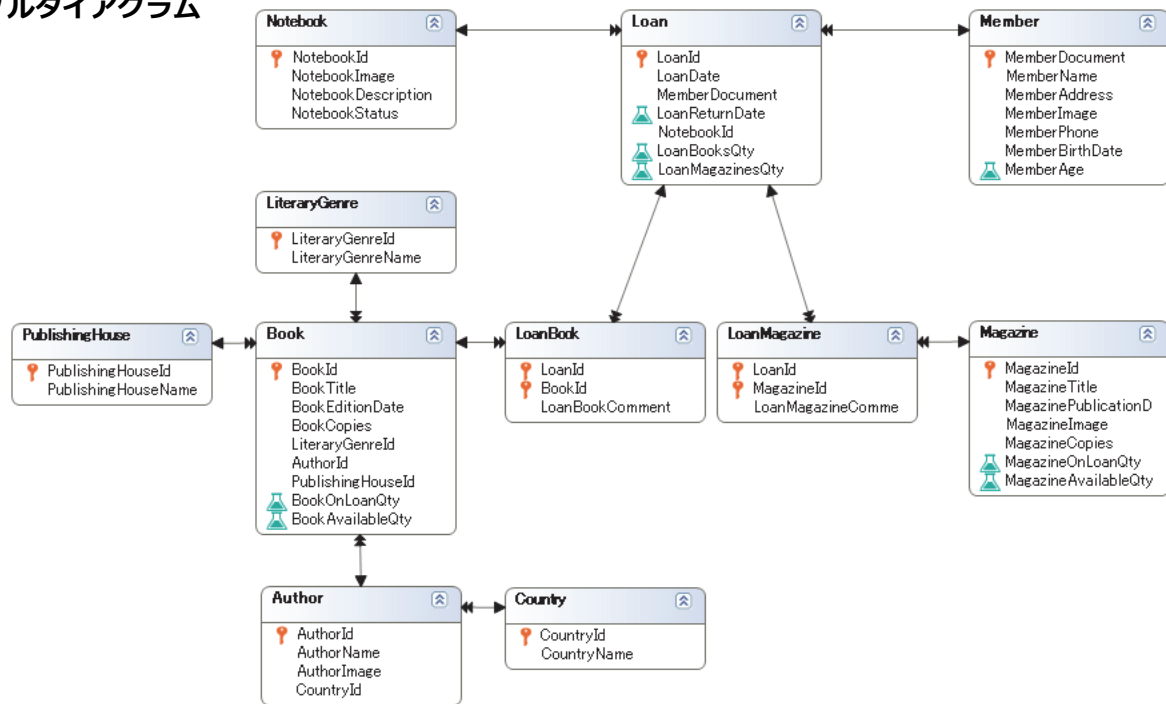
- LOANBOOK

第 2 レベル Book に関連付けられ、LoanId と BookId が複合主キーのテーブル

- LOANMAGAZINE

第 2 レベル Magazine に関連付けられ、LoanId と MagazineId が複合主キーのテーブル

テーブルダイアグラム



図書館からの要望に基づき、データモデルの設計を行いました。
現時点で、GeneXus が作成するテーブルの関係性を Diagram オブジェクトで確認できます。

まとめ

常にエンドユーザーと協力しながら、モデル化する現実を分析、評価し、適切だと判断したものを実装する



常に、モデル化する現実を分析、評価し、適切だと判断したものを実装する必要があります。

そのためには、設計の選択を導いてくれるエンドユーザーと常に協力することが不可欠です。

アプリケーションは、エンドユーザーがビジネスをより適切に管理し、発展させるためのサポートツールでなければならないということを忘れないでください。

*GeneXus*TM

training.genexus.com
wiki.genexus.com