**A Longer Example of UML Database Modeling for Relational Database**

**[1] Initial refined specifications**

**Small Library**

An extremely small library wants to keep track of its collection. Information about all books should be stored with unique US library of congress call numbers. The author names, publishers, dates of publications and categories should be stored. A book must be under at least one category, but it may be under many categories. A patron may borrow books. There may be more than one copies of the same book. For example, there may be five copies of the book "Gone with the wind" and thus five patrons can check the book out at the same time. Names, SS# (unique), addresses, phone numbers and email addresses of patrons should be stored.

A patron may check out books, which have a check-out period of two weeks. There is a penalty of 50 cents per day per book. The penalty account is updated at the end of every month and when books are returned. A patron may specify favorite categories, such as 'horror', 'computer', 'romance', etc. When new books in this category arrive, the patron may be informed through email.

It should be possible for the library to find out the circulation of a given book.

A librarian may also change the penalty balance of a patron. However, when this is done, the staff ID of the librarian should be noted.

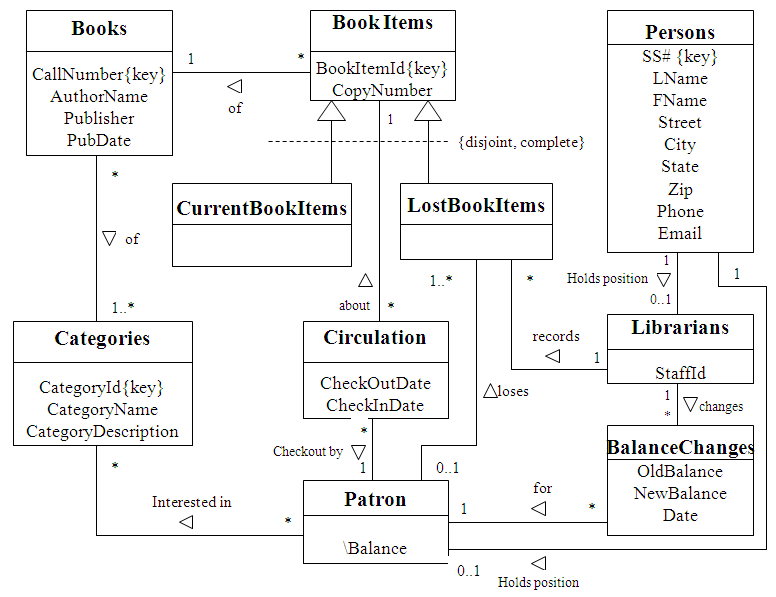
If a book is lost, it should be recorded. A lost may be reported by a patron or a book is simply lost with no one responsible for it. If a book is lost by a patron, a replacement charge will be charged to the patron. The librarian who records the lost should be stored.

**Task:** Construct a class diagram using UML for the design of the following application. It is not necessary to specify operations of the classes. It is also not necessary to specify the data types and visibility of the attributes. Make reasonable assumptions.

**An example result**

**Some example possible assumptions:**

* Email of librarian is desirable.
* A copy number is used to identify the copy of a book.
* No book renewal



**[2] Conversion to relational schema (examples)**

Books(CallNumber, AuthorName, Publisher, PubDate)

Categories(CategoryID, CategoryName, CategoryDescription)

BookCategories(CallNumber, CategoryID)

* Call Number is a foreign key of Books(BookCallNumber).
* CategoryID is a foreign key of Categories(CategoryID).

BookItems(BookItemId, CopyNumber, BookCallNumber)

* BookCallNumber is a foreign key of Books(BookCallNumber).

CurrentBookItems(CurrentBookItemId)

* CurrentBookItemId is a foreign key of BookItems(BookItemId).

LostBookItem(LostBookItemId, PatronSSNum, RecorderStaffId)

* LostBookItemId is a foreign key of BookItems(BookItemId).
* PatronSSNum is a foreign key of Patron(PatronSSNum).
* RecorderStaffId is a foreign key of Librarians(StaffId).

Circulation(CirculationId, CheckOutDate, CheckInDate, BookItemId, PatronSSNum)

* BookItemId is a foreign key of BookItems(BookItemId).
* PatronSSNum is a foreign key of Patron(PatronSSNum).

Perons(SSNum, LName, FName, Street, City, State, Zip, Phone, EMail)

Patrons(PatronSSNum, Balance)

* PatronSSNum is a foreign key of Persons(SSNum).

Librarians(StaffId, LibrarianSSNum)

* LibrarianSSNum is a foreign key of Persons(SSNum)

BalanceChanges(BalanceChangesId, OldBalance, NewBalance, Date, PatronSSNum, ChangeStaffId)

* PatronSSNum is a foreign key of Patron(PatronSSNum).
* ChangeStaffId is a foreign key of Librarians(StaffId).

Interests(PatronSSNum, CategoryId)

* PatronSSNum is a foreign key of Patron(PatronSSNum).
* CategoryId is a foreign key of Categories(CategoryId).