

Design of a Library Management System in PL/SQL

Adeline Southard & Brian Sandon

Application Description

Title: Library Management Software

Purpose: Facilitate the locating, checkout, and return of library resources (books, periodicals, film, etc) by library patrons. As well as facilitate library clerks in providing customer support and librarians in managing/curating the library's collection.

User Groups: library patron, library clerk, librarian, administrator, teacher/instructor

Usage:

(1) Library patron

- Search for and check out books and other materials
- Request holds for checked out resources
- Request an extension for a resource they aren't done with
- Return books and other materials
- May have to pay fines on overdue materials

(2) Teacher/Instructor

- Checks out books and resources for instructional use
- Arranges events with librarians

(3) Library clerk

- Issue library cards to new patrons
- Help library patrons find and checkout books and other materials
- Check returned books and materials back into the catalogue
- Contact patrons about books and materials they are waiting for
- Call patrons about overdue books and late fees
- Update library logs

(4) Librarian

- Handles library administration, including
 - i. Adding/removing books and resources from the library's collection
- Schedules library events

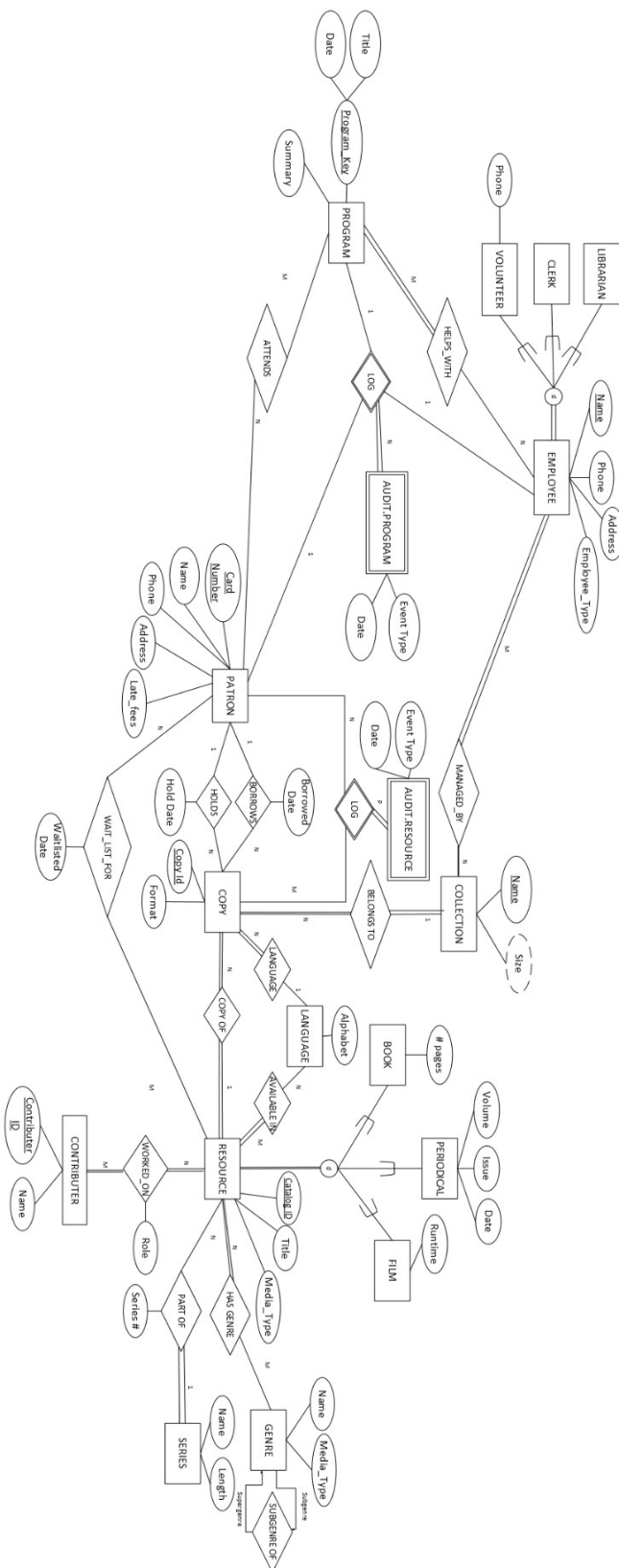
(5) Administrator

- Handles administration of the library management tool's database

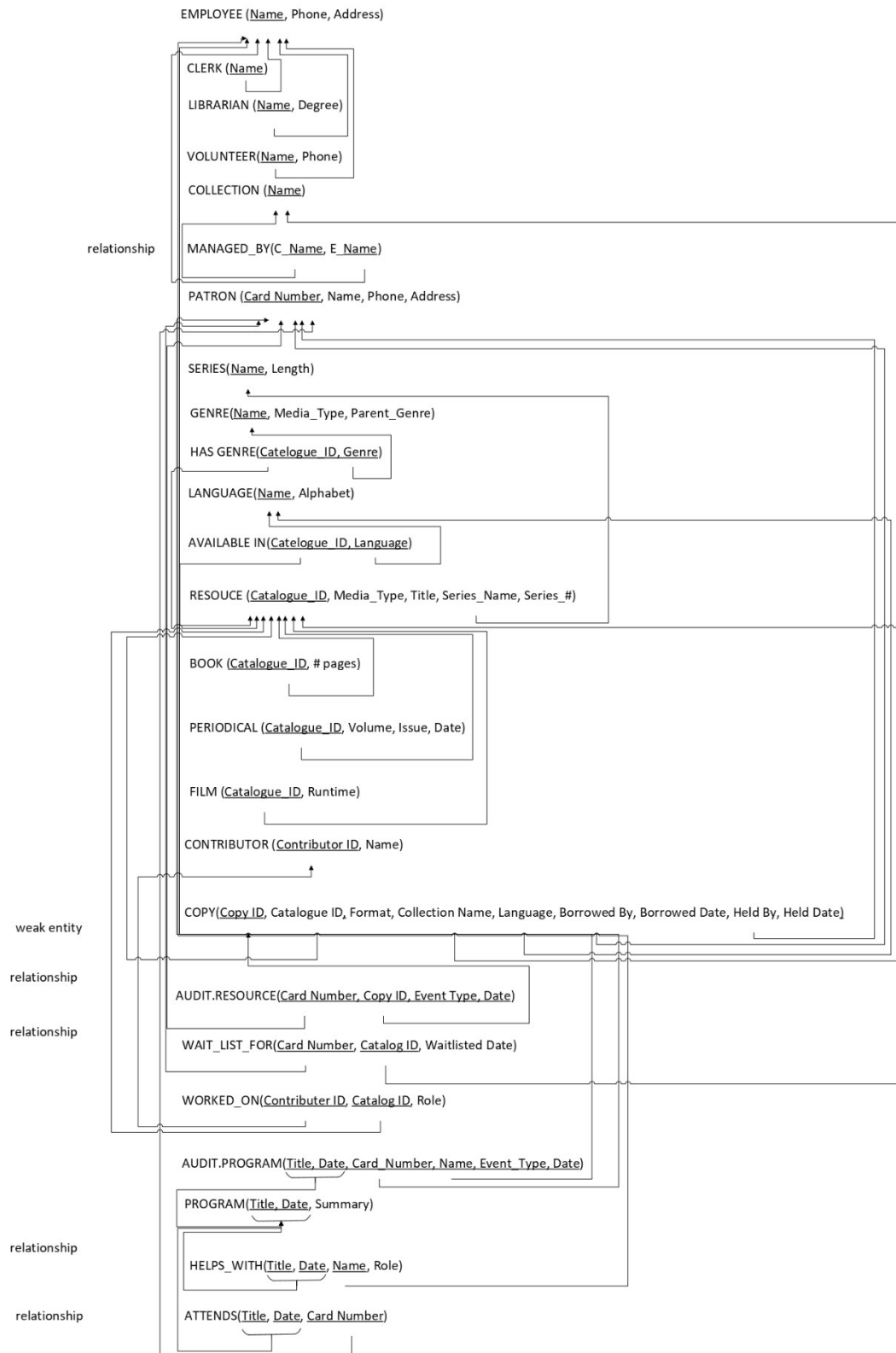
Business Rules:

- A patron may only take out 15 resources at a time.
- There may be multiple copies of a resource in the library.
- A copy of a resource must belong to a single collection.
- A patron may checkout out a copy of a resource for a 2 week period.
- A patron may renew a checked-out resource any time before the 2 week period expires.
- A patron can request to be put on a waiting list for a resource.
- When a copy of a resource is returned, the patron first on the waiting list for that resource is notified and that resource may be held until the patron picks it up, cancels the hold, or 2 weeks pass.
- A patron must register for a library card to use the library.
- For a library card, the library must know your name, place of residence, and phone number.
- Librarians and library clerks must update inventory records when books are taken out and returned.
- Library staff members have differing permissions regarding the resources inventory.
- Library staff must keep logs for patron activities (what resources are checked out and returned, and when).
- A library may have separate collections. E.g. a children's section and an adult library.
- A library clerk or librarian may be responsible for one or more collections.

Conceptual Schema



Relational Schema



Additional Constraints

PATRON (Card Number, Name, Phone, Address)

- Card Number must have the following format: #####-#####
- Phone must be a valid phone number
- For a patron to be added to the system they must have a valid residence/home address within the town/area the library services. They must also have a phone number so that the library may contact them.

BOOK (Catalogue_ID, # pages)

- # pages must be more than 0

FILM (Catalogue_ID, Runtime)

- Runtime must be more than 0

RESOUCE (Catalogue_ID, Title, Series_Name, Series_#)

- Only librarians (not clerks) are allowed to add resources to or remove resources from the library's collections

COPY(Copy ID, Catalogue ID, Format, Collection Name, Language, Borrowed By, Borrowed Date, Held By, Held Date)

- Only librarians (not clerks) are allowed to add copies of a resource to or remove copies of a resource from the library's collections
- Language must be a language that the related resource has a LANGUAGES AVAILABLE relationship with.
- Only Borrowed By and Borrowed Date or Held By and Held Date can be non-null for a given entry at a time.
- Borrowed Date is the date the copy of a resource was first borrowed or the date it was last renewed
 - Previous borrowing events and renewals are still tracked in the system.
- Only 15 copy rows may be "Borrowed By" a single patron at a time.
- Valid Formats depend on the class of the associated resource
 - Book formats can be "Hardcover", "Softcover", or "eBook"
 - Film formats can be "VCR" or "DVD"

CONTRIBUTOR (Contributor ID, Name)

- Valid roles depend on the class of the associated resource
 - Book and periodical roles can be "Author", "Editor", "Publisher", "Contributor", "Illustrator"

- Film roles can be “Director”, “Producer”, “Actor/Actress”, “Voice Actor/Actress”

LOG_ENTRY(Card Number, Copy ID, Event Type, Date)

- Event Type can be “Checked out”, “Returned”, or “Renewed”
 - A patron can’t check out a copy of a resource again while they have that resource checked out.
 - A patron can only renew a resource after they checked it out and before they return it.
 - A patron cannot renew a resource 2 weeks past the most recent checkout or renewal of that resource.

Implementation Specification

Triggers

Trigger ResourceReturned

- Trigger is called before an update of the Copy table
- It checks if the checkout and hold fields are going from some value to null.
- If check succeeds trigger sends a notification that a copy of the resource is available.

Trigger ResourceActivity

- Trigger is called before an update of the Copy table.
- The trigger compares the checkout and hold fields for a record.
 - If Held_By and Held_Date fields have been updated from null to some value, function inserts a “On hold” event into the AUDIT.RESOURCE table.
 - If Held_By and Held_Date fields have been updated from some value to null, function inserts a “Hold released” event into the AUDIT.RESOURCE table.
 - If Borrowed_By and Borrowed_Date fields have been updated from null to some value, function inserts a “Checked out” event into the AUDIT.RESOURCE table.
 - If Borrowed_By and Borrowed_Date fields have been updated from some value to null, function inserts a “Returned” event into the AUDIT.RESOURCE table.

- If Borrowed_Date was updated from one date to a later date, function inserts a “Renewed” event into the AUDIT.RESOURCE table.

Trigger ProgramActivity

- Trigger is called after an insert or update on the PROGRAM table
- If the Program was inserted, function inserts a “Program Scheduled” event into the AUDIT.PROGRAM table.
- If the Title field was updated, function inserts a “Program Renamed” event into the AUDIT.PROGRAM table.
- If the Date field was updated, function inserts a “Program Rescheduled” event into the AUDIT.PROGRAM table.

Trigger ProgramContributorActivity

- Trigger is called after an insert, update, or delete on the HELPS_WITH table
- On insert, function inserts a “Contributor Added” event into the AUDIT.PROGRAM table
- On delete, function inserts a “Contributor Removed” event into the AUDIT.PROGRAM table
- If the Name field is updated, a “Contributor Removed” event is created for the old EMPLOYEE record, while a “Contributor Added” event is created for the new EMPLOYEE record.

Trigger ProgramParticipantActivity

- Trigger is called after an insert, update, or delete on the ATTENDS table
- On insert, function inserts a “Participant Signed Up” event into the AUDIT.PROGRAM table
- On delete, function inserts a “Participant Cancelled” event into the AUDIT.PROGRAM table
- If the Card_Number field is updated, a “Participant Cancelled” event is created for the old PATRON record, while a “Participant Signed Up” event is created for the new PATRON record.

Functions

Function AddNewPatron(name, phone , address) returns card_number

- This function inserts a new record into table PATRON
- Function checks if the patron already exists by seeing if there is an identical record (save for card_number) in the database

- Function also checks that name, phone, and address are not null. This is already enforced by not null constraints on the table, but the additional error checking helps ensure function displays proper error messages.
- The function also generates a new card number for the patron
- In any error above occurs, function prints an error message and returns card_number = null, otherwise the function returns the new card_number

Function AddResource(media_type, title, series_name, series_number, languages, genres, contributors, resource_type_fields) returns resource_id

- This function inserts a new record into table RESOURCE.
- This function will also insert the corresponding records in the WORKS_ON, AVAILABLE_IN, and HAS_GENRE table
- This function checks that a title and a resource_type is provided, and that the resource type corresponds to a valid resource type.
- This function also checks that languages, genres, and contributors each contain at least one value.
- Function checks that for all genres passed in, genre.media_type matches the resource.media_type or is "General"
- Function generates a new resource_id value
- Function switches on media_type for specific processing:
 - For BOOK:
 - Function checks that resource_type_fields includes a #_pages field and #_pages is a valid value
 - Function also checks that for each contributor its role is one of the following: "Author", "Editor", "Publisher", "Contributor", "Illustrator"
 - Function inserts a new record into table BOOK
 - For PERIODICAL:
 - Function checks that resource_type_fields includes volume, issue, and date fields. It also verifies that these fields have valid values
 - Function also checks that for each contributor its role is one of the following: "Author", "Editor", "Publisher", "Contributor", "Illustrator"
 - Function inserts a new record into table PERIODICAL
 - For FILM
 - Function checks that resource_type_fields includes a runtime field and runtime is a valid value
 - Function also checks that for each contributor its role is one of the following: "Director", "Producer", "Actor/Actress", "Voice Actor/Actress"
 - Function inserts a new record into table FILM

- If any error occurs during the process, an error message is printed, all changes are rolled back, and the function returns resource_id = null, otherwise the function returns the generated resource_id.

Function AddCopy(catalogue_id, format, collection_name, language) returns copy_id

- This function adds a new record to the COPY table
- The function checks that language corresponds to a language the associated RESOURCE is AVAILABLE_IN.
- Function switches on the corresponding RESOURCE's media_type
 - For BOOK
 - Function checks that format is one of the following: "softcover", "hardcover", "e-book"
 - For FILM
 - Function checks that format is one of the following: "VCR", "DVD"
- Function generates a new copy_id
- If any error occurs during the process, an error message is printed, all changes are rolled back, and the function returns copy_id = null, otherwise the function returns the generated copy_id

Function AddToWaitlist(Card_number,Catalogue_id) returns error_code

- This function inserts a record to the table WAIT_LIST_FOR
- The function assigns the current date to Waitlisted_date
- The function returns error_code 0

Function HoldCopy(Card_number,Copy_id) returns error_code

- This function updates a COPY record to hold that copy for a patron.
- The function checks that the COPY is not currently held or checked out by another patron
- This function also inserts a "Hold" event to the LOG_ENTRY table
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function CheckoutCopy(Card_number,Copy_id) returns error_code

- This function updates a Copy record to checkout that copy to a patron.
- The function checks that the Copy is not currently held or checked out by another patron.
- If the patron checking out the copy has it on hold the hold information is nulled out and the checkout information is updated

- The function also checks that the patron trying to hold the resource doesn't currently have 15 resources checked out
- This function also inserts a "Checked out" event to the LOG_ENTRY table
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function Renew(Card_number,Copy_id) returns error_code

- This function updates a COPY record to change the Borrowed_date to the current date
- The function checks if the specified PATRON has that copy checked out.
- The function also checks that the previous Borrowed_date was less than 2 weeks ago (PATRONS can't renew after the resource they borrowed is overdue).
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function ReleaseHold(Card_number,Copy_id) returns error_code

- This function updates a COPY record to release a hold on that copy.
- The function checks if the specified PATRON has a hold on that copy.
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function ReturnCopy(Card_number,Copy_id) returns error_code

- This function updates a COPY record to mark it as available
- The function checks if the specified PATRON has that copy checked out.
- The function checks if the current date is more than 2 weeks after the Borrowed_date, if so the function will also update the PATRON's Late_fees field
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function PayFee(Card_number,Payment) returns error_code

- This function updates a PATRON record to apply late fee payments.
- The function checks if the Payment is less than or equal to the PATRON's Late_fees
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function AddProgram (title, date) returns program_id

- This function inserts a new program into the PROGRAM table
- This function checks that both a title and date are provided

- This function checks that the (title, date) key of the program is unique (there cannot be two of the same program occurring on the same day)
- If any error occurs during the process, an error message is printed, all changes are rolled back, and the function returns program_id = null, otherwise the function returns the generated program_id

Function UpdateProgramAttendee (card_number, program_title) returns error_code

- This function inserts a new patron into the ATTENDS table
- This function checks that a patron is not already attending the program
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function UpdateProgramHelper (name, role, program_title) returns error_code

- This function inserts a new employee into the HELPS_WITH table
- This function checks that an employee is not already helping with a program
- If any error occurs during the process, an error message is printed and the function returns error_code -1, otherwise the function returns error_code 0

Function CollectionSize (name) returns collection_size

- This function gets the count of all items in a COLLECTION

Procedures

Procedure ShowResource(catalogue_id)

- This procedure shows the information on the requested resource.
- Shows any available copies of the resource

Procedure ShowCollection(name)

- This procedure shows the information on the requested collection
- This procedure also shows the copies of resources in the collection, and the employees who manage the collection

Procedure ShowEmployee (name)

- This procedure shows the information on the requested employee
- Shows any programs an employee is helping with

Procedure ShowPatron (card_number)

- This procedure shows the information on the requested patron
- Also shows any copies checked out by a patron and their date, any holds and their hold date, and any waitlists the patron is on.
- Shows any programs a patron is attending

Procedure ShowEvent (title, date)

- This procedure shows the information on the requested event
- Also shows the employees helping out with their event and their roles, and any patrons attending the event