

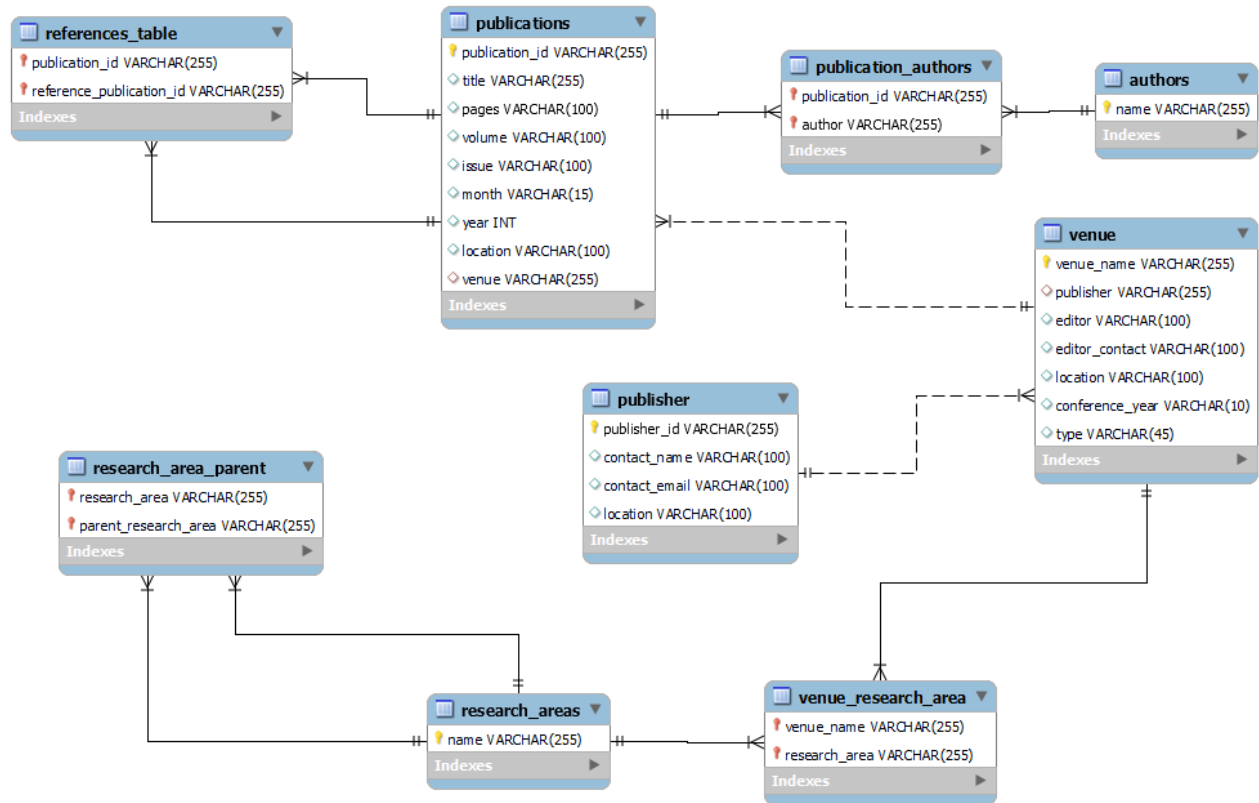
CSCI 3901
Dalhousie University
Master of Applied Computer Science

Final Project

Author: Dheeraj Dinesh Bhat
(Dalhousie ID: B00928874)

Database Design

Here is the database design for this project:



I have added the sql script to create these tables under the docs folder in the git repository.

Key Algorithm

To convert the given input file into required output with updated citations we will follow the following steps:

- Read the input file and extract the citations.
- Replace the citations with the appropriate IEEE formatted reference numbers.
- Extract the required data from the database for converting the citations into IEEE format.
- Generate the IEEE formatted reference list using the retrieved data at the end of the paper.
- Write the output file with the updated text and the appended references.

Code design

The code will contain the following classes:

2. **DatabaseComm**

This class will be used to connect, disconnect, and communicate with the available databases in mysql.

3. **PublicationLibrary**

We will be implementing the PublicationLibrary class as described in the problem.

It contains the following methods:

- boolean addPublication(String identifier, Map<String, String> publicationInformation)
- boolean addReferences(String identifier, Set<String> references)
- boolean addVenue(String venueName, Map<String, String> venueInformation, Set<String> researchAreas)
- boolean addPublisher(String identifier, Map<String, String> publisherInformation)
- boolean addArea(String researchArea, Set<String> parentArea)
- Map<String, String> getPublications(String key)
- int authorCitations(String author)
- Set<String> seminalPapers(String area, int paperCitation, int otherCitations)
- Set<String> collaborators(String author, int distance)
- Set<String> authorResearchAreas(String author, int threshold)

4. **PaperConversion**

We will Implement a main method in this class that will accept input and output file strings and we pass it to the convertPaper method of the PaperConverter class.

5. **IEEEReference**

This class is used to hold the reference index and the reference string for each cited publication. This will be used by PaperConverter class to extract and create IEEE citations and references.

6. **PaperConverter**

This class is used to read the input file and convert the citations into IEEE format. It contains the following methods:

- void convertPaper(String inputFile, String outputFile)
- Set<String> extractCitations(String inputFile)
- Map<String, IEEEReference> createIEEEReferences(Set<String> citations)
- void replaceCitations(String inputFile, String outputFile, Map<String, IEEEReference> ieeeReferences)

7. **Two classes are created for each of the table in the database. One class is POJO class to hold a row of data from the table. The other class would be a service class which is used to handle the database operations for each of the tables.**
8. **Finally there are a couple of helper classes which has static methods to help the other class perform their functionality.**

Test cases:

PublicationLibrary

1. addPublication

Input validation tests:

- Identifier is null
- Identifier is empty
- publicationInformation is null
- publicationInformation is empty
- publicationInformation has invalid keys
- publicationInformation has invalid values
- publicationInformation has extra keys that are not allowed

Boundary tests:

- Add to an empty publications list
- Add a publication with the minimum required information (e.g., authors, title, journal, pages, volume, issue, month, and year)
- Add a publication with a very large number of authors

Control flow tests:

- Add a publication to the library, then attempt to add the same publication again (duplicate identifier)
- Add multiple valid publications with different identifiers

2. addReferences

Input validation tests:

- identifier is null
- identifier is empty
- references is null
- references is empty
- A reference in the references set is null
- A reference in the references set is empty

Boundary tests:

- Add references to a publication that has not been referenced before
- Add a single reference to a publication
- Add multiple references to a publication
- Add references to a publication that has already been referenced multiple times
- Add references to a publication that references itself

Control flow test:

- Add references to a publication that does not exist in the library
- Add references to a publication after adding the same references before
- Add references to a publication, then retrieve the publication's information and verify that the references were added
- Add references to multiple publications in the library, then retrieve their information and verify that the references were added correctly
- Add references to a publication that does not exist in the library

3. addVenue**Input validation tests:**

- venueName is null
- venueName is empty
- venueInformation is null
- venueInformation is empty
- venueInformation does not contain any mandatory fields
- researchAreas is null
- One of the research areas in the researchAreas set is null
- One of the research areas in the researchAreas set is empty

Boundary tests:

- Add a venue to the library for the first time
- Add a venue with a single research area
- Add a venue with research areas that are subsets of other research areas in the library

Control flow tests

- Add a venue that already exists in the library
- Add a venue, then retrieve the venue's information and verify that the venue was added
- Add a venue with the same name as a previously added venue but with different information
- Add a venue with the same name and information as a previously added venue
- Add a venue, then add a publication that references it and attempt to retrieve the publication's information and verify that the venue is correctly referenced

4. addPublisher**Input validation tests:**

- identifier is null
- identifier is empty
- publisherInformation is null
- publisherInformation is empty

Boundary tests:

- Add a publisher to the library for the first time
- Add a publisher to the library after adding other publishers

Control flow tests:

- Add a publisher that already exists in the library
- Add a publisher, then retrieve the publisher's information and verify that the publisher was added
- Add a publisher with the same identifier as a previously added publisher but with different information
- Add a publisher with the same identifier and information as a previously added publisher
- Add a publisher, then add a publication that cites it and attempt to retrieve the publication's information and verify that the publisher is correctly cited

5. addArea**Input validation tests:**

- researchArea is null
- researchArea is empty
- parentArea is null
- One of the areas in the parentArea set is null
- One of the areas in the parentArea set is empty

Boundary tests:

- Add a research area to the library for the first time
- Add a research area with no parent areas
- Add a research area with one parent area
- Add a research area with multiple parent areas

Control flow tests:

- Add a research area that already exists in the library
- Add multiple research areas consecutively
- Add a research area with no parent areas, then retrieve the research area's information and verify that the research area was added
- Add a research area with one parent area, then retrieve the research area's information and verify that the research area was added with the parent area
- Add a research area with multiple parent areas, then retrieve the research area's information and verify that the research area was added with the parent areas
- Add a research area, then add a publication that is related to it and attempt to retrieve the publication's information and verify that the research area is correctly related to the publication

6. `getPublications`

Input validation tests:

- key is null
- key is empty

Boundary tests:

- Retrieve the information from empty library
- Retrieve the information for a publication that was just added to the library
- Retrieve the information for a publication that does not exist in the library

Control flow tests:

- Retrieve the information for a publication that has no references
- Retrieve the information for a publication that references another publication in the library
- Retrieve the information for a publication that references multiple publications in the library
- Retrieve the information for a publication that references a publication that does not exist in the library
- Retrieve the information for a publication, then add a new reference to the publication and verify that the reference is correctly added to the publication's information

7. `authorCitations`

Input validation tests:

- author is null
- author is empty

Boundary tests:

- Retrieve information from an empty library
- author does not correspond to any authors in the publications in the library
- author has zero citations in the library
- library has only one publication with the given author
- library has multiple publications with the given author
- library has no publications with the given author

Control flow tests:

- Verify that the method correctly counts multiple citations of the same author in the same publication
- Verify that the method correctly counts citations from publications that reference the given author
- Verify that the method does not count citations from publications that do not reference the given author

- Verify that the method works correctly when the library is empty and no author citations can be counted

8. seminalPapers

Input validation tests:

- area is null
- area is empty
- paperCitation is negative
- otherCitations is negative

Boundary tests:

- There are no publications in the library
- There is only one publication in the library that matches the area and citation criteria

Control flow tests:

- Verify that the method correctly identifies publications that match the area and citation criteria
- Verify that the method does not return any papers that cite more than the specified number of papers in the given area
- Verify that the method works correctly when there are multiple publications that match the area and citation criteria
- Verify that the method works correctly when there are no publications that match the area and citation criteria, and returns an empty set

9. collaborators

Input validation tests:

- author is null
- author is empty
- distance is negative

Boundary tests:

- There are no authors in the library
- The given author has no co-authors in the library
- The given author has co-authors in the library, but none of them are reachable within the specified author distance

Control flow tests:

- Verify that the method correctly identifies all co-authors of the given author within the specified distance
- Verify that the method works correctly when there are no co-authors of the given author within the specified distance

- Verify that the method correctly handles cases where the specified distance is 0, and only returns the given author if they have published papers

10. authorResearchAreas

Input validation tests:

- author is null
- author is empty
- threshold is negative

Boundary tests:

- There are no research areas in the library
- There is only one research area in the library, and the threshold is greater than 0
- There are multiple research areas in the library, but none of them have papers by the given author
- There are multiple research areas in the library, and the given author has papers in some but not all of them
- The given author has papers in all research areas in the library, and the threshold is set to 0

Control flow tests:

- The given author has papers in all research areas, and the threshold is set to a value equal to the number of papers they have in at least one research area
- The given author has papers in all research areas, and the threshold is set to a value greater than the number of papers they have in any single research area
- The given author has papers in some research areas, but the threshold is set to 0
- The given author has papers in some research areas, and the threshold is set to a value equal to the number of papers they have in at least one research area
- The given author has papers in some research areas, and the threshold is set to a value greater than the number of papers they have in at least one research area, but less than the number of papers they have in all research areas

Data flow tests for PublicationLibrary

- Adding publications and fetching their information
- Adding publications
- Adding publication, updating references, and fetching information
- Adding publication, research area, and checking author research areas
- Adding publications, updating references, and checking author citations
- Adding publications, updating references, and retrieving author collaborators
- Adding publications, , venues, publishers, and updating references, and then retrieving seminal papers