THE CITY LIBRARY

CS 631

Data Management System Design

Deliverable 2

Project By:

Sahaj Singh Marwah (ssm226@njit.edu)
Chintan Modi (cvm6@njit.edu)
Charanpreet Kaur Dhir (ckd22@njit.edu)

Under Professor:

Dimitri Theodoratos
(dth@njit.edu)

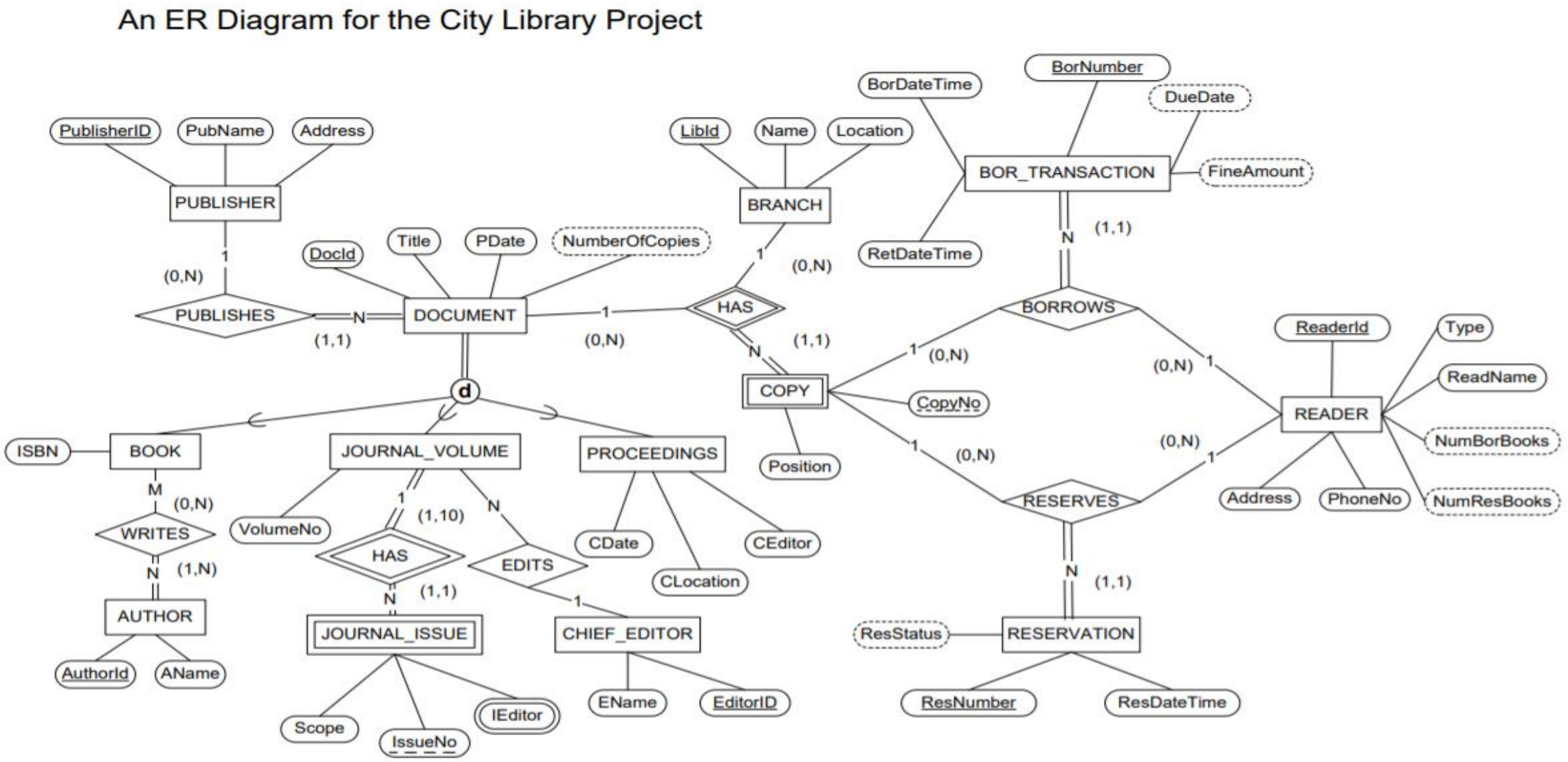
1. a. Goal for this phase of the project

To map EER schema which was submitted in deliverable 1 to the Relational schema for city library. In addition to this, the modification made to the EER diagram have also been enlisted in this phase.

1. b. Modification of EER diagram

- There was no separate entity for *copy* so we modified our existing attribute into an entity(Weak Entity). Also, we added two different attributes to *copy* entity namely *Copy number* as partial key and position.
- We Modified the *has* relation into a ternary relation. We connected it to entities namely *copy, branch* and *document.*
- We removed our *holds* relation which was not required and further its attributes were added into the *copy* entity.
- We added *Bor_transaction* entity and added attributes to it which we removed from our *BORROWS* relation. Also, we removed *due date* and *fine amount* from reader entity and added to the *Bor_transaction* entity.
- We removed *ONLINE CATALOGUE* entity from our model and instead created a new entity called reservation. This new entity has attributes *ResNumber, ResStatus* and *ResDateTime*. This entity is connected to the relation Reserves which is further connected to two entities namely *COPY* and *READER*.

ER DIAGRAM USED AS REFERENCE PROVIDED BY PROFESSOR



EER to Relational mapping

a. Illustration of translated from the EER diagram to your relational schema.

Following steps were followed to change the EER diagram to relational schema.
Step 1: For each regular (non-weak) entity type in the ER schema, we created a relation that includes all the simple (single-valued) attributes of the given entity type.

Step 2: For each weak entity type in the ER schema with owner entity type , we created a relation and included all simple (single-valued) attributes (or simple components of composite attributes) of weak entity type as attributes of relation. In addition, we included as foreign key attributes of Relation the primary key attribute(s) of the owner entity type(s).

Step 3: For each binary 1:1 relationship type Relation in the ER schema, we identified the relations that correspond to the entity types participating in relation. For every 1:1 relationship we chose one of the relations and include as foreign key in that relation to the primary key of other relationship. We included all the attributes of 1:1 relationship.

Step 4: For each regular binary 1:N relationship type, we identified the relation that represents the participating entity type at the N-side of the relationship type. We included as foreign key in the primary key of the relation that represents the other entity type participating in relation. We included simple (single-valued) attributes (or simple components of composite attributes) of the 1:N relationship type.

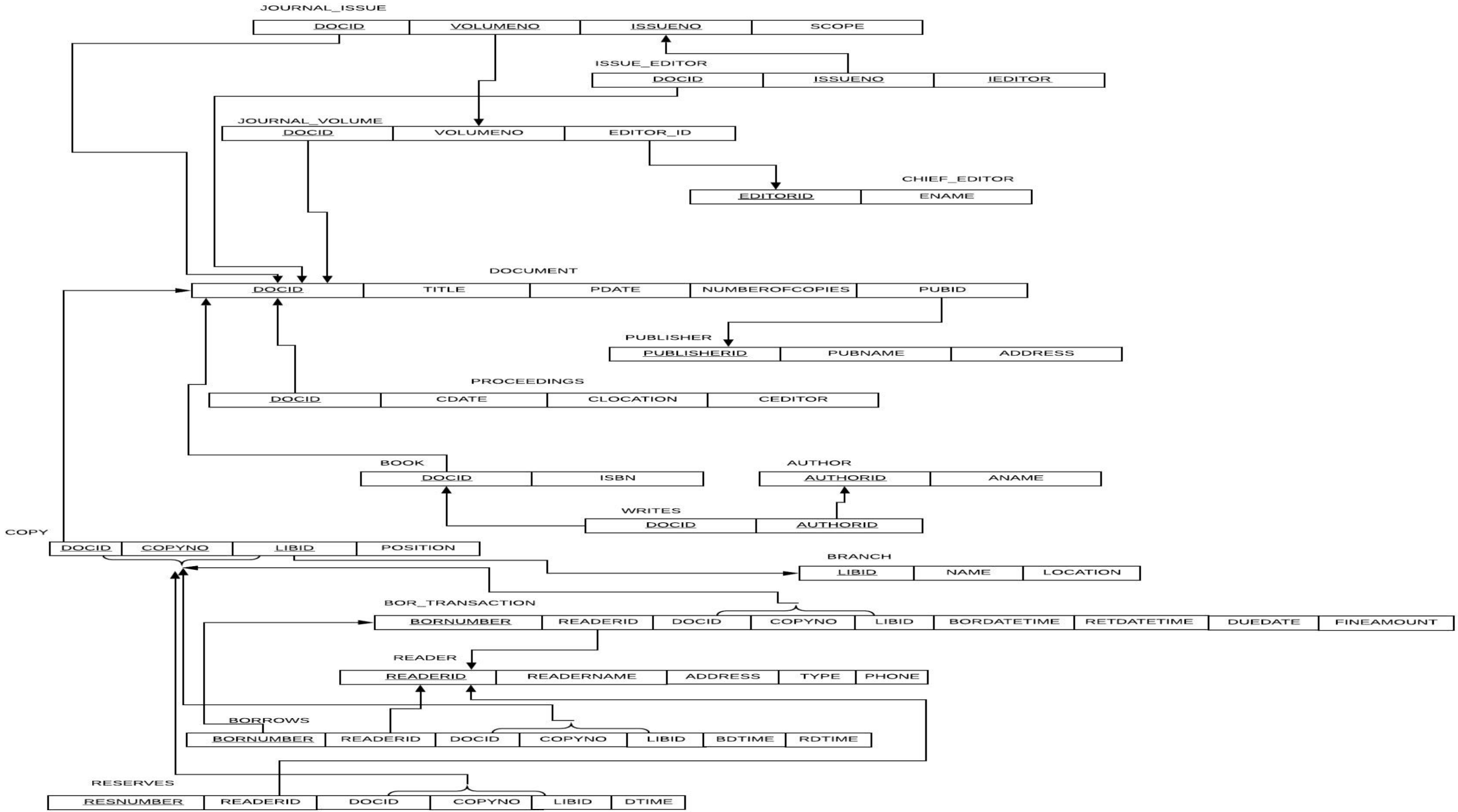
Step 5: For each binary M:N relationship type , we created a new relation that represents the relationship type. Include as foreign key attributes in the primary keys of the relations that represent the participating entity types. We also include as attribute of any simple (single-valued) attribute (or simple component of composite attribute) of the M:N relationship type.

Step 6: For each multivalued attribute we created a new relation. This relation will include an attribute corresponding to some attribute, plus the primary key attribute K -- as a foreign key in relation-- of the relation corresponding to the entity type or relationship type that has an attribute.

Step 7: For each regular n-ary relationship, where n > 2, we created a new relation to represent n-ary relationship. Included as foreign key attributes in new relation the primary keys of the relations that represent the participating entity types. Also we included any simple (singlevalued) attributes (or simple component attributes of a composite attribute) of the n-ary relationship type as attributes to new relationship.

Step 8: Attrs(R) denote the attributes of relation R PK(R) denote the primary key of R. We assumed a superclass C with m subclasses S_1, …, S_m. C has attributes K, A_1, …, A_n. K is the key of C. Create a relation L for C with attributes Atts(L) = {K, A_1, …, A_n }, and PK(L) = K.
We Create a relation L_i for each subclass S_i, 1 <= i <= m such that Atts(L_i) = {K} ? {attributes of Si}, PK(L_i) = K, and K is a foreign key to L.

Relational Schema of City Library Database System



Primary Keys and Foreign Keys identified in Relational schema

Primary Keys are:

- DocID, VolumeNo and *IssueNo* in *JOURNAL_ISSUE* entity
- DocID, IEDITOR and *IssueNo* in *ISSUE_EDITOR* entity
- *VolumeNo* in *JOURNAL_VOLUME* entity
- *EditorID* in *CHIED_EDITOR* entity
- *DocID in Document entity*
- *DocID in PROCEEDINGS entity*
- *PublisherID* in *Publisher* entity
- *DocID in BOOK entity*
- *DocID and AUTHORID in WRITES entity*
- *AUTHORID in AUTHOR entity*
- *LIBID in BRANCH entity*
- *RESNUMBER in RESERVES entity*
- *DOCID, LIBID and COPYNO in COPY entity*
- *BORNumber in BORROWS entity*
- *BORNumber in BOR_TRANSACTION entity*

- ***Foreign Keys are:***

- *DOCID* and *VOLUMENO* in *JOURNAL_ISSUE* entity
- *ISSUENO* and *DOCID* in *ISSUE_EDITOR* entity
- *DOCID* and *EDITOR_ID* in *JOURNAL_VOLUME* entity
- *PUBID* in *DOCUMENT* entity
- *DOCID* in *PROCEEDINGS* and *BOOKS* entity
- *DOCID* and *AUTHORID* in *WRITES* entity
- *LIBID* in *COPY* entity
- *DOCID, READERID, COPYNO, LIBID* in *BOR_TRANSACTION* entity
- *BORNUMBER, READERID, DOCID, COPYNO, LIBID* in *BORROWS* entity
- *READERID, DOCID, COPYNO, LIBID* in *RESERVES* entity

Referential Integrity Constraints:

1. A journal volume can have issues upto 10 .

2. A reader cannot borrow or reserve more than 10 documents.

3. A reserved book has to be picked up before 6 pm

4. A document should be returned within maximum of 20 days.

3. Difficulties Identified

During relational schema designing of City Library Database System, We faced difficulties in:

1. Deciding primary and foreign keys which we sorted using the lecture slides.

2. Defining relationships between few entities.

3. Showing constraints for referential integrity constraints for each table