

WolfPubDb

for the *WolfCity* publishing house

CSC540 Database Management Systems
Project 1

Vamsi Varada, Aswin Itha, Mohan Kumar, Ishwari Zare
February 15, 2022

Assumptions:

1. Each article is published only in one issue.
2. All articles belong to an issue.
3. Each chapter belongs to exactly one book.
4. Shipping cost is borne by WolfPublications.
5. We are not keeping track of payment for individual orders but the total outstanding balance.
6. Periodicity is defined only for an issue.
7. Every author/editor is always working on some or the other book/issue at any given time
8. Authors are only for Books and Issues and not for Individual chapters.
9. Table of contents and text of articles are a single attribute
10. Every staff author/editor gets a fixed salary and invited author/editors are paid differently.

1. Problem Statement:

There is a publishing house named WolfCity Publication House. It wants to store and maintain the data related to all its operations. These include details of the orders placed by its customers who may be distributors, retailers or libraries. Details of all the employees and distributors that are associated with the house. Salaries are being paid to all the employees and authors/editors who may be permanent employees or hired temporarily. Transactions are being made to the publishing house by distributors.

Traditional File systems allow access to only single files. The issues with traditional file systems such as data redundancy, inconsistency, data searching, integrity, data concurrency etc are eliminated using Database management System. DBMS allows controlled access to the data and makes sure that the data is consistent . It allows you to define correct rules for managing database. As the data is added, edited and accessed by multiple users who have to have different permissions, it is best to use a database rather than storing data in files.

2. Intended users

- Editing team: Manage information regarding a new publication and the contents of the publication. The editors can view the publication information.
- Administrative Staff (Like Managers and Work assigners): Complete control over WolfPubDb. They have access to all publication information, transaction information, staff information etc.
- Analytics: Generate monthly reports, number and total price of copies of each publication bought per distributor per month, total revenue, total expense.
- Sales: Enter new distributor, update distributor information, delete distributor input orders from distributors per distributor per date., billing distributor and changing their outstanding balance.

3. Five main entities

1. Publication: Publication ID, title, topics, periodicity, type
2. Distributors: distributor ID, type, city, phone number, point of contact, outstanding balance
3. Employee: Employee ID, name, age, gender, phone number, address, job title, date of joining
4. Order: order ID, date of order placed, delivery date, price of order, shipping cost
5. Transaction: Transaction Id, type, amount, payment mode, date of payment

4. Task and Operations - Realistic Situations

- **Situation 1:** Publishing a book by an invited editor: Assign an invited editor to the new publication. Enter basic info on the new publication. (Editing and publishing). Update publication information. Enter payment for the invited author and keep track of when each payment was claimed by its addressee. (Production of a book edition or of an issue of a publication)
- **Situation 2:** Distributor placing an order: This involves two/many operations i.e. find books and articles by topic, date, author's name (Production of book edition or of an issue of publication) and input order from distributors by date (Distribution). First we are checking if the book is available. After that, we need to bill the distributors. When the distributor pays for the order, change the outstanding balance of the distributor on receiving payment.

5. Application Program Interfaces(APIs)

- **Editing and Publishing:**

§ createPublication(title, type, topic, cost)

return confirmation with generated PID.

§ updatePublication(pubID, title, type, topic, cost)

return confirmation

- If there are NULL values for any of the fields, then they will not be updated.

§ assignEditorsToPublication(PID, empID)

return confirmation

§ getPublicationsByEmpId(empID)

return all publications (pubID, title, type, topic) based on employee ID if he is an editor/author. else returns empty list

§ updateTableOfContentsForPublication(pubID, list of articles/chapters) returns confirmation after updating the table of contents. This will only update the book with the given list. No append operation.

- **Production of a book edition or of an issue of a publication: §**

createBookEdition(ISBN, edition, dateOfCreation, publicationDate, tableOfContents, PID)

returns confirmation.

§ createIssue(issueNumber, dateOfIssue, type, tableOfContents, periodicity, PID)

returns confirmation.

§ updateBookEdition(ISBN, edition, dateOfCreation, publicationDate, tableOfContents)

updates based on ISBN and return confirmation

- If there are NULL values for any of the fields, then they will not be updated.

§ updateIssue(issueNumber, dateOfIssue, type, tableOfContents, periodicity)

updates based on issueNumber and return confirmation

- If there are NULL values for any of the fields, then they will not be updated.

§ deleteBookEdition(ISBN)

returns confirmation after deletion

§ deleteIssue(issueNumber)

returns confirmation after deletion

§ createChapter(ISBN, chapterName, content)

returns confirmation

§ createArticle(issueNumber, articleName, articleText, dateOfCreation)
returns confirmation

§ updateChapter(ISBN, chapterName, authorName, topic, date)
returns confirmation after updating based on chapterName and ISBN §
updateArticle(issueNumber, articleName, topic, date)
returns confirmation after updating based on articleName and
issueNumber

§ deleteChapter(PID)
returns confirmation after deletion
§ deleteArticle(PID)
returns confirmation after deletion

§ updateArticleText(issueNumber, articleName, articleText)
updates text of the article and returns confirmation
§ updateChapterText(ISBN, chapterName, contents)
updates content of the chapter and returns confirmation
§ getBooksByTopic(type, topic)
returns list of books based on topics
§ getArticlesByTopic(type, topic)
returns list of all articles based on topic
§ getBooksByDate(type, date)
returns list of all books based on date
§ getArticlesByDate(type, date)
returns list of all articles based on date
§ getBooksByAuthor(type, author)
returns list of all books based on author
§ getArticlesByAuthor(type, author)
returns list of all articles based on author

§ createNewAuthorPayment(empID)
returns confirmation by generating Transaction ID (TID)
§ showPaymentStatus(TID)
returns TransactionDate

- **Distribution:**

- § createDistributor(name, phoneNumber, pointOfContact, city, type)
returns confirmation with generated Distributor ID (DID)

- § updateDistributor(DID, name, phoneNumber, pointOfContact, city, type)

updates based on DID and return confirmation

- If there are NULL values for any of the fields, then they will not be updated.

§ deleteDistributor(DID)

returns confirmation after deletion

§ createOrder(deliveryDate, shippingCost, DID)

returns confirmation with generated orderID

§ addPublicationToOrder(orderID, PID, numberOfCopies)

returns confirmation after adding publicationID into order and also updating the priceCost in orderID

§ generateBillForDistributor()

returns confirmation after updating outstanding balance for distributor by adding the price of the orderID

- **Reports:**

§ generateMonthlyReportForPublications(startDate, endDate)

returns confirmation after generating a new excel created in a pre-defined folder with a report on number of copies and total price of copies of each publication bought per distributor per month.

§ generateReportOnRevenue(startDate, endDate)

returns confirmation after generating a new excel created in a pre-defined folder with a report on monthly revenue.

§ generateReportOnExpense(startDate, endDate)

returns confirmation after generating a new excel created in a pre-defined folder with a report on monthly expense

§ generateDistributorCount()

returns confirmation after calculating the count of the total number of distributors with the publication.

§ generateReportOnRevenuePerCity(startDate, endDate)

returns confirmation after calculating the total number of incoming transactions related to per city

§ generateReportOnRevenuePerDistributor(startDate, endDate)

returns confirmation after calculating the total number of incoming transactions related to per distributor

`$generateReportOnRevenuePerLocation(startDate, endDate)`

returns confirmation after calculating the total number of transactions related to per location

`$generateReportOnRevenuePerTimePeriod(startDate, endDate)`

returns confirmation after calculating the total number of transactions based on specified time period

`$generateReportOnRevenuePerWorkType(startDate, endDate)`

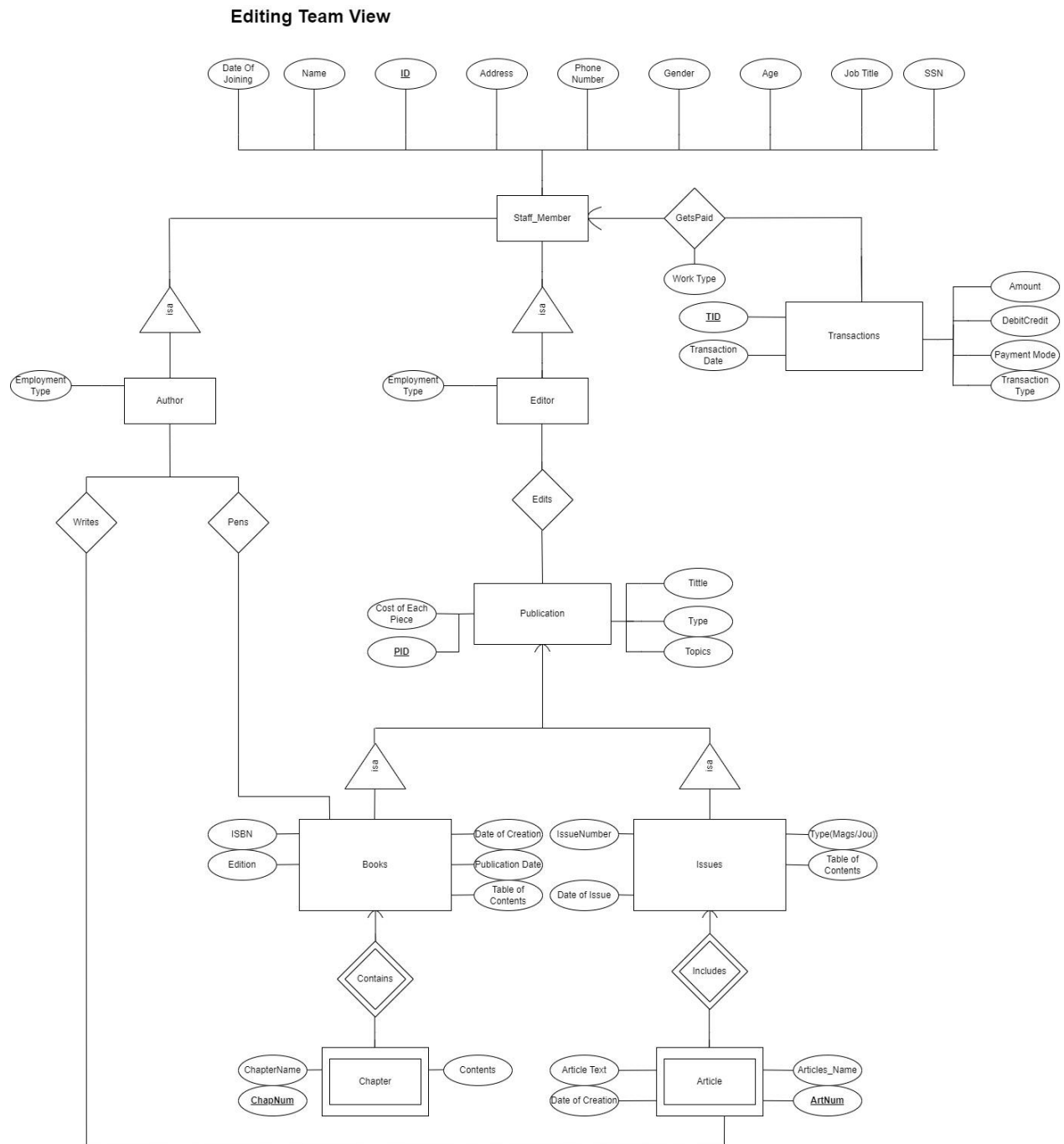
returns confirmation after calculating the total number of transactions related to per work type

6. Description of views

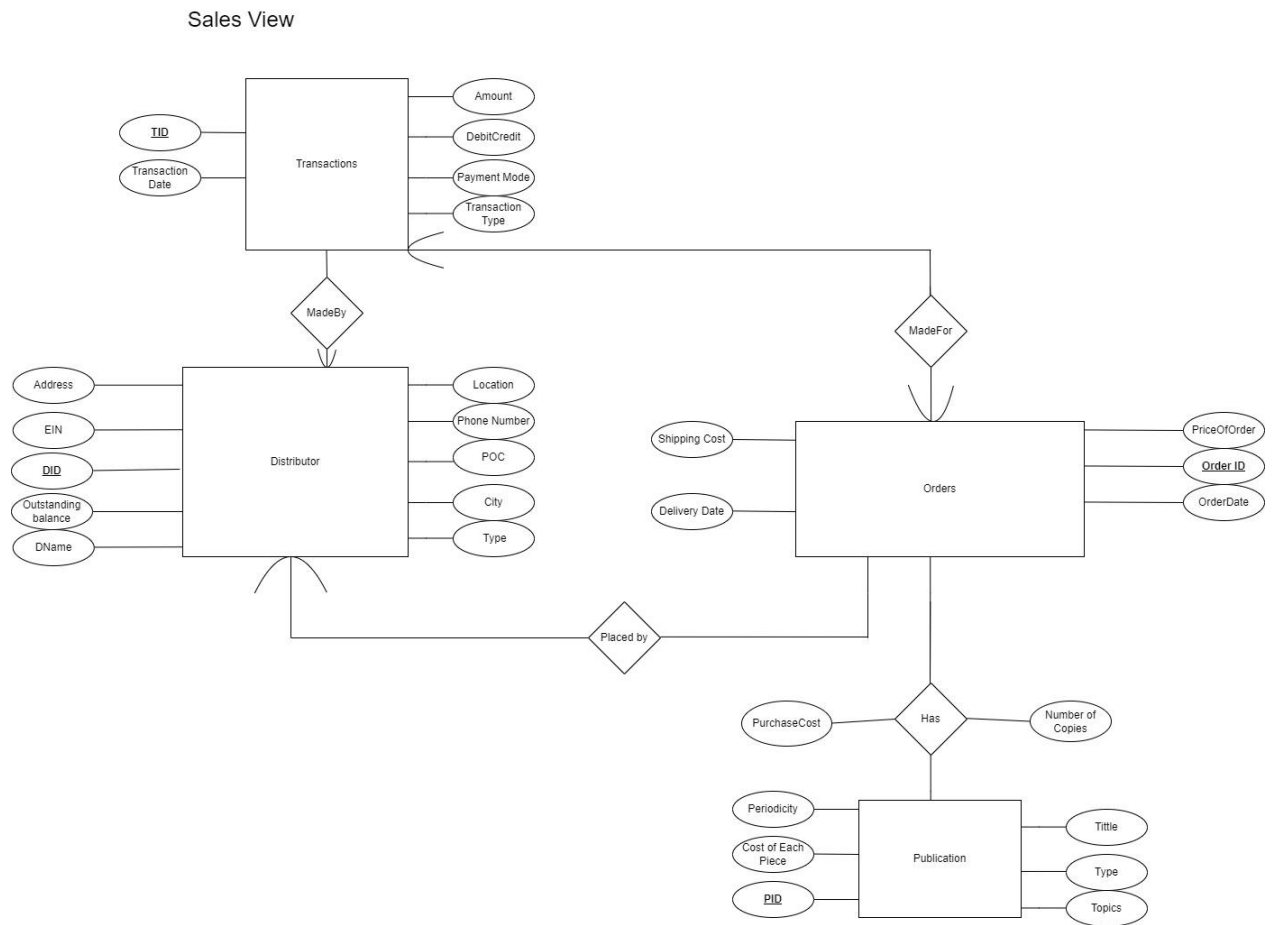
- Editing View: Editing View has READ permission to data related to Author as a StaffMember, CRUD of Publications, Books, Issues, Chapters and Articles. They have READ permission to the transactions related to Authors and Editors as StaffMembers.
- Sales View: CRUD permissions for Transactions, Distributor, Orders and Publication.
- Analytics View: READ permissions for Editor, Publications, Orders, Transactions, Distributor and Staff Members.
- Admin View: CRUD permissions for Staff Members, Authors, Editors, Publications, Books, Issues, Chapters, Articles, Transactions, Distributor, Orders.

7. Local E/R diagrams

Editing Team View

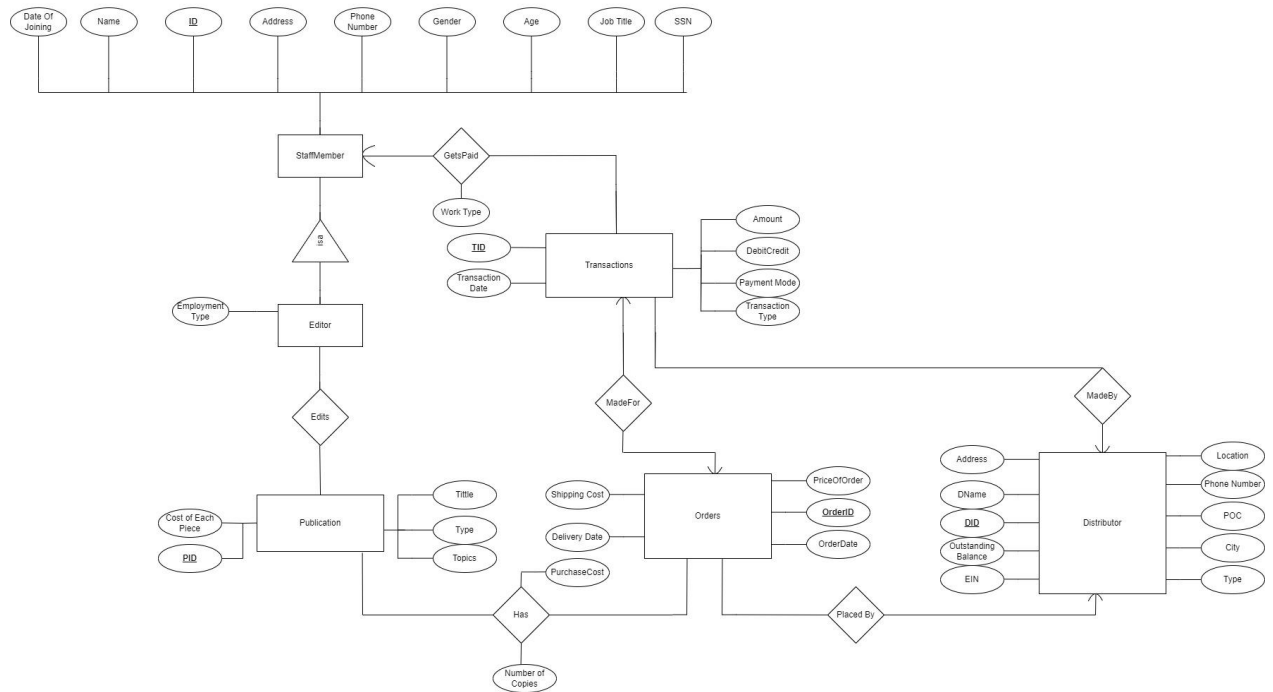


Sales View

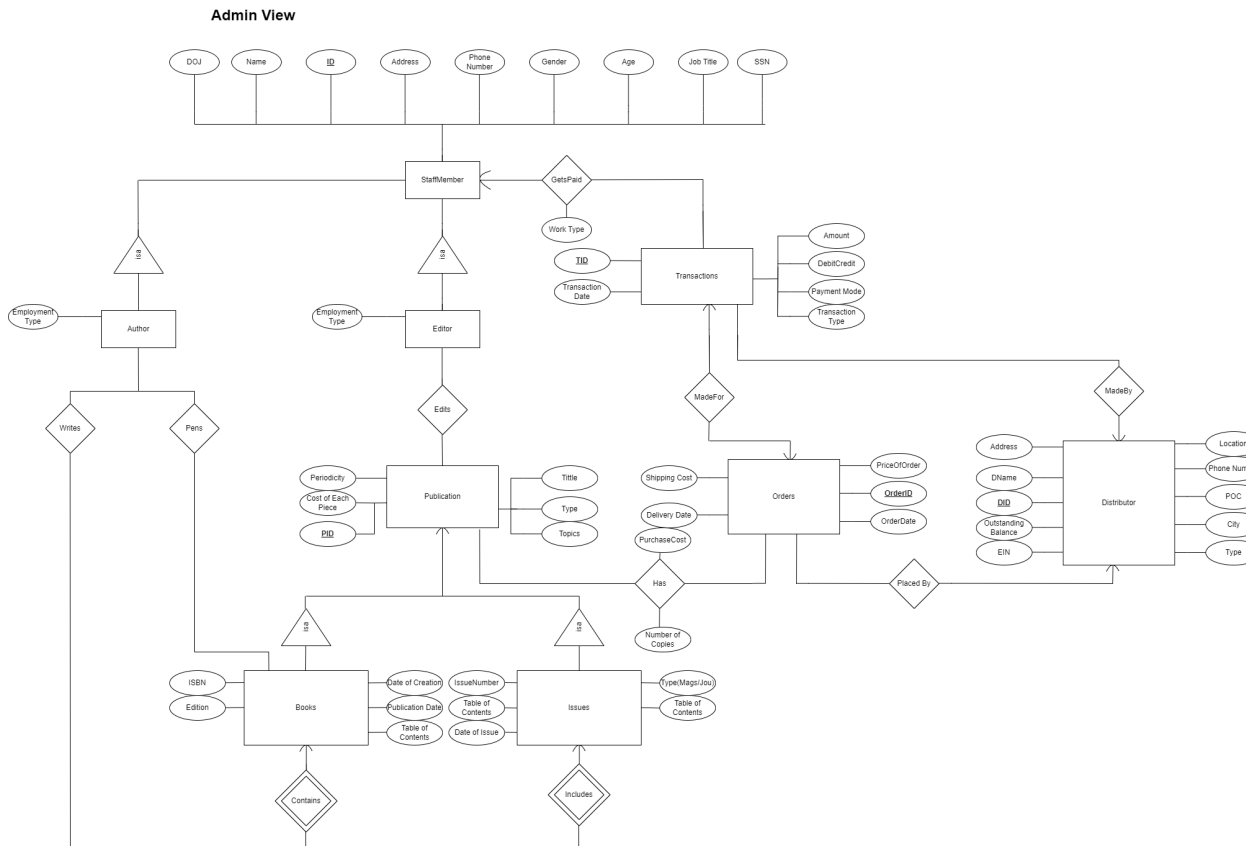


Analyst View

Analyst View



Admin View



8. Description of Local E/R diagrams

- Staff member is an author, editor, analyst or a sales team member -
The authors write books and articles for the issues.
- The editor publishes publications(books/issues) which are identified by the publication ID (PID) and has some other information associated with it.
- A publication is a book or an issue. A book is identified by ISBN and the issue is identified by an issueNumber.
- Issues have an attribute specifically defined for them named periodicity. -
An issue can be a journal or a magazine
- Every book has chapters associated with it which uses ISBN and ChapterNumber as the key to identify the chapters since no chapter can exist without a book.
- Every issue has articles associated with it which uses issueNumber and ArticleNumber as the key to identify the articles since no article can exist without an issue. - Chapters and articles are weak entities.
- Every chapter is present in only one book and every article is present in only one issue. -
Transactions are done for multiple reasons and are identified by their transaction ID (TID)
- Authors and editors who are permanent employees get paid (transactions) a fixed salary while invited authors or editors get paid for their work type(book authorship, article authorship or editorial work)
- Distributors are identified by distributor id (DID) and have some other information associated with them.
- Distributors place orders which are identified by order id(OID) and the cost is calculated based on the number of copies of a particular publication.
- Based on the payment made by the distributors (transactions) the outstanding balance is maintained accordingly.
- The publication house bears the shipping cost for orders and these transactions are identified by the TID and OID
- Each transaction given as salary is associated with only one staff member.

9. Local Relational Schema

- Admin

- StaffMember(ID, SSN, Name, Address, PhoneNumber, Gender, Age, JobTitle, DOJ)
 - Author(ID, EmploymentType)
 - Editor (ID, Employment type)
 - Sales(ID)
 - Analyst(ID)
 - Admin(ID)
- Writes(ID, ArtNum, PID)
- edits(ID, PID)
- Publication(PID, Title, Type, Topics, CostofEachPiece)
 - Books(PID, ISBN, Edition, PublicationDate, DateofCreation, TableofContents)
 - Chapter(PID, ChapNum, ChapterName, Contents)
 - Issues(PID, IssueNumber, Type, DateofIssue, Periodicity, TableofContents)
 - Articles(PID, ArtNum, ArticleName, ArticleText, DateofCreation)
- Transactions(TID, TransactionDate, Amount, DebitCredit, PaymentMode, TransactionType)
- GetsPaid(TID, ID, WorkType)
- MadeBy(TID, DID)
- MadeFor(TID, OrderID)
- Distributor(DID, EIN, DName, OutstandingBalance, Type, Address, City, Location, POC, PhoneNumber)
- Orders(OrderID, DID, OrderDate, PriceOfOrder, DeliveryDate, ShippingCost)
- Has(PID, OrderID, NumberofCopies, PurchaseCost)

- Editing team

- StaffMember(ID, SSN, Name, Address, PhoneNumber, Gender, Age, JobTitle, DOJ)
 - Author(ID, EmploymentType)
 - Editor (ID, Employment type)
 - Sales(ID)
 - Analyst(ID)
 - Admin(ID)

- Writes(ID, ArtNum, PID)
- Edits(ID, PID)
- Publication(PID, Title, Type, Topics, CostofEachPiece)
 - Books(PID, ISBN, Edition, PublicationDate, DateofCreation, TableofContents)
 - Chapter(PID, ChapNum, ChapterName, Contents)
 - Issues(PID, IssueNumber, Type, DateofIssue, Periodicity, TableofContents)
 - Articles(PID, ArtNum, ArticleName, ArticleText, DateofCreation)
- Transactions(TID, TransactionDate, Amount, DebitCredit, PaymentMode, TransactionType)

- Sales

- Transactions(TID, TransactionDate, Amount, DebitCredit, PaymentMode, TransactionType)
- GetsPaid(TID, ID, WorkType)
- MadeBy(TID, DID)
- MadeFor(TID, OrderID)
- Distributor(DID, EIN, DName, OutstandingBalance, Type, Address, City, Location, POC, PhoneNumber)
- Orders(OrderID, DID, OrderDate, PriceOfOrder, DeliveryDate, ShippingCost)
- Has(PID, OrderID, NumberofCopies, PurchaseCost)
- Publication(PID, Title, Type, Topics, CostofEachPiece)

- Analyst

- StaffMember(ID, SSN, Name, Address, PhoneNumber, Gender, Age, JobTitle, DOJ)
 - Author(ID, EmploymentType)
 - Editor (ID, Employment type)
 - Sales(ID)
 - Analyst(ID)
 - Admin(ID)
- Writes(ID, ArtNum, PID)
- Edits(ID, PID)
- Publication(PID, Title, Type, Topics, CostofEachPiece)
- Transactions(TID, TransactionDate, Amount, DebitCredit, PaymentMode, TransactionType)
- GetsPaid(TID, ID, WorkType)
- MadeBy(TID, DID)
- MadeFor(TID, OID)

- Distributor(DID, EIN, DName, OutstandingBalance, Type, Address, City, Location, POC, PhoneNumber)
- Orders(OrderID, DID, OrderDate, PriceOfOrder, DeliveryDate, ShippingCost)
- Has(PID, OrderID, NumberOfCopies, PurchaseCost)

10. Local Schema Documentation

- Entity sets to relationships

- The entity sets in our diagram were made into relations with the attributes the same for author, editor, sales, analyst and admin along with employment type added as attributes for the author, editor.

- Combining many-one relationships

- We combined the many to one relationships to reduce the redundancy and decrease the overhead
- We have chapters and articles as the weak entity sets. Weak entity sets were made into relation with all its attributes along with chapterName and contents for chapter and article Name and text for articles
- Every order is placed by only one distributor. Hence, their relationship is maintained in the order entity using the DID key.

- Relationships to relation

- Getspaid is a relation between transaction and staff member, worktype is an attribute added to it to keep track of book authorship, article authorship and editorial work