**Chapter 10 - Sample Project: Planning the Distribution of the Relational Database for The Art Gallery**

**Step 10.1 – Write out the normalized set of relations developed in Chapter 6 as the global schema.**

We will assume The Art Gallery has expanded to three locations: Midtown (which is the original gallery), Uptown, and Downtown. We wish to distribute the database among the three locations. We will use the normalized set of relations developed in Chapter 6 as the global schema. The schema is:

(1) **Artist** (artistId, firstName, lastName, interviewDate, interviewerName, *areaCode, telephoneNumber*, street, *zip*, salesLastYear, salesYearToDate, socialSecurityNumber, usualMedium, usualStyle, usualType)

(2) **Zips** (zip, city, state)

(3) **PotentialCustomer** (potentialCustomerId, firstname, lastName, areaCode, telephoneNumber, street, *zip,* dateFilledIn, *preferredArtistId,* preferredMedium, preferredStyle, preferredType)

(4) **Artwork** (artworkId, *artistId*, workTitle, askingPrice, dateListed, dateReturned, dateShown, status, workMedium, workSize, workStyle, workType, workYearCompleted, *collectorSocialSecurityNumber*)

(5) **ShownIn** (*artworkId,showTitle*)

(6) **Collector** (socialSecurityNumber, firstName, lastName, street, *zip*, interviewDate, interviewerName, areaCode, telephonenumber, salesLastYear, salesYearToDate, *collectionArtistId,* collectionMedium, collectionStyle, collectionType, SalesLastYear, SalesYearToDate)

(7) **Show** (showTitle, *showFeaturedArtistId,* showClosingDate, showTheme, showOpeningDate)

(8) **Buyer** (buyerId, firstName, lastName, street, zip, areaCode, telephoneNumber, purchasesLastYear, purchasesYearToDate)

(9) **Sale** (InvoiceNumber, *artworkId,* amountRemittedToOwner, saleDate, salePrice, saleTax, *buyerId, salespersonSocialSecurityNumber*)

(10) **Salesperson** (socialSecurityNumber, firstName, lastName, street, zip)

**Step 10.2 - Write out a set of end user locations and the applications performed at each.**

The three locations are Midtown (the main site), Uptown, and Downtown. The applications performed at each branch for that branch’s own data are:

* 1. Maintaining artwork records
  2. Producing sales invoice
  3. Maintaining sales records
  4. Maintaining the potential customer records
  5. Producing the Works for Sale report
  6. Producing the Sales This Week report
  7. Producing the Buyer Sales Report
  8. Producing the Preferred Customer Report
  9. Producing the Salesperson Performance Report
  10. Producing the Aged Artworks Report
  11. Producing the Owner Payment Stub
  12. Producing the Art Show Details report

In addition, the following applications are performed at Midtown only:

* 1. Maintaining artist records
  2. Maintaining collector records
  3. Producing the Active Artists Summary report
  4. Producing the Individual Artist Sales report
  5. Producing the Collectors Summary report
  6. Producing the Individual Collector Sales report

**Step 10.3 - For each application, decide what tables are required.**

1. Maintaining artwork records-Artwork, Artist, and Collector table
2. Producing sales invoice-Sale, Buyer, Salesperson, Artist, Collector, and Zip tables
3. Maintaining sales records-Sale, Buyer, Salesperson, and Artwork tables
4. Maintaining the potential customer records-PotentialCustomer and Artist tables
5. Producing the Works for Sale report-Artwork, Artist, and Collector tables
6. Producing the Sales This Week report-Sale, Salesperson, Buyer, Artist, and Collector tables
7. Producing the Buyers Sales Report-Buyer, Zip, Sale, Artwork, and Artist tables
8. Producing the Preferred Customer Report-PotentialCustomer, Zip, Buyer, and Artist tables
9. Producing the Salesperson Performance Report-Sale, Artwork, Artist, Salesperson and Zip tables
10. Producing the Aged Artworks Report-Artwork, Collector, and Artist tables
11. Producing the Owner Payment Stub-Sale, Collector, Zip, and Artist tables
12. Producing the Art Show Details report-Show, ShownIn, Artwork, and Artist tables
13. Maintaining artist records-Artist table
14. Maintaining collector records-Collector and Artist tables
15. Producing the Active Artists Summary report-Artist, Zip, Sale, and Artwork tables
16. Producing the Individual Artist Sales report-Artist, Zip, Sale, and Artwork tables
17. Producing the Collectors Summary report-Collector, Artist, Zip, Sale, and Artwork tables
18. Producing the Individual Collector Sales report-Collector, Artist, Zip, Sale, and Artwork tables

**Step 10.4 - Using the normalized relations, perform selection and projection operations, to create the set of vertical, horizontal and mixed data fragments needed for each application.**

**Artist**. Parts of table are used at all sites for all their applications (1 through 12). Only the artistId is needed for most of these, but the name is needed for a few. The required data could be produced by projecting the Artist table onto the required columns, creating a fragment we will call ArtistFragment1.

ArtistFragment1 = ΠartistId, firstName, lastName(Artist).

At each branch, the Owner Payment Stub also requires the artist’s name, address and social security number, if the artist is the owner. We form another fragment for this projection.

ArtistFragment2 = ΠartistId, firstName, lastName, street, zip, socialSecurityNumber(Artist).

Midtown needs the entire table to maintain the artist records (13), and the collector records (14), and for all its reports, 15 through 18.

**Zips**. Since the table is rarely updated and is needed at every location, we will replicate the entire table at each branch.

**PotentialCustomer.** Each branch will have its own potential customer records.

PotentialCustomerDowntown = σID LIKE ‘D%’ (PotentialCustomer)

PotentialCustomerMidtown = σID LIKE ‘M%’ (PotentialCustomer)

PotentialCustomerUptown =σID LIKE ‘U%’ (PotentialCustomer)

**Artwork**. Each branch needs this table to maintain records of artworks at that branch (1), and sales records (3). We can form these using a selection operation, if we assume the artworkId contains a code indicating the branch, as we did for PotentialCustomer. The fragments are identified using

ArtworkDowntown= σID LIKE ‘D%’(Artwork)

ArtworkMidtown = σID LIKE ‘M%’(Artwork)

ArtworkUptwn = σID LIKE ‘U%’(Artwork)

The branch also uses these same table fragments to produce its own Works for Sale report (5), Buyer Sales report (7), Salesperson Performance report 9), Aged Artworks report (10), and Art Show Details report (12).

Midtown also uses the entire table for reports 15, 16, 17, and 18.

**ShownIn**. Each branch uses this table for its Art Show Details report (12). However, it needs only the horizontal fragment for its own works and shows. We will add an attribute, branch, to the Show table, identifying the branch each show is in. The ShownIn table can then be fragmented horizontally, using and SQL subquery to identify the appropriate branch. For example, for the downtown branch, we create the horizontal fragment

ShownInDowntown as

SELECT \*

FROM ShownIn

WHERE showTitle IN (SELECT showTitle

FROM Show

WHERE branch =’Downtown’);

Similarly we form ShownInMidtown, and ShownInUptown.

**Collector**. Each branch uses this table for maintaining artwork records (1), producing the sales invoice (2), and for reports 5, 6, 10 and 11. For some of these, the collectorId and name are needed. We can fragment this table by using projection as we did for the Artist table

CollectorFragment1 = ΠcollectorId, firstName, lastName(Collector).

The Owner Payment Stub requires the collector’s name, address and social security number, if a collector is the owner. We form another Fragment for this projection.

CollectorFragment2 = ΠcollectorId, firstName, lastName, street, zip, socialSecurityNumber(Collector).

Midtown uses the entire table for maintaining collector records (14) and for reports 17 and 18.

**Show**. Each branch uses this table for its Art Show Details report (12). We will add an attribute, branch, to the Show table, and create fragments as follows

ShowDowntown = σbranch=’Downtown’(Show)

ShowMidtown= σbranch=’Midtown’(Show)

ShowUptown= σbranch=’Uptown’(Show)

**Buyer**. Each branch uses this table for producing sales invoices (2), maintaining sales records (3) and for reports 6, 7 and 8. Assuming the buyerId is a string containing a code for the branch where the buyer makes a purchase, we can use selection to form horizontal fragments

BuyerDowntown = σID LIKE ‘D%’ (Buyer)

BuyerMidtown = σID LIKE ‘M%’ (Buyer)

BuyerUptown =σID LIKE ‘U%’ (Buyer)

**Sale**. Each branch uses this table for producing sales invoices (2), maintaining its sales records (3), and for reports 6, 7, 9, and 11. Each branch can use its own set of invoice numbers, whose initial digit identifies the branch. We can create horizontal fragments to identify the branch for each sale, using

SaleDowntown = σinvoiceNumber>0 and invoiceNumer <20000 (Sale)

SaleMidtown = σ invoiceNumber>20000 and invoiceNumer <40000 (Sale)

SaleUptown =σ invoiceNumber>40000 and invoiceNumer <60000 (Sale)

Midtown also uses the entire table for reports 15, 16, 17, and 18.

**Salesperson**. Each branch uses this table for producing sales invoices (2), maintaining sales records (3), and for reports 6 and 9. We will add an attribute, branch, to the table to identify the branch a salesperson belongs to. Using selection, we form the horizontal subsets

SalespersonDowntown = σbranch=’Downtown’(Salesperson)

SalespersonMidtown= σbranch=’Midtown’(Salesperson)

SalespersonUptown= σbranch=’Uptown’(Salesperson)

**Step 10.5 - Map the fragments to the applications and locations.**

For each fragment that is required at more than one application location, decide whether the fragment can be replicated, by considering frequency of use and of update.

**Artist**.

The Artist table will be updated very infrequently, and artist ID and name appear in many applications at all branches. Therefore, we will choose to replicate the fragment ArtistFragment1 at all branches. Midtown needs the entire table for its applications, so we will store the entire table there. We note that ArtistFragment2 (ΠartistId, firstName, lastName, street, zip, socialSecurityNumber(Artist)) contains sensitive data, the artist’s address and social security number, and is needed at branches only for the owner payment stub when the artist is the owner. Therefore, we would choose to keep this fragment only at the Midtown branch, and allow other branches to access it as needed. However, since Midtown already has the entire Artist table, we will create a view to replace ArtistFragment2, allows branches to access the view when payments are made to artists, and not create the fragment.

**Zips**. This table is needed at every location, is rarely updated, and does not contain any sensitive date, so we replicate it everywhere.

**PotentialCustomer.** Each branch stores data about its own potential customers, using the fragments, PotentialCustomerDowntown, PotentialCustomerMidtown, and PotentialCustomerUptown

**Artwork**. Each branch stores records about is own artworks, using fragments ArtworkDowntown, ArtworkMidtown, and ArtworkUptwn. Midtown stores a copy of the the entire table.

**ShownIn**. Each branch uses the fragment of this table for its own shows, namely ShownInDowntown, ShownInMidtown, and ShownInUptown.

**Collector**. Each branch stores a copy of CollectorFragment1, which has Id and name. Midtown stores the entire table and provides access to CollectorFragment2 as a view on that table.

**Show**. Each branch stores the records of its own shows, namely

ShowDowntown, ShowMidtown, and ShowUptown.

**Buyer**. Each branch stores the records of its own buyers, namely BuyerDowntown,

BuyerMidtown, and BuyerUptown.

**Sale**. Each branch has its own sales records, identified by invoice number SaleDowntown, SaleMidtown, and SaleUptown . Midtown also keeps a copy of the entire table.

**Salesperson**. Each branch keeps records of its own salespersons, namely SalespersonDowntown, SalespersonMidtown, and SalespersonUptown.

**Step 10.6 - Make a table showing a geographical network, listing nodes and applications and showing the data fragments at each node.**

The table is shown in Figure S.10.1.

|  |  |  |  |
| --- | --- | --- | --- |
| APPLICATION | Downtown | Midtown | Uptown |
| 1 Maintaining Artwork Records | ArtworkDowntown  ArtistFragment1  CollectorFragment1 | Artwork  Artist  Collector | ArtworkUptown  ArtistFragment1  CollectorFragment1 |
| 2 Producing Sales Invoice | SaleDowntown  BuyerDowntown  SalespersonDowntown  ArtistFragment1  CollectorFragment1  Zip | Sale  BuyerMidtown  SalespersonMidtown  Artist  Collector  Zip | SaleUptown  BuyerUptown  SalespersonUptown  ArtistFragment1  CollectorFragment1  Zip |
| 3 Maintaining Sales Records | SaleDowntown  BuyerDowntown  SalespersonDowntown  ArtworkDowntown | Sale  BuyerMidtown  SalespersonMidtown  Artwork | SaleUptown  BuyerUptown  SalespersonUptown  ArtworkUptown |
| 4 Maintaining Potential Customer  Record | PotentialCustomerDowntown  ArtistFragment1 | PotentialCustomerMidtown  Artist | PotentialCustomerUptown  ArtistFragment1 |
| 5 Works for Sale Report | ArtworkDowntown  ArtistFragment1  CollectorFragment1 | Artwork  Artist  Collector | ArtworkUptown  ArtistFragment1  CollectorFragment1 |
| 6 Sales This Week | SaleDowntown  BuyerDowntown  SalespersonDowntown  ArtworkDowntown | Sale  BuyerMidtown  SalespersonMidtown  Artwork | SaleUptown  BuyerUptown  SalespersonUptown  ArtworkUptown |
| 7 Buyers Sales Report | BuyerDowntown  Zip  SaleDowntown  ArtworkDowntown  ArtistFragment1 | BuyerMidtown  Zip  SaleMidtown  ArtworkMidtown  Artist | BuyerUptown  Zip  SaleUptown  ArtworkUpown  ArtistFragment1 |
| 8 Preferred Customer Report | PotentialCustomerDowntown  Zip  BuyerDowntown  ArtworkDowntown  ArtistFragment1 | PotentialCustomerMidtown  Zip  BuyerMidtown  ArtworkMidtown  Artist | PotentialCustomerUptown  Zip  BuyerUptown  ArtworkUptown  ArtistFragment |
| 9 Salesperson Performance  Report | SalespersonDowntown  SaleDowntown  ArtworkDowntown  ArtistFragment1  Zip | SalespersonMidtown  SaleMidtown  ArtworkMidtown  Artist  Zip | SalespersonUptown  SaleUptown  ArtworkUptown  ArtistFragment1  Zip |
| 10 Aged Artworks Report | ArtworkDowntown  CollectorFragment1  ArtistFragment1 | ArtworkMidtown  Collector  Artist | ArtworkUptown  CollectorFragment1  ArtistFragment1 |
| 11 Owner Payment Stub | SaleDowntown  CollectorFragment1  Zip  ArtistFragment1 | SaleMidtown  Collector  Zip  Artist | SaleUptown  CollectorFragment1  Zip  ArtistFragment1 |
| 12 Art Show Details Report | ShowDowntown  ShownInDowntown  ArtworkDowntown  ArtistFragment1 | ShowMidtown  ShownInMidtown  ArtworkMidtown  Artist | ShowUptown  ShownInUptown  ArtworkUptown  ArtistFragment1 |
| 13 Maintaining Artist Records |  | Artist |  |
| 14 Maintaining Collector Records |  | Collector |  |
| 15 Active Artists Summary Report |  | Artist, Zip, Sale, Artwork |  |
| 16 Individual Artist Sales Report |  | Artist, Zip, Sale, Artwork |  |
| 17 Collectors Summary Report |  | Collector, Artist, Zip, Sale, Artwork |  |
| 18 Individual Collector Sales |  | Collector, Artist, Zip, Sale, Artwork |  |

Figure S.10.1 - Geographical Network for The Art Gallery

**Step 10.7 - For each application in the geographical network, determine whether access will be local, remote, or compound.**

Make up a table showing each site, and the applications requiring local access, remote access, and compound access. The table is shown in Figure S.10.2

**Step l0.8 - For each of the non-local accesses, identify the application and the location of the data.**

Estimate the number of accesses required per day. If it is high, justify your choice of non-local storage. The only applications requiring remote access are the sales invoice and the owner payment stub applications, which require that the branches access the midtown location to determine the name, address, and social security number of the owner of the artwork. We have decided to maintain these in only one site for privacy reasons. The volume will correspond to the number of sales in each site. For original artwork of the type offered at The Art Gallery, the number of transactions per day will not be large.

**Step 10.9 - Make any adjustments indicated by your analysis of applications and traffic, and plan a final geographical network.**

Since most accesses are local, there is no need to adjust the geographical network shown in Figure S.10.2

|  |  |  |  |
| --- | --- | --- | --- |
| APPLICATION | Downtown | Midtown | Uptown |
| 1 Maintaining Artwork Records-  all local access | local | local | local |
| 2 Producing Sales Invoice-  remote access from branches for owner address, phone, and social security number | remote | local | remote |
| 3 Maintaining Sales Records-  all local access | local | local | local |
| 4 Maintaining Potential Customer  Records-all local access | local | local | local |
| 5 Works for Sale Report-  all local access | local | local | local |
| 6 Sales This Week-  all local access | local | local | local |
| 7 Buyers Sales Report-  all local access | local | local | local |
| 8 Preferred Customer Report-  all local access | local | local | local |
| 9 Salesperson Performance  Report-all local access | local | local | local |
| 10 Aged Artworks Report-all local access | local | local | local |
| 11 Owner Payment Stub-  remote access from branches for owner address, phone, and social security number | remote | local | remote |
| 12 Art Show Details Report-  all local access | local | local | local |
| 13 Maintaining Artist Records-  local access | local | local | local |
| 14 Maintaining Collector Records-local access | local | local | local |
| 15 Active Artists Summary Report-local access | local | local | local |
| 16 Individual Artist Sales Report-local access | local | local | local |
| 17 Collectors Summary Report-local access | local | local | local |
| 18 Individual Collector Sales-local access | local | local | local |

Figure S.10.2 - Applications with Local, Remote, and Compound Access