Herman Ostrow School of Dentistry of USC
Database Project

Team Data Dentists * Ji Young Oh, Huy Ngo, Wilson Lin * DSO 435 * Professor Shook * December 11, 2015

Table of Contents

A. Introduction

B. Planning for the Database

C. Conceptual Design

D. Logical Design with the Relational Model

E. Physical Design and Implementation with the Relational Model

A. Executive Summary

This report entails the design of a database for the business office at Herman Ostrow School of Dentistry of USC including Planning, Conceptual, Logical and Physical elements. Dentistry School serves as institution that educate students to be dentists and they are using large amount of materials each day. This project started from student worker who was not satisfied with business process at the office. Each invoice and its detail was entered by Employee and not everyone in the office could not capture the update on Purchase Orders. By having good communication cannel through student worker, it was flawless to get information we needed.

In the Planning phase, it was hard to decide and come up with functions but we narrowed our project scope to focus on inventory management and Purchase Order creation. On Function vs Entity matrix, Accounting is listed to show how invoice and purchase order management is related but it is not serving as one of a function in this project. After having Function vs Entity matrix, we requested user views of Purchase Order, Invoice, supplier information, product details, status and approvals for Purchase Order. While creating ERD for the documents, we were able to understand more about the business rules and process to apply onto the project.

During the Logical Design phase, it was important for us to come up with necessary tables to enhance function of the database. By having minimum tables and cardinalities, we prevented redundancy in information and made easier access when create, update and delete the data. Additionally, we put look up tables for supplier, receiving, invoicing and matching to users to have more effective access when they are processing the data.

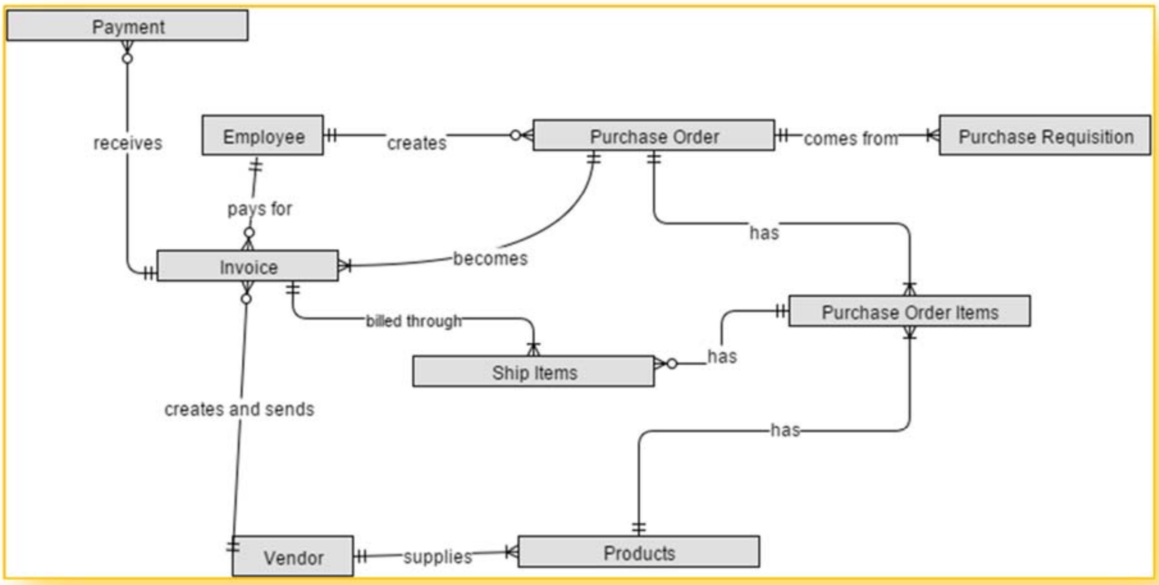
In the Physical Design step, the entire process was gathered to implement the design of the database. In terms of optimizing and having high integrity for the model, Process entity matrix, Transaction Analysis Forms, Composite Usage Map were incorporated. Furthermore, SQL code was written to have fast performance but sustaining high integrity at the same time. Additionally, putting indexing and clustering and triggers made the model even better to implement at the environment where it is needed.

B. Planning for the Database

Business Function-to-Data Entity Matrix

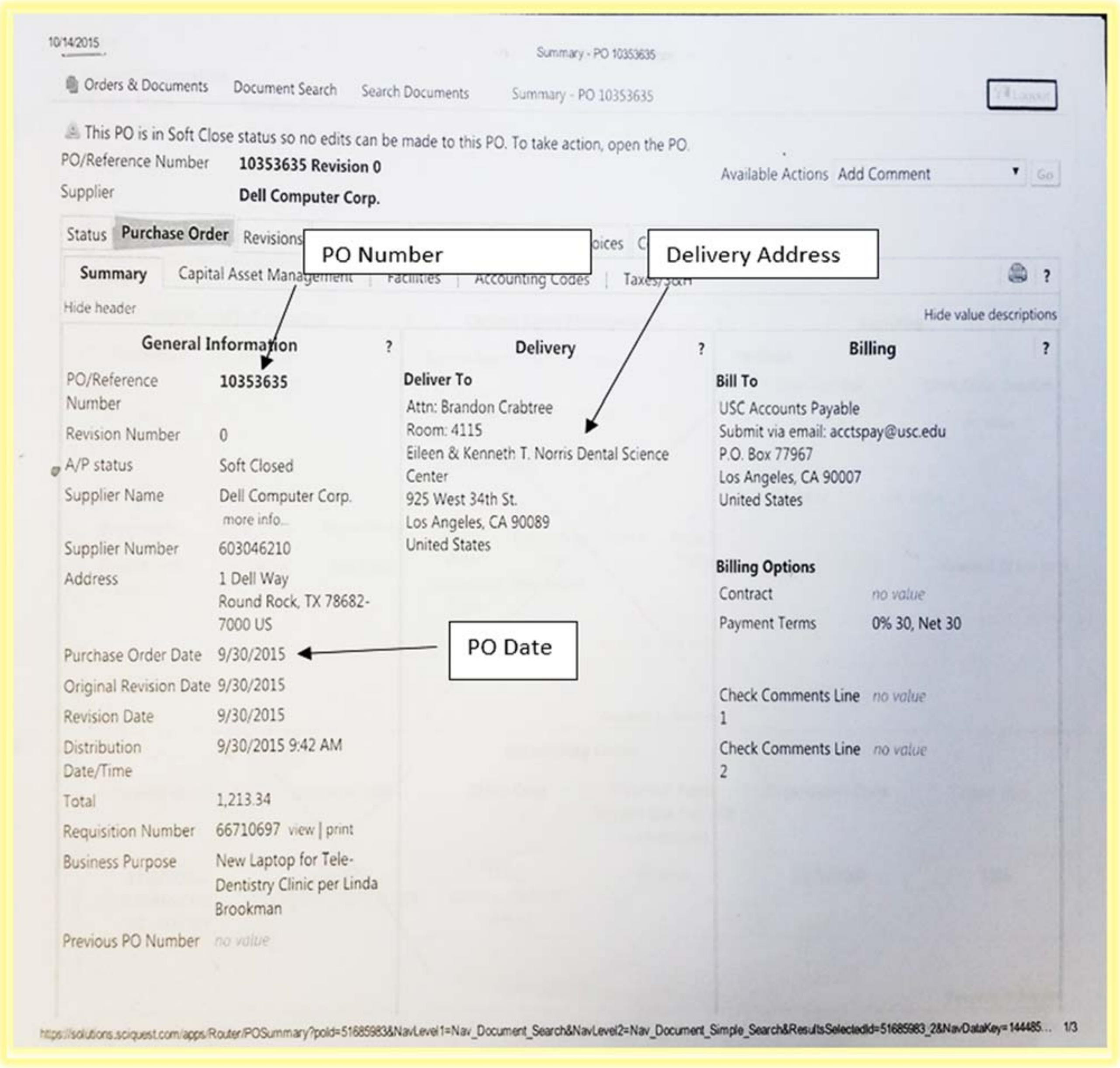
Function/ Entity	Purchase Order	Purchase Requistion	Accounts Payable Status	Employee	Recipient	Purchaser	Department	Account	Purchase Order Items	Payable Status	Vendor Product	Product	Vendor	Commodity Code	Product Size	Size	Vendor Employee	Ship Items	Invoice	Payment	Payment Type	Pay Status	Tax Type	Discount	Invoice Type
Invoice Management	X			X		X	**X**	X	X	X	X	X	X	**X**	X	X		X	X	X	**X**	X	X	X	X
Purchase Order Creation	X	X	X	X	X	X	**X**	X	X		X	X	X	**X**	X	X	X	X					X		
Accounting			X	X			**X**	X			X	X	X						X	X	**X**				

Preliminary Enterprise Data Model



C. Conceptual Design

Userview #1: Purchase Order



10/14/2015

Summary - PO 10353635

Orders & Documents   Document Search   Search Documents   Summary - PO 10353635

This PO is in Soft Close status so no edits can be made to this PO. To take action, open the PO.

PO/Reference Number **10353635 Revision 0**

Supplier **Dell Computer Corp.**

Available Actions Add Comment ▾ Go

PO Number

Delivery Address

PO Date

General Information

PO/Reference Number	**10353635**
Revision Number	0
A/P status	Soft Closed
Supplier Name	Dell Computer Corp. more info...
Supplier Number	603046210
Address	1 Dell Way Round Rock, TX 78682-7000 US
Purchase Order Date	9/30/2015
Original Revision Date	9/30/2015
Revision Date	9/30/2015
Distribution Date/Time	9/30/2015 9:42 AM
Total	1,213.34
Requisition Number	66710697 view | print
Business Purpose	New Laptop for Tele-Dentistry Clinic per Linda Brookman
Previous PO Number	*no value*

Delivery

Deliver To

Attn: Brandon Crabtree
Room: 4115
Eileen & Kenneth T. Norris Dental Science Center
925 West 34th St.
Los Angeles, CA 90089
United States

Billing

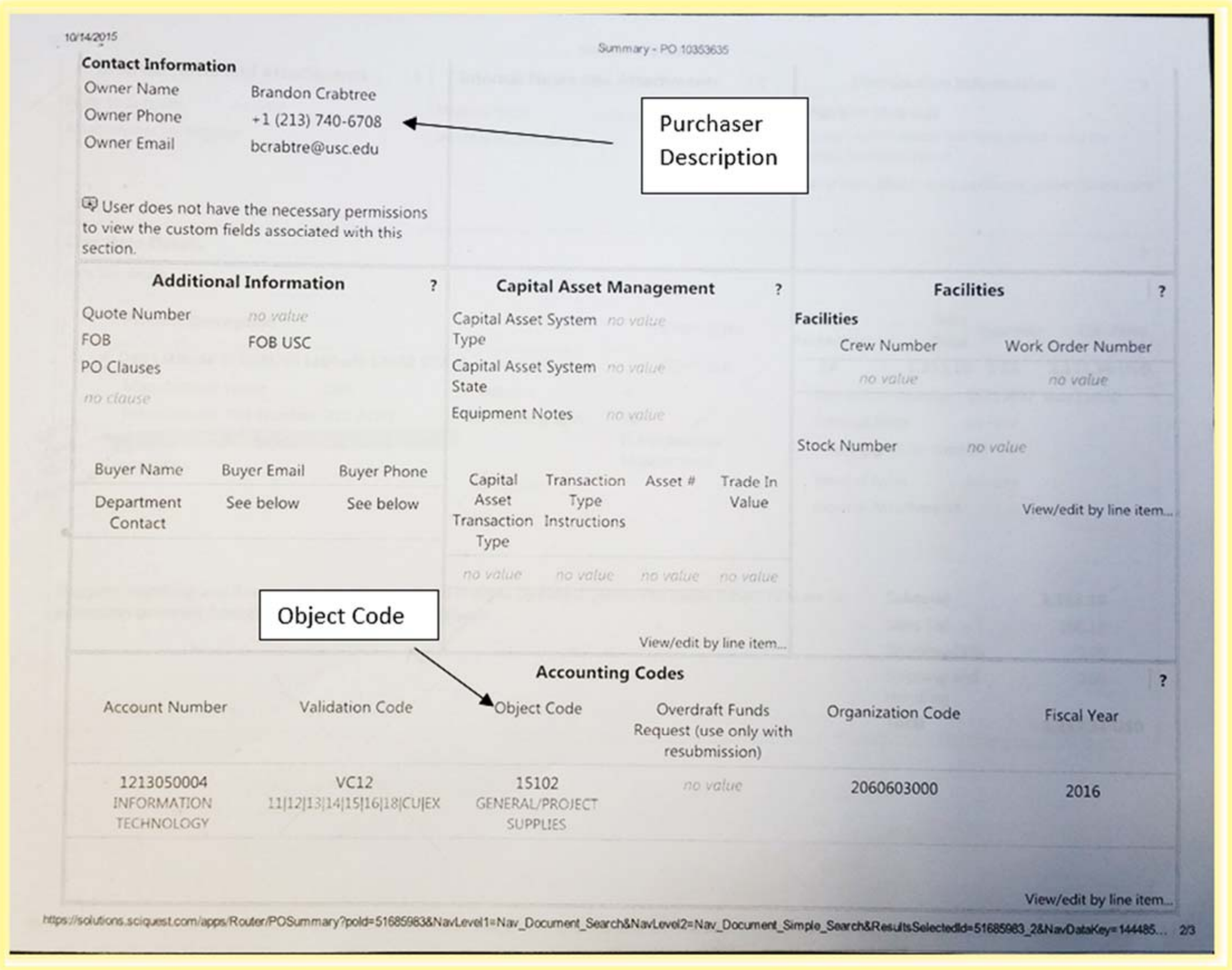
Bill To

USC Accounts Payable
Submit via email: acctspay@usc.edu
P.O. Box 77967
Los Angeles, CA 90007
United States

Billing Options

Contract	*no value*
Payment Terms	0% 30, Net 30
Check Comments Line 1	*no value*
Check Comments Line 2	*no value*

https://solutions.sciquest.com/apps/Router/POSummary?poId=51685983&NavLevel1=Nav_Document_Search&NavLevel2=Nav_Document_Simple_Search&ResultsSelectedId=51685983_2&NavDataKey=144485... 1/3



10/14/2015

Summary - PO 10353635

Contact Information

Owner Name	Brandon Crabtree
Owner Phone	+1 (213) 740-6708
Owner Email	bcrabtre@usc.edu

Purchaser Description

User does not have the necessary permissions to view the custom fields associated with this section.

Additional Information ?

Quote Number	no value
FOB	FOB USC
PO Clauses	

no clause

Buyer Name	Buyer Email	Buyer Phone
Department Contact	See below	See below

Capital Asset Management ?

Capital Asset System Type	no value
Capital Asset System State	no value
Equipment Notes	no value

Capital Asset Transaction Type	Transaction Type Instructions	Asset #	Trade In Value
no value	no value	no value	no value

View/edit by line item...

Facilities ?

Facilities

Crew Number	Work Order Number
no value	no value

Stock Number no value

View/edit by line item...

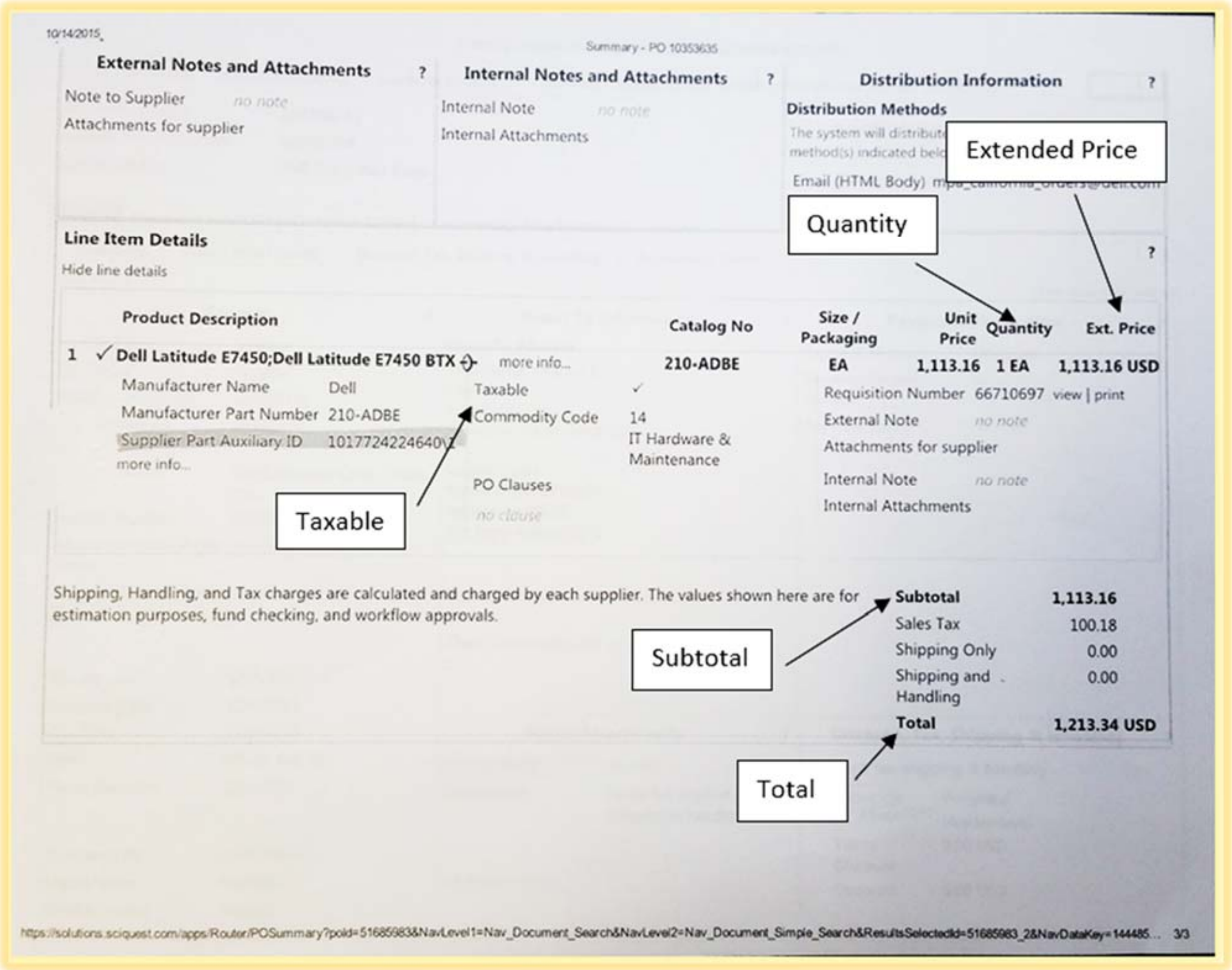
Object Code

Accounting Codes ?

Account Number	Validation Code	Object Code	Overdraft Funds Request (use only with resubmission)	Organization Code	Fiscal Year
1213050004 INFORMATION TECHNOLOGY	VC12 11|12|13|14|15|16|18|CU|EX	15102 GENERAL/PROJECT SUPPLIES	no value	2060603000	2016

View/edit by line item...

https://solutions.sciquest.com/apps/Router/POSummary?poId=51685983&NavLevel1=Nav_Document_Search&NavLevel2=Nav_Document_Simple_Search&ResultsSelectedId=51685983_2&NavDataKey=144485... 2/3



10/14/2015

Summary - PO 10353635

External Notes and Attachments ?

Note to Supplier *no note*

Attachments for supplier

Internal Notes and Attachments ?

Internal Note *no note*

Internal Attachments

Distribution Information ?

Distribution Methods

The system will distribut
method(s) indicated bel

Email (HTML Body) mpa_california_orders@dell.com

Line Item Details ?

Hide line details

		Product Description		Catalog No	Size / Packaging	Unit Price	Quantity	Ext. Price
1	✓	**Dell Latitude E7450;Dell Latitude E7450 BTX**	more info...	**210-ADBE**	**EA**	**1,113.16**	**1 EA**	**1,113.16 USD**

Manufacturer Name	Dell
Manufacturer Part Number	210-ADBE
Supplier Part Auxiliary ID	1017724224640\1

more info...

Taxable	✓
Commodity Code	14 IT Hardware & Maintenance
PO Clauses	

no clause

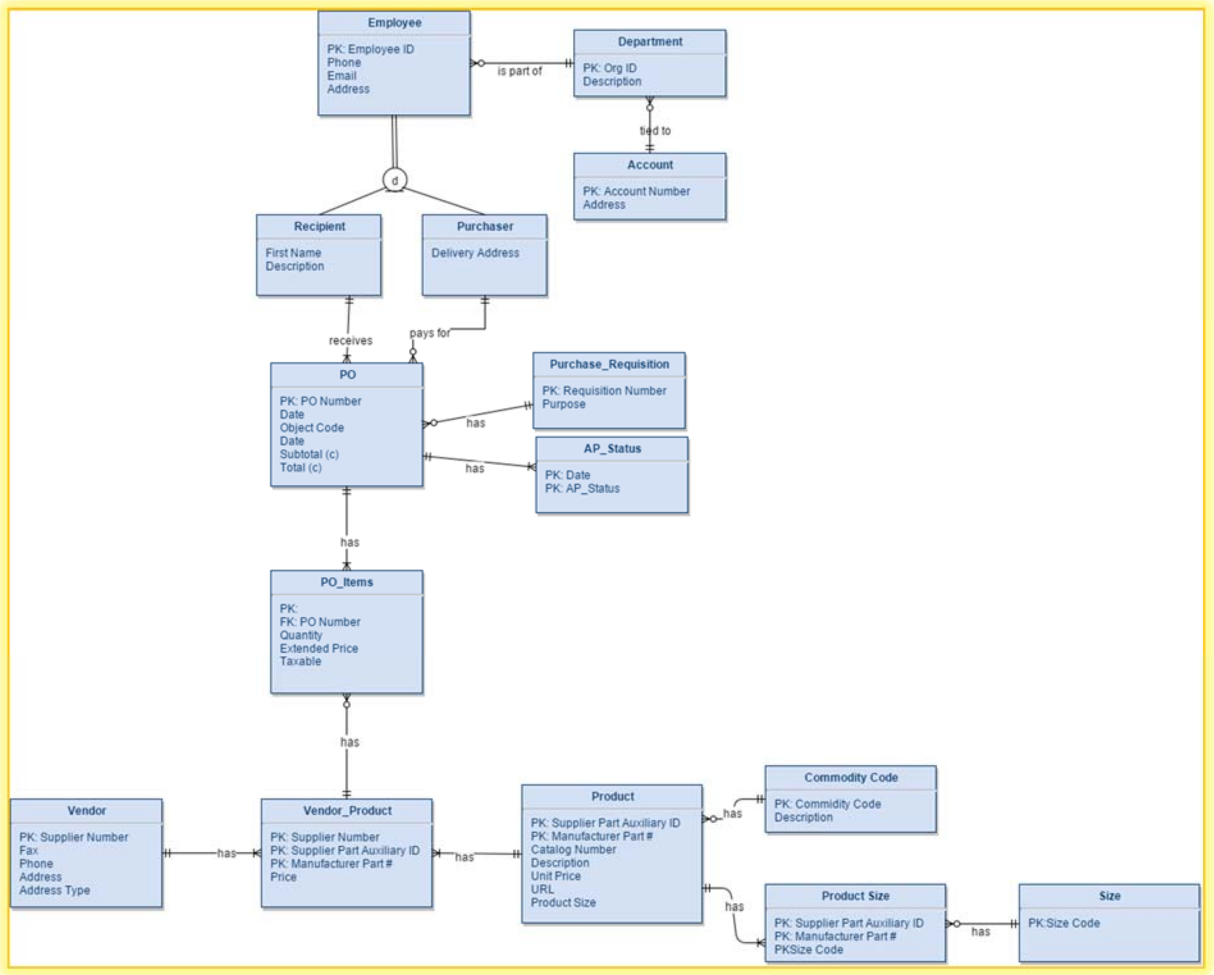
Requisition Number	66710697	view | print
External Note	*no note*	
Attachments for supplier		
Internal Note	*no note*	
Internal Attachments		

Shipping, Handling, and Tax charges are calculated and charged by each supplier. The values shown here are for estimation purposes, fund checking, and workflow approvals.

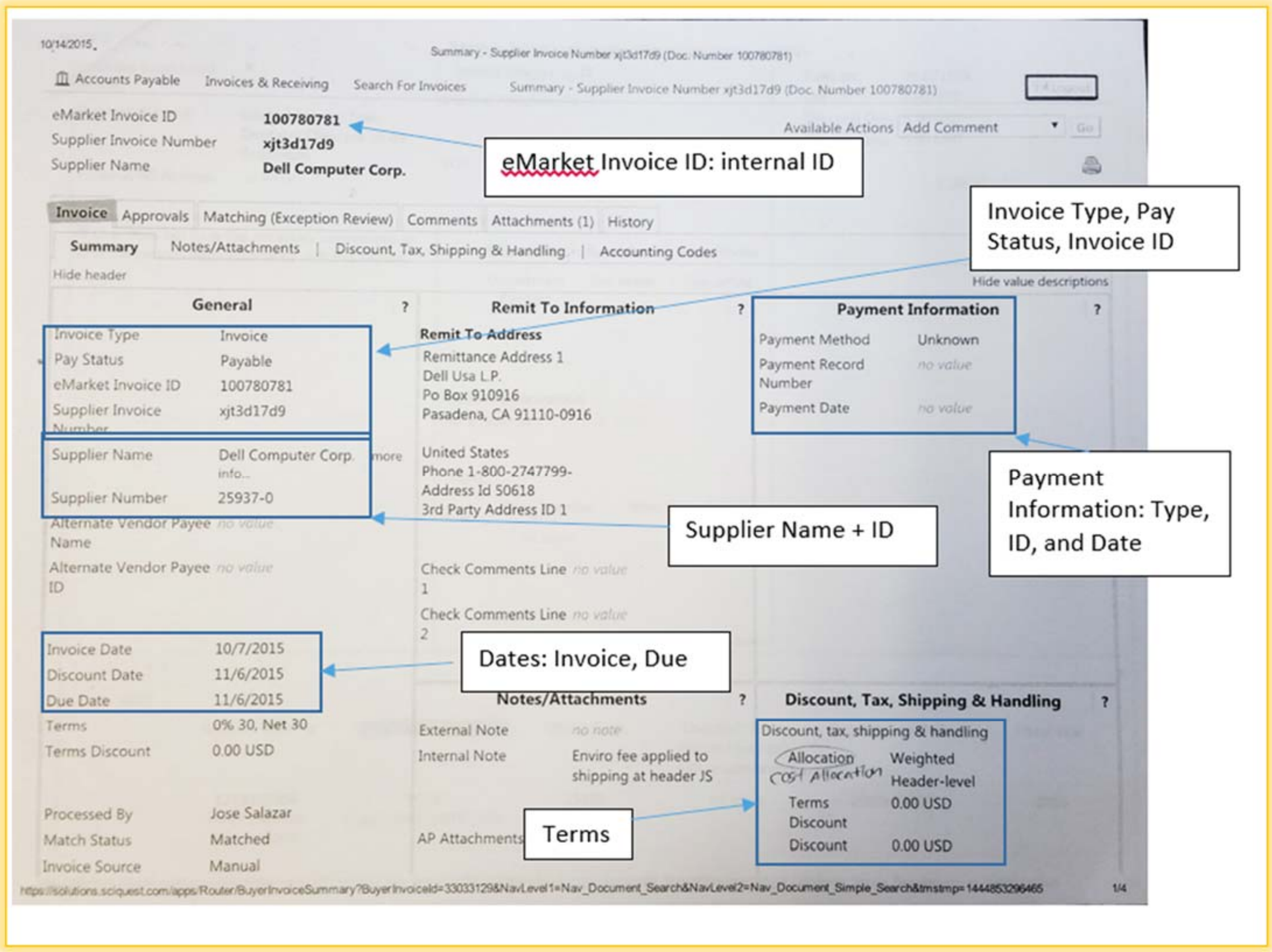
Subtotal	**1,113.16**
Sales Tax	100.18
Shipping Only	0.00
Shipping and Handling	0.00
Total	**1,213.34 USD**

https://solutions.sciquest.com/apps/Router/POSummary?poId=51685983&NavLevel1=Nav_Document_Search&NavLevel2=Nav_Document_Simple_Search&ResultsSelectedId=51685983_2&NavDataKey=144485... 3/3

Purchase Order ERD



Userview #2: Invoice



10/14/2015

Summary - Supplier Invoice Number xjt3d17d9 (Doc. Number 100780781)

Summary - Supplier Invoice Number xjt3d17d9 (Doc. Number 100780781)

eMarket Invoice ID	100780781
Supplier Invoice Number	xjt3d17d9
Supplier Name	Dell Computer Corp.

Available Actions: Add Comment

General

Invoice Type	Invoice
Pay Status	Payable
eMarket Invoice ID	100780781
Supplier Invoice Number	xjt3d17d9
Supplier Name	Dell Computer Corp. more info...
Supplier Number	25937-0
Alternate Vendor Payee Name	no value
Alternate Vendor Payee ID	no value
Invoice Date	10/7/2015
Discount Date	11/6/2015
Due Date	11/6/2015
Terms	0% 30, Net 30
Terms Discount	0.00 USD
Processed By	Jose Salazar
Match Status	Matched
Invoice Source	Manual

Remit To Information

Remit To Address
Remittance Address 1
Dell Usa L.P.
Po Box 910916
Pasadena, CA 91110-0916
United States
Phone 1-800-2747799-
Address Id 50618
3rd Party Address ID 1

Check Comments Line 1	no value
Check Comments Line 2	no value

Payment Information

Payment Method	Unknown
Payment Record Number	no value
Payment Date	no value

Notes/Attachments

External Note	no note
Internal Note	Enviro fee applied to shipping at header JS
AP Attachments	

Discount, Tax, Shipping & Handling

Discount, tax, shipping & handling

Allocation cost Allocation	Weighted Header-level
Terms Discount	0.00 USD
Discount	0.00 USD

https://solutions.sciquest.com/apps/Router/BuyerInvoiceSummary?BuyerInvoiceId=33033129&NavLevel1=Nav_Document_Search&NavLevel2=Nav_Document_Simple_Search&tmstmp=1444853296465 1/4

eMarket Invoice ID: internal ID

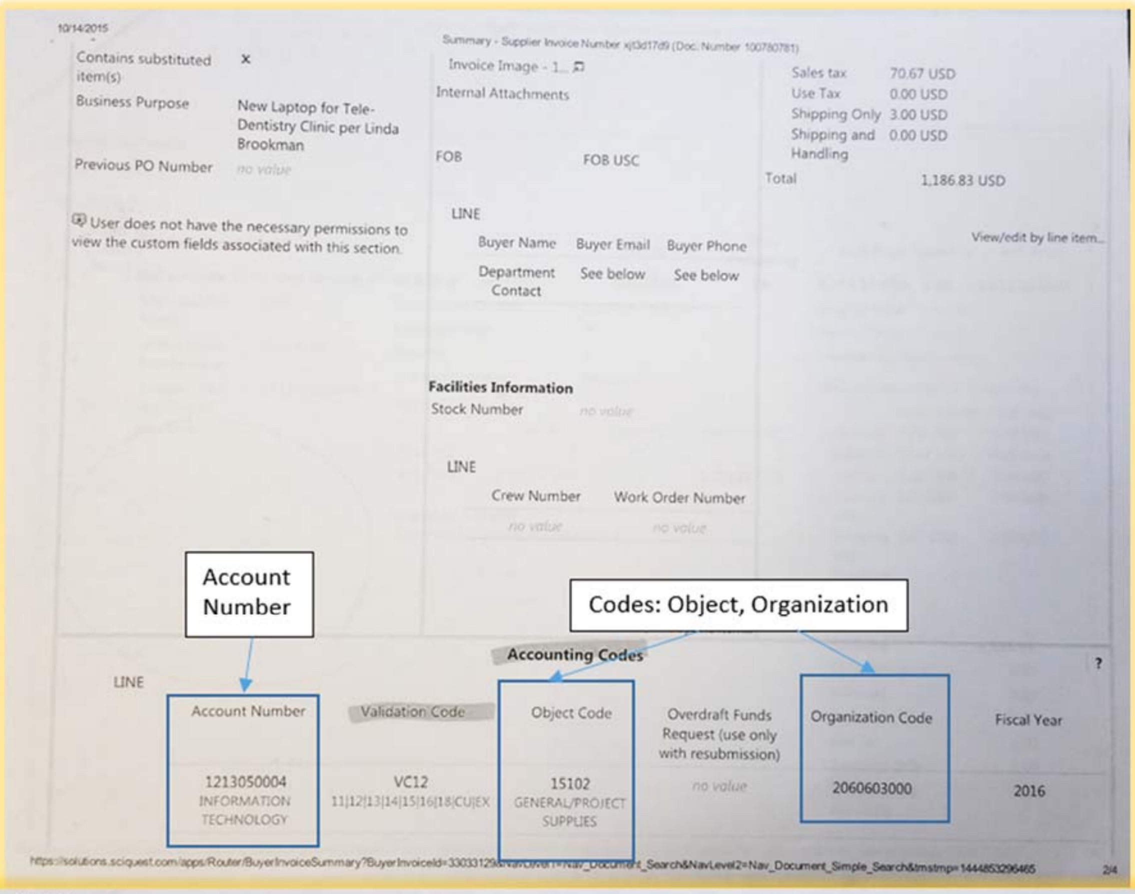
Invoice Type, Pay Status, Invoice ID

Payment Information: Type, ID, and Date

Supplier Name + ID

Dates: Invoice, Due

Terms



10/14/2015

Summary - Supplier Invoice Number xjt3d17d9 (Doc. Number 100780781)

Contains substituted item(s)	×
Business Purpose	New Laptop for Tele-Dentistry Clinic per Linda Brookman
Previous PO Number	no value

User does not have the necessary permissions to view the custom fields associated with this section.

Invoice Image - 1...

Internal Attachments

FOB FOB USC

LINE

Buyer Name	Buyer Email	Buyer Phone
Department Contact	See below	See below

Sales tax	70.67 USD
Use Tax	0.00 USD
Shipping Only	3.00 USD
Shipping and Handling	0.00 USD
Total	1,186.83 USD

View/edit by line item...

Facilities Information

Stock Number no value

LINE

Crew Number	Work Order Number
no value	no value

Account Number

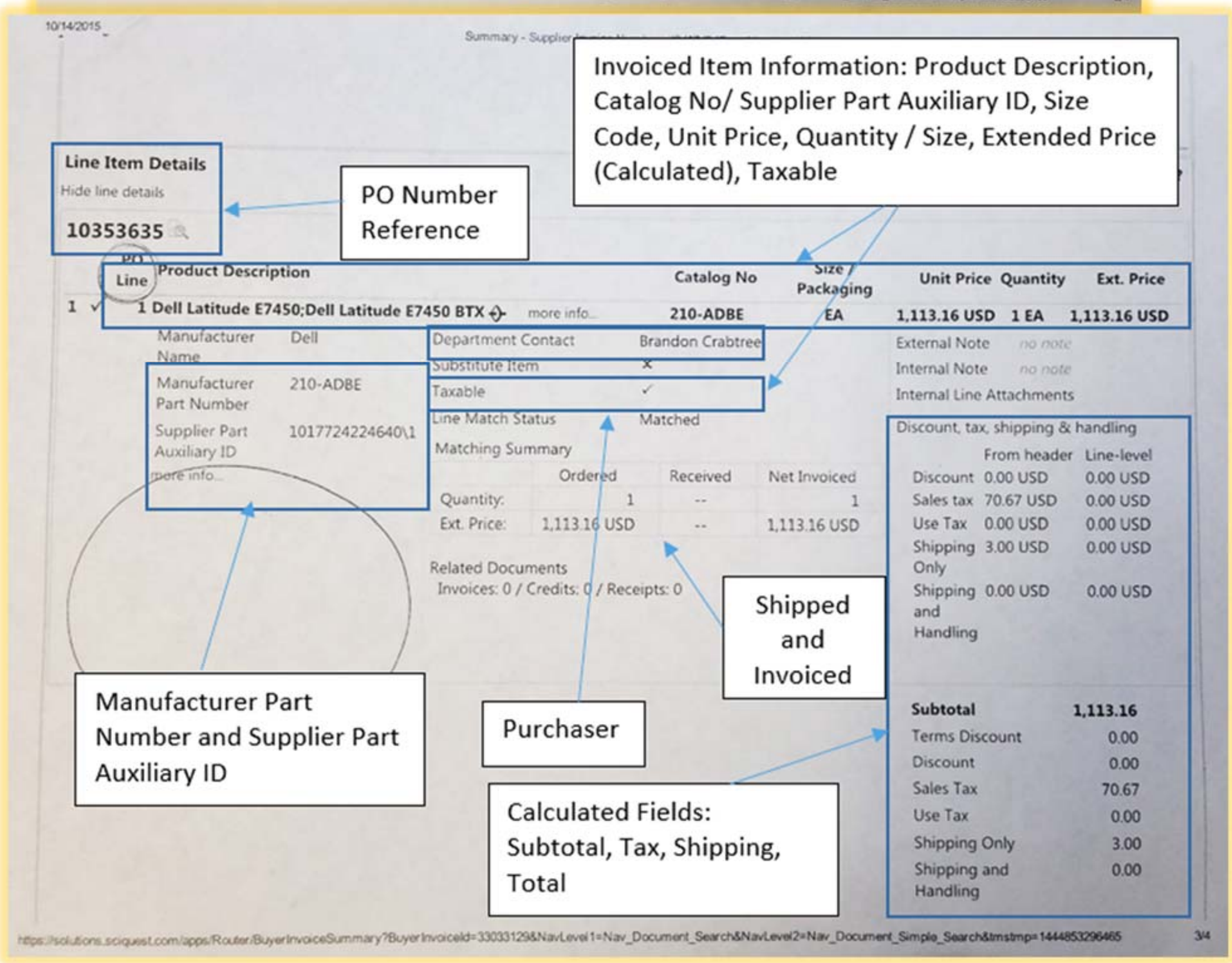
Codes: Object, Organization

Accounting Codes ?

LINE

Account Number	Validation Code	Object Code	Overdraft Funds Request (use only with resubmission)	Organization Code	Fiscal Year
1213050004 INFORMATION TECHNOLOGY	VC12 11|12|13|14|15|16|18|CU|EX	15102 GENERAL/PROJECT SUPPLIES	no value	2060603000	2016

https://solutions.sciquest.com/apps/Router/BuyerInvoiceSummary?BuyerInvoiceId=33033129&NavLevel1=Nav_Document_Search&NavLevel2=Nav_Document_Simple_Search&tmstmp=1444853296465 2/4



10/14/2015

Summary - Supplier

Invoiced Item Information: Product Description, Catalog No/ Supplier Part Auxiliary ID, Size Code, Unit Price, Quantity / Size, Extended Price (Calculated), Taxable

Line Item Details

Hide line details

10353635

PO Number Reference

	PO Line	Product Description		Catalog No	Size / Packaging	Unit Price	Quantity	Ext. Price
1 ✓	1	**Dell Latitude E7450;Dell Latitude E7450 BTX**	more info...	**210-ADBE**	**EA**	**1,113.16 USD**	**1 EA**	**1,113.16 USD**

Manufacturer Name	Dell
Manufacturer Part Number	210-ADBE
Supplier Part Auxiliary ID	1017724224640\1

more info...

Department Contact	Brandon Crabtree
Substitute Item	×
Taxable	✓
Line Match Status	Matched

Matching Summary

	Ordered	Received	Net Invoiced
Quantity:	1	--	1
Ext. Price:	1,113.16 USD	--	1,113.16 USD

Related Documents

Invoices: 0 / Credits: 0 / Receipts: 0

External Note	no note
Internal Note	no note
Internal Line Attachments	

Discount, tax, shipping & handling

	From header	Line-level
Discount	0.00 USD	0.00 USD
Sales tax	70.67 USD	0.00 USD
Use Tax	0.00 USD	0.00 USD
Shipping Only	3.00 USD	0.00 USD
Shipping and Handling	0.00 USD	0.00 USD

Subtotal	**1,113.16**
Terms Discount	0.00
Discount	0.00
Sales Tax	70.67
Use Tax	0.00
Shipping Only	3.00
Shipping and Handling	0.00

Shipped and Invoiced

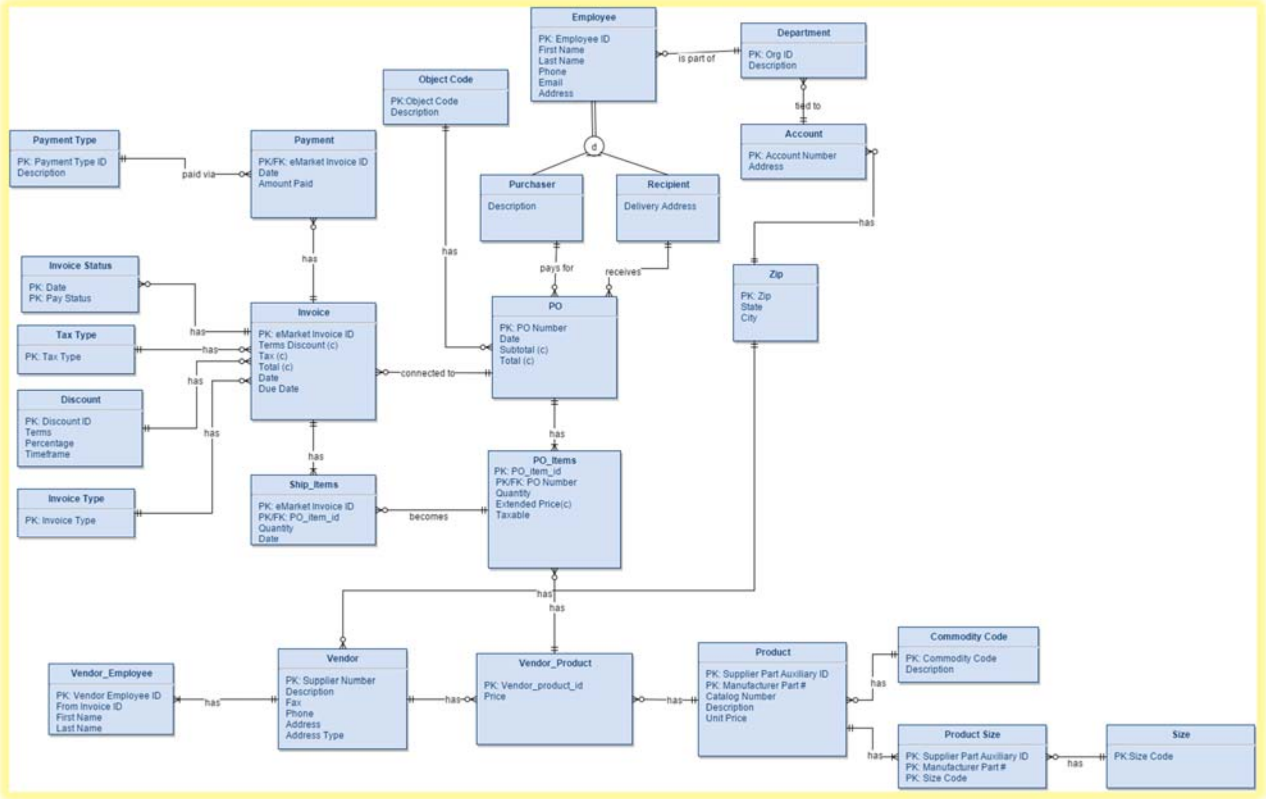
Manufacturer Part Number and Supplier Part Auxiliary ID

Purchaser

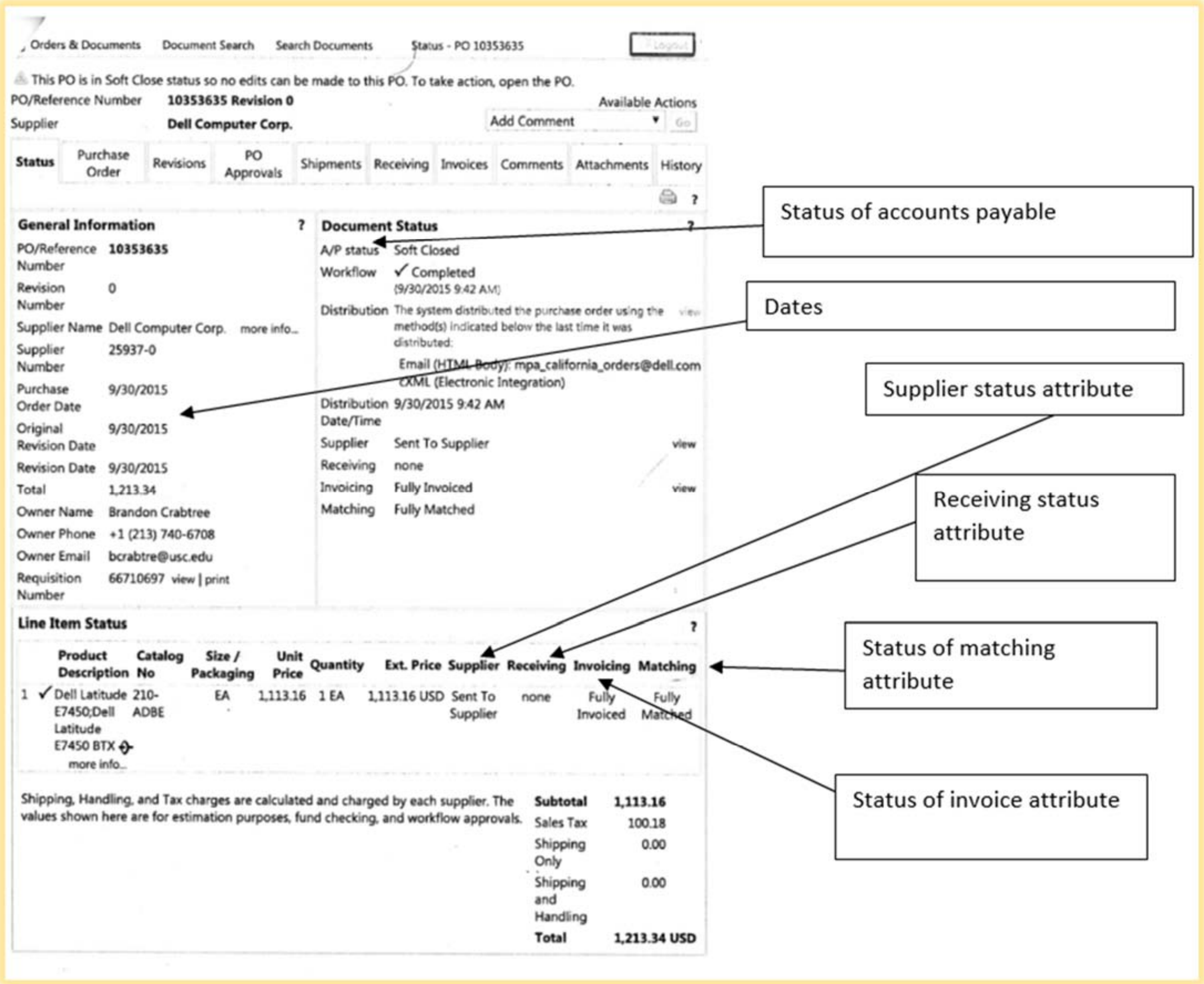
Calculated Fields: Subtotal, Tax, Shipping, Total

https://solutions.sciquest.com/apps/Router/BuyerInvoiceSummary?BuyerInvoiceId=33033129&NavLevel1=Nav_Document_Search&NavLevel2=Nav_Document_Simple_Search&tmstmp=1444853296465 3/4

Invoice ERD


Employee
PK: Employee ID
First Name
Last Name
Phone
Email
Address
is part of
Department
PK: Org ID
Description
tied to
Account
PK: Account Number
Address
Object Code
PK:Object Code
Description
d
Purchaser
Description
Recipient
Delivery Address
Payment Type
PK: Payment Type ID
Description
paid via
Payment
PK/FK: eMarket Invoice ID
Date
Amount Paid
has
has
pays for
receives
has
Zip
PK: Zip
State
City
Invoice Status
PK: Date
PK: Pay Status
Tax Type
PK: Tax Type
Discount
PK: Discount ID
Terms
Percentage
Timeframe
Invoice Type
PK: Invoice Type
has
has
has
has
Invoice
PK: eMarket Invoice ID
Terms Discount (c)
Tax (c)
Total (c)
Date
Due Date
connected to
PO
PK: PO Number
Date
Subtotal (c)
Total (c)
has
has
Ship_Items
PK: eMarket Invoice ID
PK/FK: PO_item_id
Quantity
Date
becomes
PO_Items
PK: PO_item_id
PK/FK: PO Number
Quantity
Extended Price(c)
Taxable
has
has
Vendor_Employee
PK: Vendor Employee ID
From Invoice ID
First Name
Last Name
has
Vendor
PK: Supplier Number
Description
Fax
Phone
Address
Address Type
has
Vendor_Product
PK: Vendor_product_id
Price
has
Product
PK: Supplier Part Auxiliary ID
PK: Manufacturer Part #
Catalog Number
Description
Unit Price
has
Commodity Code
PK: Commodity Code
Description
has
Product Size
PK: Supplier Part Auxiliary ID
PK: Manufacturer Part #
PK: Size Code
has
Size
PK:Size Code

Userview #3: PO Status



This PO is in Soft Close status so no edits can be made to this PO. To take action, open the PO.

PO/Reference Number	**10353635 Revision 0**
Supplier	**Dell Computer Corp.**

General Information

PO/Reference Number	**10353635**
Revision Number	0
Supplier Name	Dell Computer Corp. more info...
Supplier Number	25937-0
Purchase Order Date	9/30/2015
Original Revision Date	9/30/2015
Revision Date	9/30/2015
Total	1,213.34
Owner Name	Brandon Crabtree
Owner Phone	+1 (213) 740-6708
Owner Email	bcrabtre@usc.edu
Requisition Number	66710697 view | print

Document Status

A/P status	Soft Closed	
Workflow	✓ Completed (9/30/2015 9:42 AM)	
Distribution	The system distributed the purchase order using the method(s) indicated below the last time it was distributed: Email (HTML Body): mpa_california_orders@dell.com cXML (Electronic Integration)	view
Distribution Date/Time	9/30/2015 9:42 AM	
Supplier	Sent To Supplier	view
Receiving	none	
Invoicing	Fully Invoiced	view
Matching	Fully Matched	

Line Item Status

		Product Description	Catalog No	Size / Packaging	Unit Price	Quantity	Ext. Price	Supplier	Receiving	Invoicing	Matching
1	✓	Dell Latitude E7450;Dell Latitude E7450 BTX more info...	210-ADBE	EA	1,113.16	1 EA	1,113.16 USD	Sent To Supplier	none	Fully Invoiced	Fully Matched

Shipping, Handling, and Tax charges are calculated and charged by each supplier. The values shown here are for estimation purposes, fund checking, and workflow approvals.

Subtotal	**1,113.16**
Sales Tax	100.18
Shipping Only	0.00
Shipping and Handling	0.00
Total	**1,213.34 USD**

Status of accounts payable

Dates

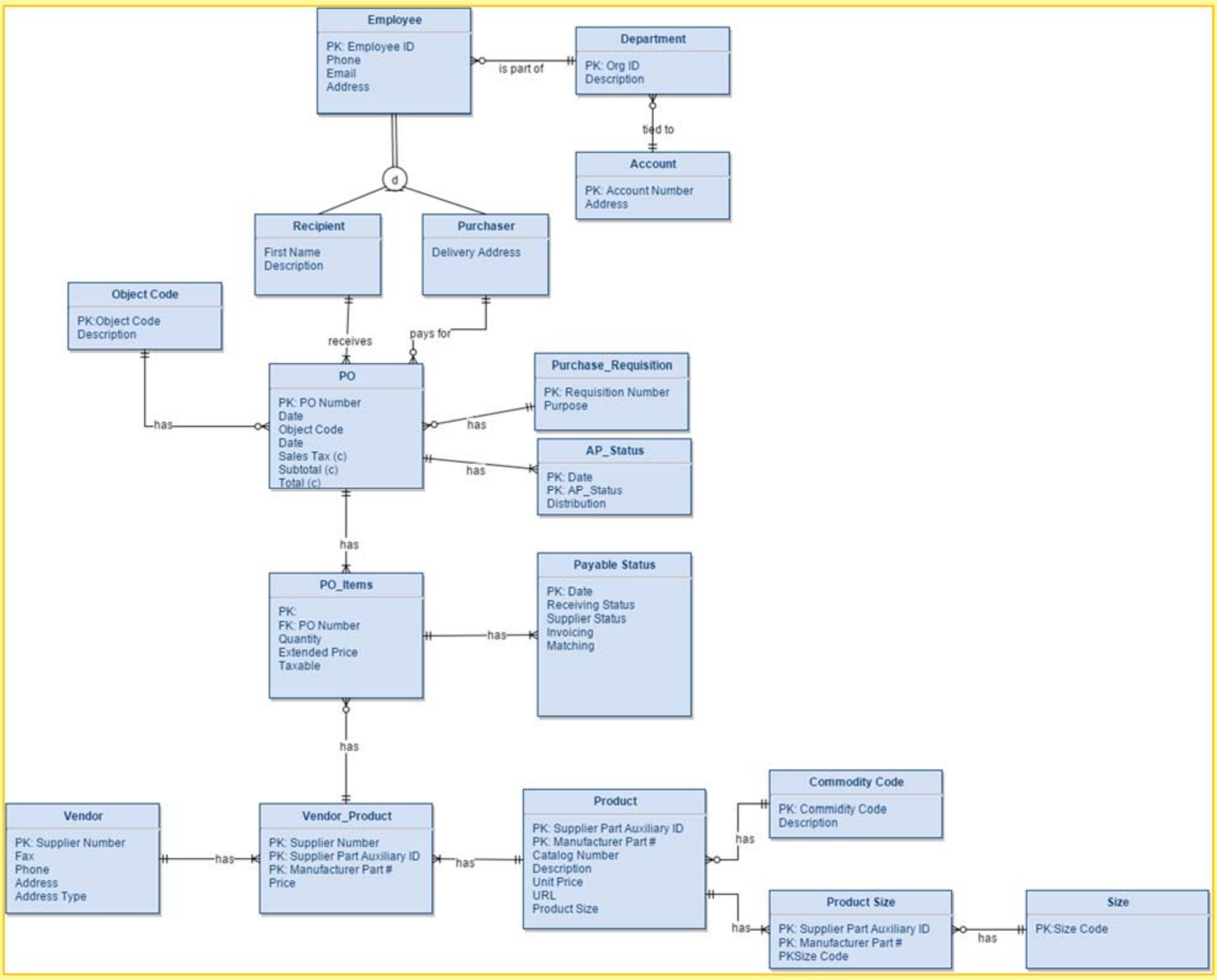
Supplier status attribute

Receiving status attribute

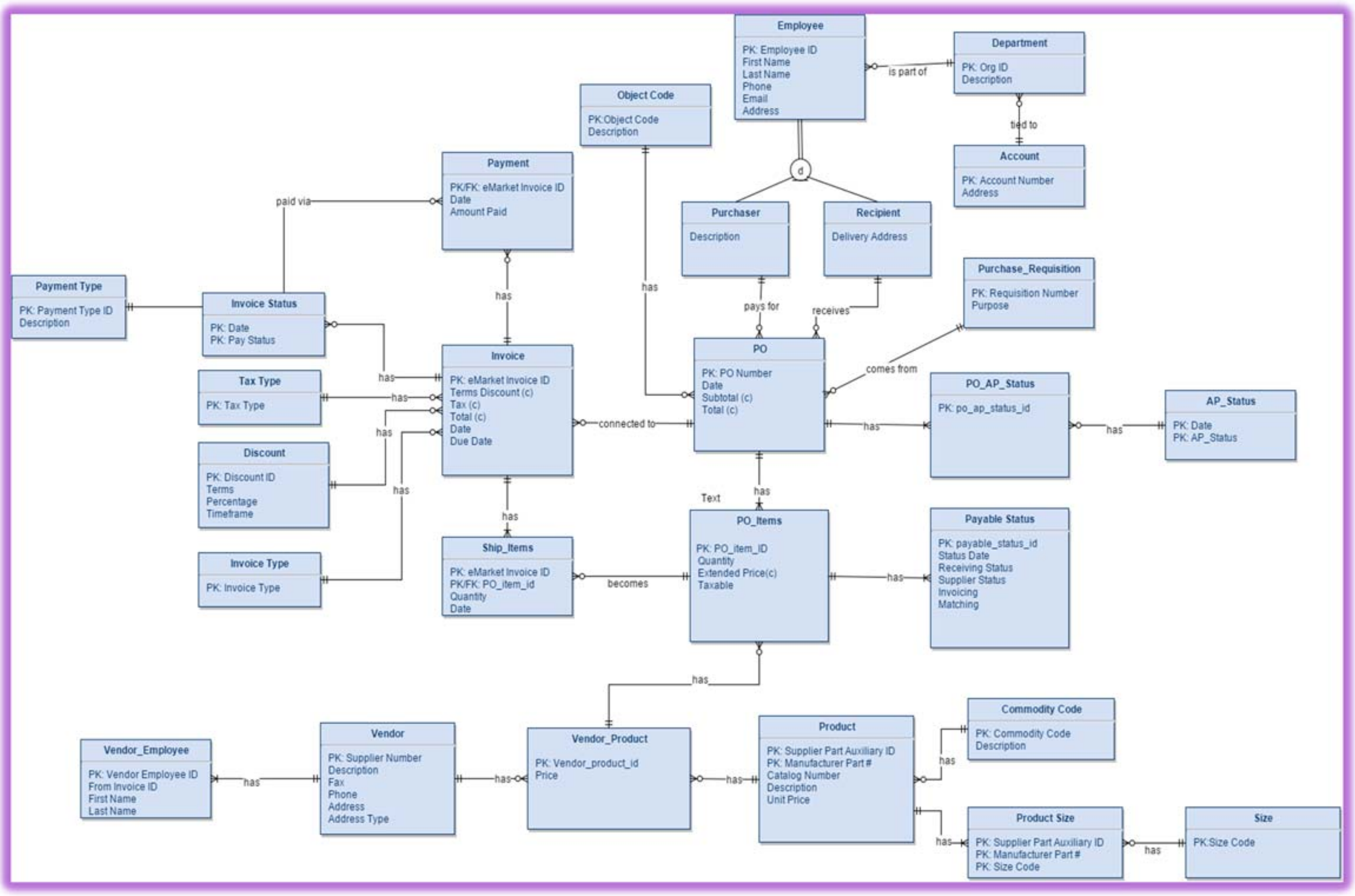
Status of matching attribute

Status of invoice attribute

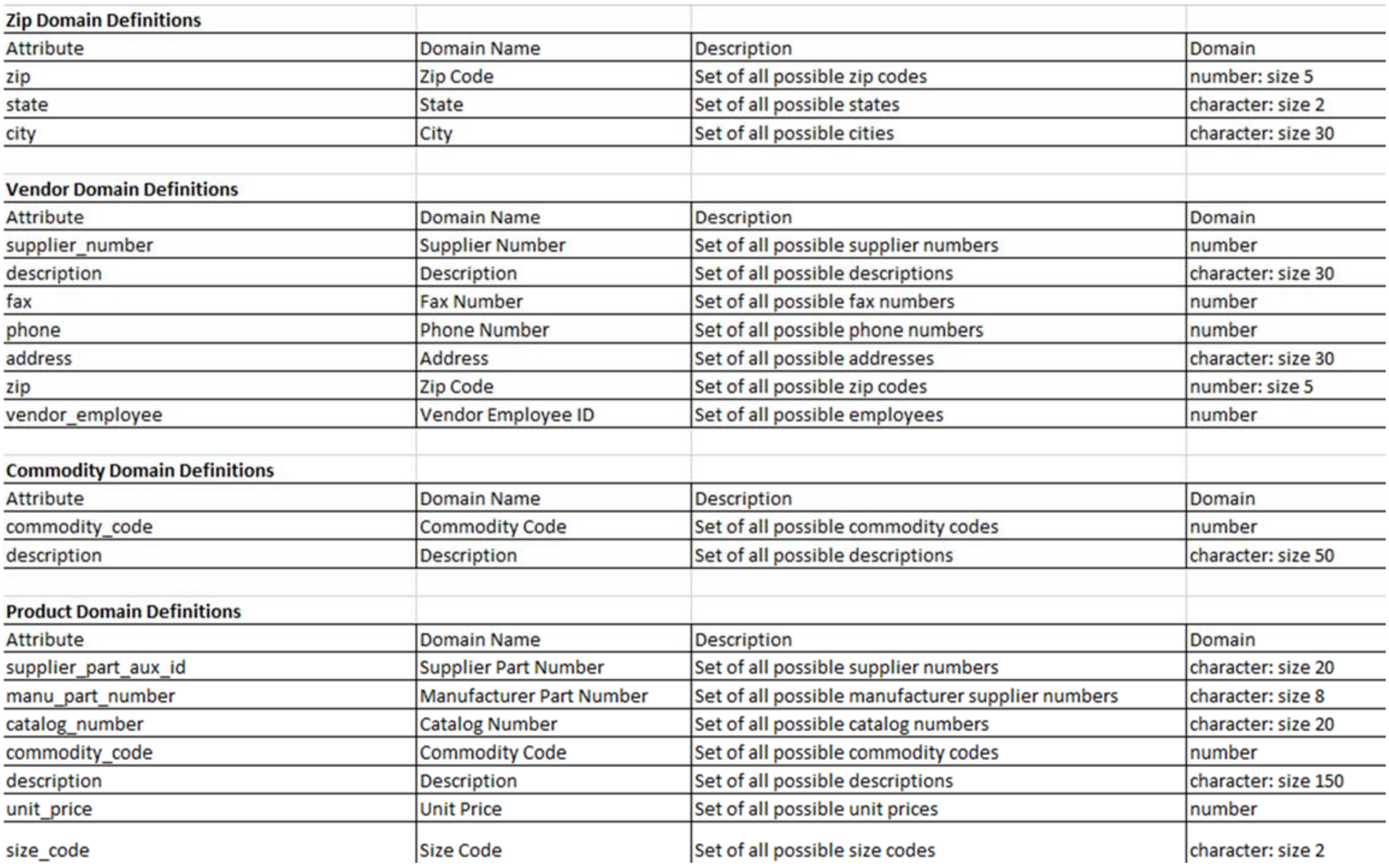
PO Status ERD



Conceptual Data model (CDM)



Domains & Constraints



Zip Domain Definitions

Attribute	Domain Name	Description	Domain
zip	Zip Code	Set of all possible zip codes	number: size 5
state	State	Set of all possible states	character: size 2
city	City	Set of all possible cities	character: size 30

Vendor Domain Definitions

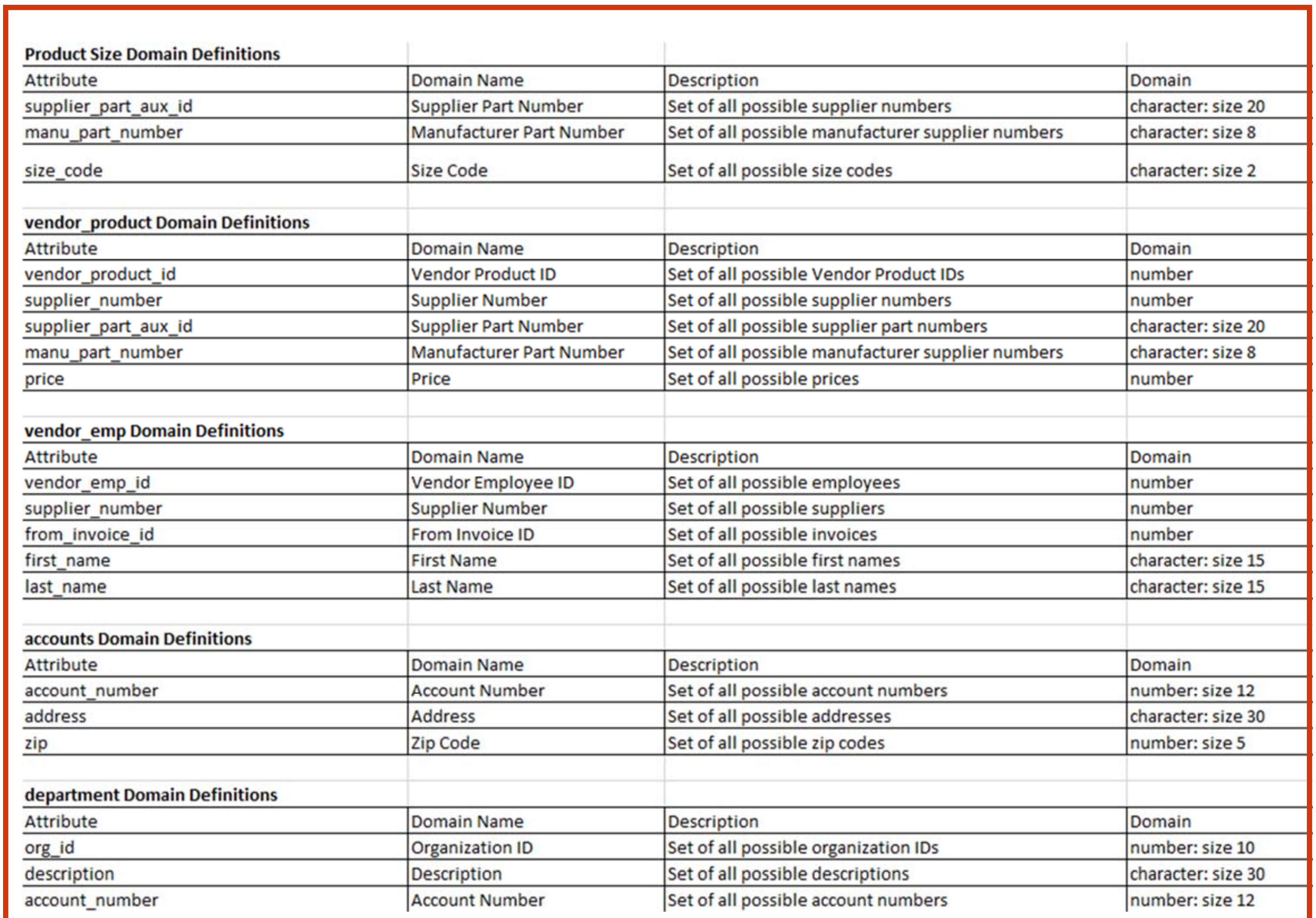
Attribute	Domain Name	Description	Domain
supplier_number	Supplier Number	Set of all possible supplier numbers	number
description	Description	Set of all possible descriptions	character: size 30
fax	Fax Number	Set of all possible fax numbers	number
phone	Phone Number	Set of all possible phone numbers	number
address	Address	Set of all possible addresses	character: size 30
zip	Zip Code	Set of all possible zip codes	number: size 5
vendor_employee	Vendor Employee ID	Set of all possible employees	number

Commodity Domain Definitions

Attribute	Domain Name	Description	Domain
commodity_code	Commodity Code	Set of all possible commodity codes	number
description	Description	Set of all possible descriptions	character: size 50

Product Domain Definitions

Attribute	Domain Name	Description	Domain
supplier_part_aux_id	Supplier Part Number	Set of all possible supplier numbers	character: size 20
manu_part_number	Manufacturer Part Number	Set of all possible manufacturer supplier numbers	character: size 8
catalog_number	Catalog Number	Set of all possible catalog numbers	character: size 20
commodity_code	Commodity Code	Set of all possible commodity codes	number
description	Description	Set of all possible descriptions	character: size 150
unit_price	Unit Price	Set of all possible unit prices	number
size_code	Size Code	Set of all possible size codes	character: size 2



Product Size Domain Definitions

Attribute	Domain Name	Description	Domain
supplier_part_aux_id	Supplier Part Number	Set of all possible supplier numbers	character: size 20
manu_part_number	Manufacturer Part Number	Set of all possible manufacturer supplier numbers	character: size 8
size_code	Size Code	Set of all possible size codes	character: size 2

vendor_product Domain Definitions

Attribute	Domain Name	Description	Domain
vendor_product_id	Vendor Product ID	Set of all possible Vendor Product IDs	number
supplier_number	Supplier Number	Set of all possible supplier numbers	number
supplier_part_aux_id	Supplier Part Number	Set of all possible supplier part numbers	character: size 20
manu_part_number	Manufacturer Part Number	Set of all possible manufacturer supplier numbers	character: size 8
price	Price	Set of all possible prices	number

vendor_emp Domain Definitions

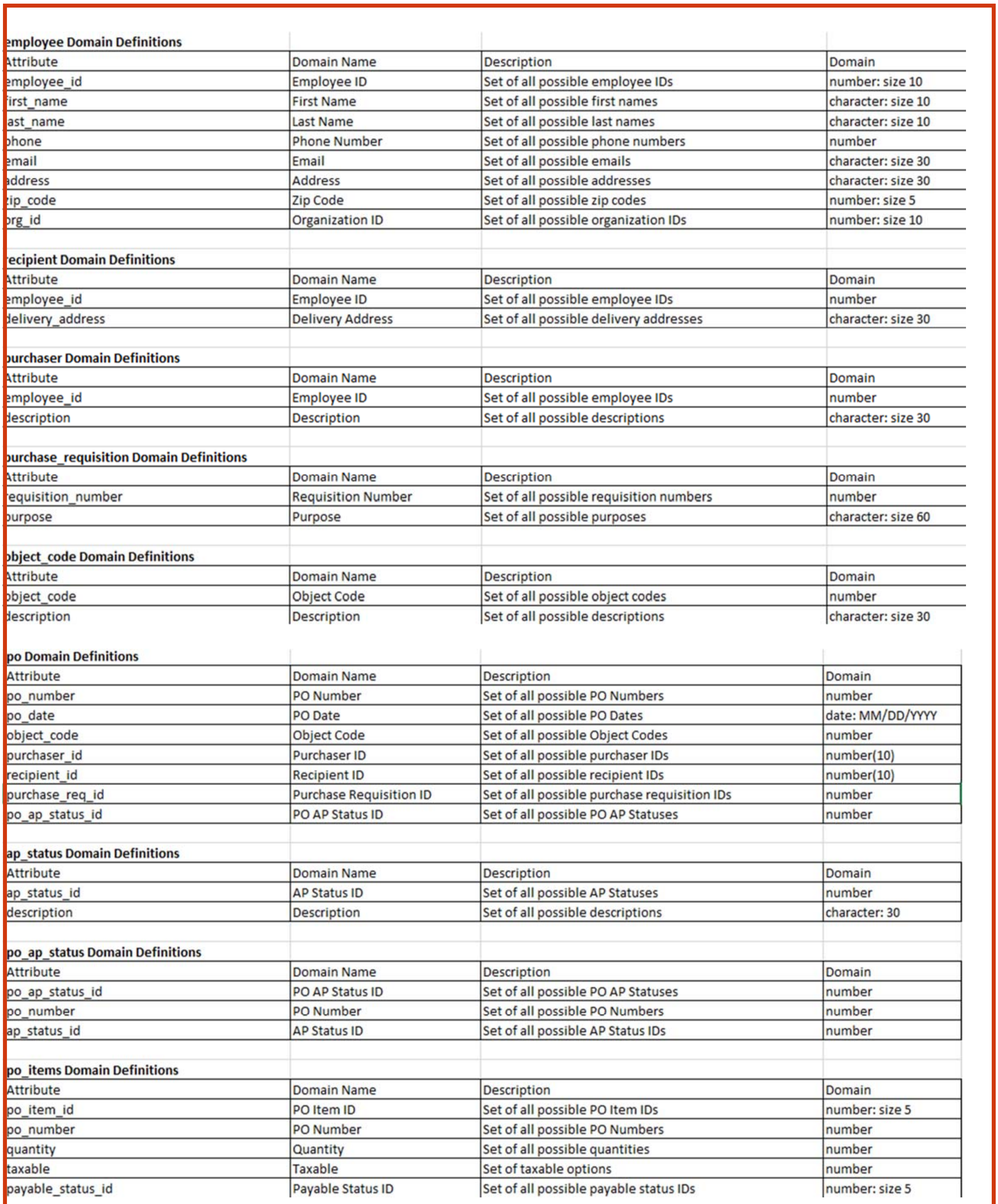
Attribute	Domain Name	Description	Domain
vendor_emp_id	Vendor Employee ID	Set of all possible employees	number
supplier_number	Supplier Number	Set of all possible suppliers	number
from_invoice_id	From Invoice ID	Set of all possible invoices	number
first_name	First Name	Set of all possible first names	character: size 15
last_name	Last Name	Set of all possible last names	character: size 15

accounts Domain Definitions

Attribute	Domain Name	Description	Domain
account_number	Account Number	Set of all possible account numbers	number: size 12
address	Address	Set of all possible addresses	character: size 30
zip	Zip Code	Set of all possible zip codes	number: size 5

department Domain Definitions

Attribute	Domain Name	Description	Domain
org_id	Organization ID	Set of all possible organization IDs	number: size 10
description	Description	Set of all possible descriptions	character: size 30
account_number	Account Number	Set of all possible account numbers	number: size 12



employee Domain Definitions

Attribute	Domain Name	Description	Domain
employee_id	Employee ID	Set of all possible employee IDs	number: size 10
first_name	First Name	Set of all possible first names	character: size 10
ast_name	Last Name	Set of all possible last names	character: size 10
phone	Phone Number	Set of all possible phone numbers	number
email	Email	Set of all possible emails	character: size 30
ddress	Address	Set of all possible addresses	character: size 30
zip_code	Zip Code	Set of all possible zip codes	number: size 5
org_id	Organization ID	Set of all possible organization IDs	number: size 10

ecipient Domain Definitions

Attribute	Domain Name	Description	Domain
employee_id	Employee ID	Set of all possible employee IDs	number
delivery_address	Delivery Address	Set of all possible delivery addresses	character: size 30

purchaser Domain Definitions

Attribute	Domain Name	Description	Domain
employee_id	Employee ID	Set of all possible employee IDs	number
description	Description	Set of all possible descriptions	character: size 30

purchase_requisition Domain Definitions

Attribute	Domain Name	Description	Domain
equisition_number	Requisition Number	Set of all possible requisition numbers	number
purpose	Purpose	Set of all possible purposes	character: size 60

object_code Domain Definitions

Attribute	Domain Name	Description	Domain
object_code	Object Code	Set of all possible object codes	number
description	Description	Set of all possible descriptions	character: size 30

po Domain Definitions

Attribute	Domain Name	Description	Domain
po_number	PO Number	Set of all possible PO Numbers	number
po_date	PO Date	Set of all possible PO Dates	date: MM/DD/YYYY
object_code	Object Code	Set of all possible Object Codes	number
purchaser_id	Purchaser ID	Set of all possible purchaser IDs	number(10)
recipient_id	Recipient ID	Set of all possible recipient IDs	number(10)
purchase_req_id	Purchase Requisition ID	Set of all possible purchase requisition IDs	number
po_ap_status_id	PO AP Status ID	Set of all possible PO AP Statuses	number

ap_status Domain Definitions

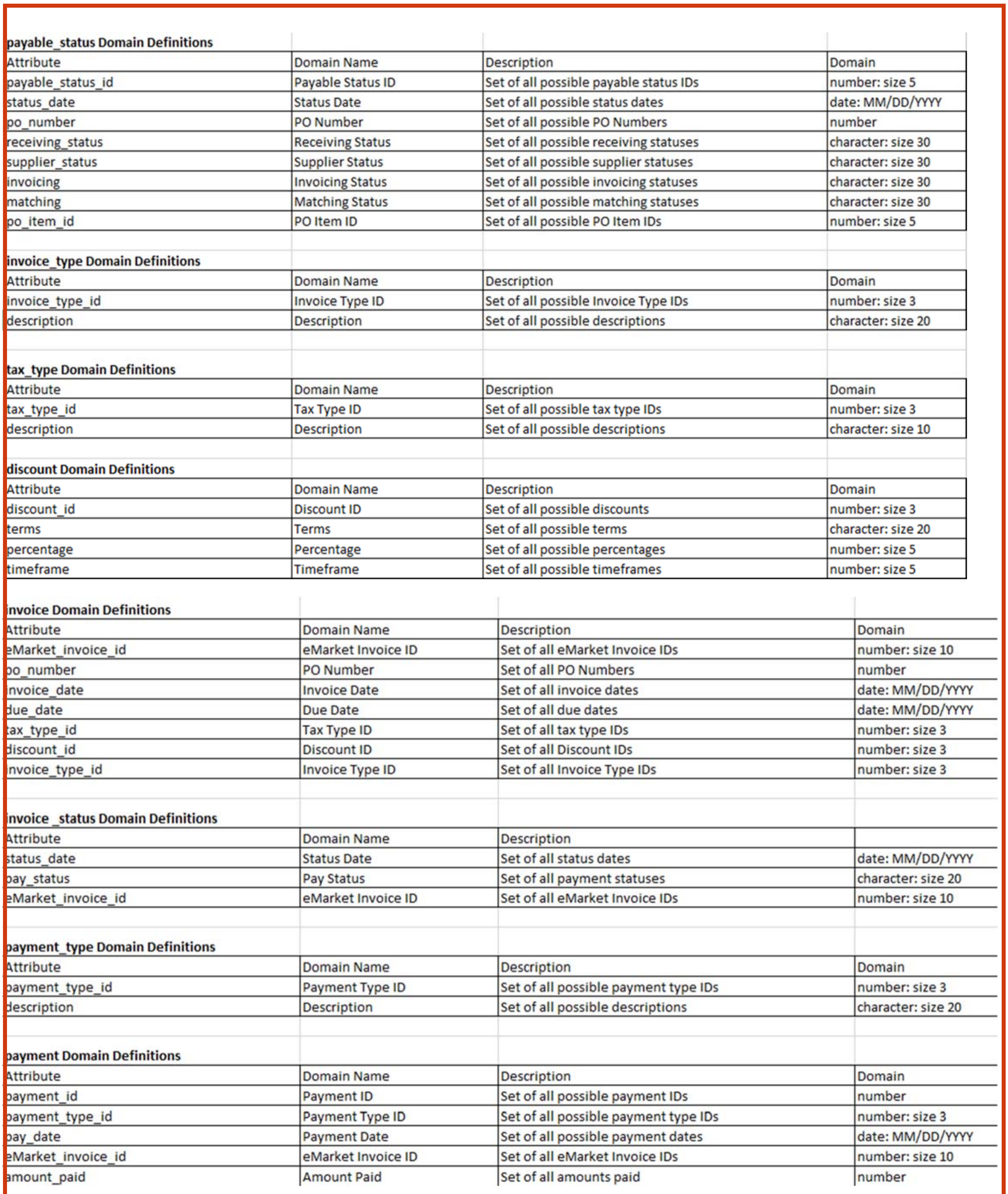
Attribute	Domain Name	Description	Domain
ap_status_id	AP Status ID	Set of all possible AP Statuses	number
description	Description	Set of all possible descriptions	character: 30

po_ap_status Domain Definitions

Attribute	Domain Name	Description	Domain
po_ap_status_id	PO AP Status ID	Set of all possible PO AP Statuses	number
po_number	PO Number	Set of all possible PO Numbers	number
ap_status_id	AP Status ID	Set of all possible AP Status IDs	number

po_items Domain Definitions

Attribute	Domain Name	Description	Domain
po_item_id	PO Item ID	Set of all possible PO Item IDs	number: size 5
po_number	PO Number	Set of all possible PO Numbers	number
quantity	Quantity	Set of all possible quantities	number
taxable	Taxable	Set of taxable options	number
payable_status_id	Payable Status ID	Set of all possible payable status IDs	number: size 5



payable_status Domain Definitions

Attribute	Domain Name	Description	Domain
payable_status_id	Payable Status ID	Set of all possible payable status IDs	number: size 5
status_date	Status Date	Set of all possible status dates	date: MM/DD/YYYY
po_number	PO Number	Set of all possible PO Numbers	number
receiving_status	Receiving Status	Set of all possible receiving statuses	character: size 30
supplier_status	Supplier Status	Set of all possible supplier statuses	character: size 30
invoicing	Invoicing Status	Set of all possible invoicing statuses	character: size 30
matching	Matching Status	Set of all possible matching statuses	character: size 30
po_item_id	PO Item ID	Set of all possible PO Item IDs	number: size 5

invoice_type Domain Definitions

Attribute	Domain Name	Description	Domain
invoice_type_id	Invoice Type ID	Set of all possible Invoice Type IDs	number: size 3
description	Description	Set of all possible descriptions	character: size 20

tax_type Domain Definitions

Attribute	Domain Name	Description	Domain
tax_type_id	Tax Type ID	Set of all possible tax type IDs	number: size 3
description	Description	Set of all possible descriptions	character: size 10

discount Domain Definitions

Attribute	Domain Name	Description	Domain
discount_id	Discount ID	Set of all possible discounts	number: size 3
terms	Terms	Set of all possible terms	character: size 20
percentage	Percentage	Set of all possible percentages	number: size 5
timeframe	Timeframe	Set of all possible timeframes	number: size 5

invoice Domain Definitions

Attribute	Domain Name	Description	Domain
eMarket_invoice_id	eMarket Invoice ID	Set of all eMarket Invoice IDs	number: size 10
po_number	PO Number	Set of all PO Numbers	number
invoice_date	Invoice Date	Set of all invoice dates	date: MM/DD/YYYY
due_date	Due Date	Set of all due dates	date: MM/DD/YYYY
tax_type_id	Tax Type ID	Set of all tax type IDs	number: size 3
discount_id	Discount ID	Set of all Discount IDs	number: size 3
invoice_type_id	Invoice Type ID	Set of all Invoice Type IDs	number: size 3

invoice _status Domain Definitions

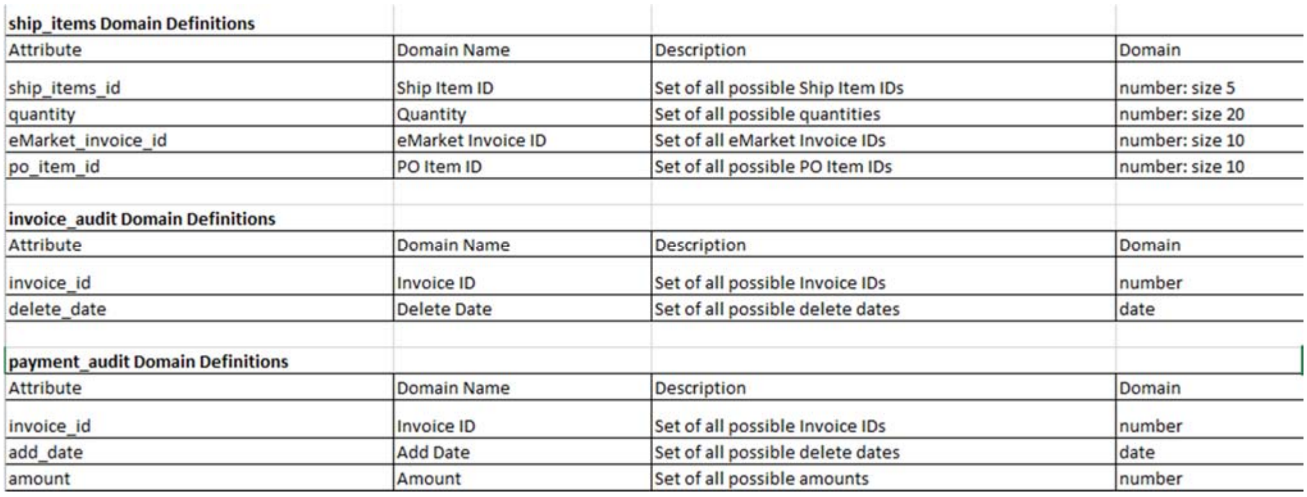
Attribute	Domain Name	Description	
status_date	Status Date	Set of all status dates	date: MM/DD/YYYY
pay_status	Pay Status	Set of all payment statuses	character: size 20
eMarket_invoice_id	eMarket Invoice ID	Set of all eMarket Invoice IDs	number: size 10

payment_type Domain Definitions

Attribute	Domain Name	Description	Domain
payment_type_id	Payment Type ID	Set of all possible payment type IDs	number: size 3
description	Description	Set of all possible descriptions	character: size 20

payment Domain Definitions

Attribute	Domain Name	Description	Domain
payment_id	Payment ID	Set of all possible payment IDs	number
payment_type_id	Payment Type ID	Set of all possible payment type IDs	number: size 3
pay_date	Payment Date	Set of all possible payment dates	date: MM/DD/YYYY
eMarket_invoice_id	eMarket Invoice ID	Set of all eMarket Invoice IDs	number: size 10
amount_paid	Amount Paid	Set of all amounts paid	number



ship_items Domain Definitions

Attribute	Domain Name	Description	Domain
ship_items_id	Ship Item ID	Set of all possible Ship Item IDs	number: size 5
quantity	Quantity	Set of all possible quantities	number: size 20
eMarket_invoice_id	eMarket Invoice ID	Set of all eMarket Invoice IDs	number: size 10
po_item_id	PO Item ID	Set of all possible PO Item IDs	number: size 10

invoice_audit Domain Definitions

Attribute	Domain Name	Description	Domain
invoice_id	Invoice ID	Set of all possible Invoice IDs	number
delete_date	Delete Date	Set of all possible delete dates	date

payment_audit Domain Definitions

Attribute	Domain Name	Description	Domain
invoice_id	Invoice ID	Set of all possible Invoice IDs	number
add_date	Add Date	Set of all possible delete dates	date
amount	Amount	Set of all possible amounts	number

D. Logical Design with the Relational Model

Third Normal Form (3NF)

Underlined = Primary Key
Italicized = Foreign Key

zip (zip, state, city)
vendor (supplier_number, description, fax, phone, address, *zip_code, vendor_employee*)
size (size_code)
product (supplier_part_auxiliary_id, manu_part_number, catalog_number, commodity_code, description, unit_price, *size_code*)
product_size (*supplier_part_auxiliary_id, manu_part_number, size_code*)
commodity (commodity_code, description)
vendor_product(vendor_product_id, *supplier_number, supplier_part_auxiliary_id*, *manu_part_number*, price)
vendor_emp(vendor_emp_id, *supplier_number*, from_invoice_id, first_name, last_name)

accounts (account_number, address, *zip*)
department (org_id, description, *account_number*)
employee (employee_id, first_name, last_name, phone, email, address, *zip_code, org_id*)
recipient (*employee_id*, delivery_address)
purchaser (*employee_id*, description)

purchase_requisition (requisition_number, purpose)
object_code (object_code, description)
po (po_number, po_date, object_code, *purchaser_id*, *recipient_id, purchase_req_id, po_ap_status_id*)
ap_status (ap_status_id, description)
po_ap_status (po_ap_status_id, *po_number, ap_status_id*)
po_items (po_item_id, *po_number*, quantity, taxable, *payable_status_id*)

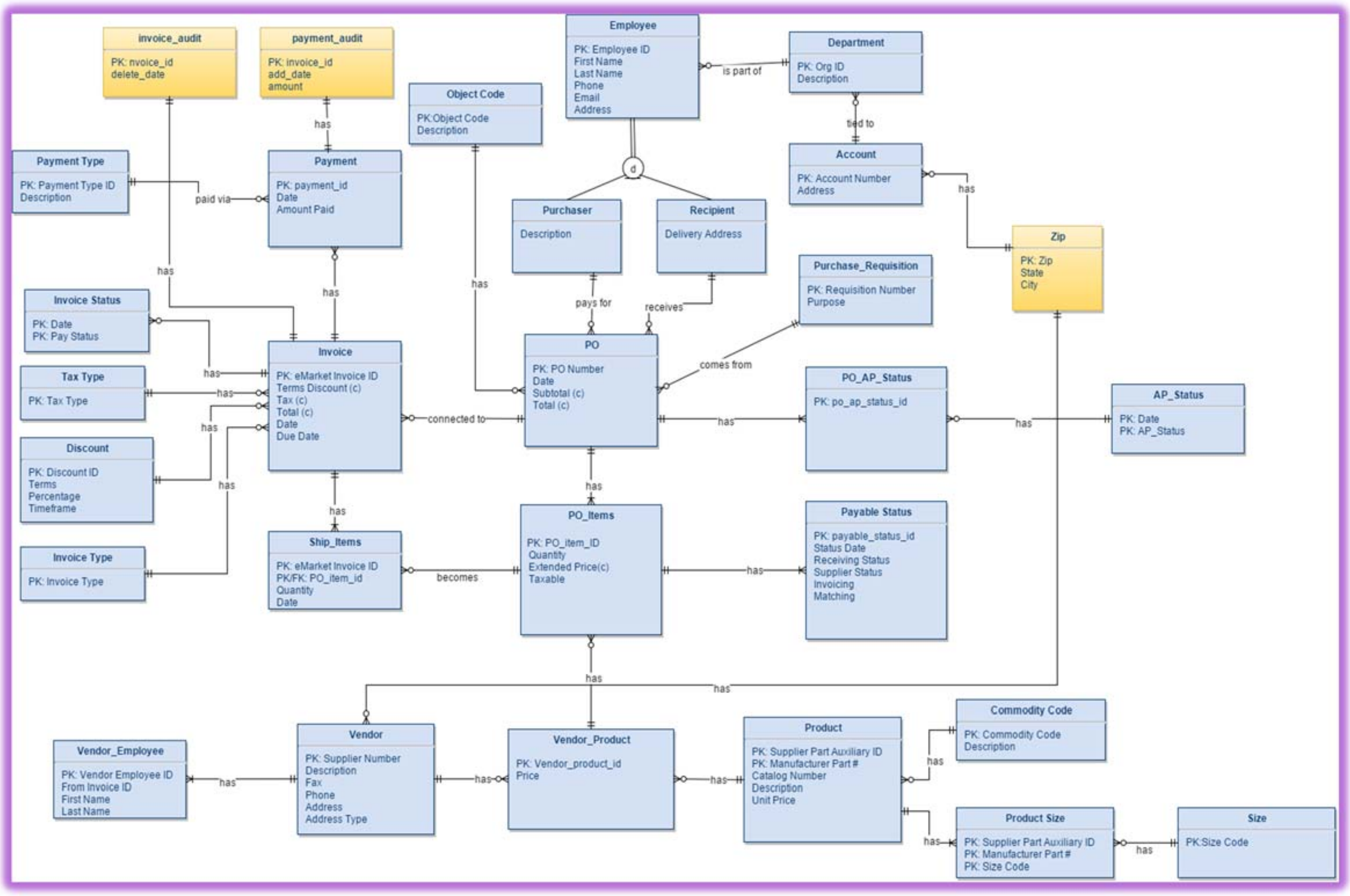
payable_status (payable_status_id, status_date, *po_number*, receiving_status, supplier_status, invoicing, matching, *po_item_id*)

invoice_type(invoice_type_id, description)
tax_type (tax_type_id, description)
discount (discount_id, terms, percentage, timeframe)
invoice(eMarket_invoice_id, *po_number*, invoice_date, due_date, *tax_type_id, discount_id, invoice_type_id*)

invoice_status(status_date, pay_status, *eMarket_invoice_id)*
payment_type (payment_type_id, description)
payment(payment_id, *payment_type_id*, pay_date, *eMarket_invoice_id*, amount_paid)
ship_items(ship_items_id, *quantity, eMarket_invoice_id, po_item_id*)

invoice_audit (invoice_id, delete_date)
payment_audit (invoice_id, add_date, amount)

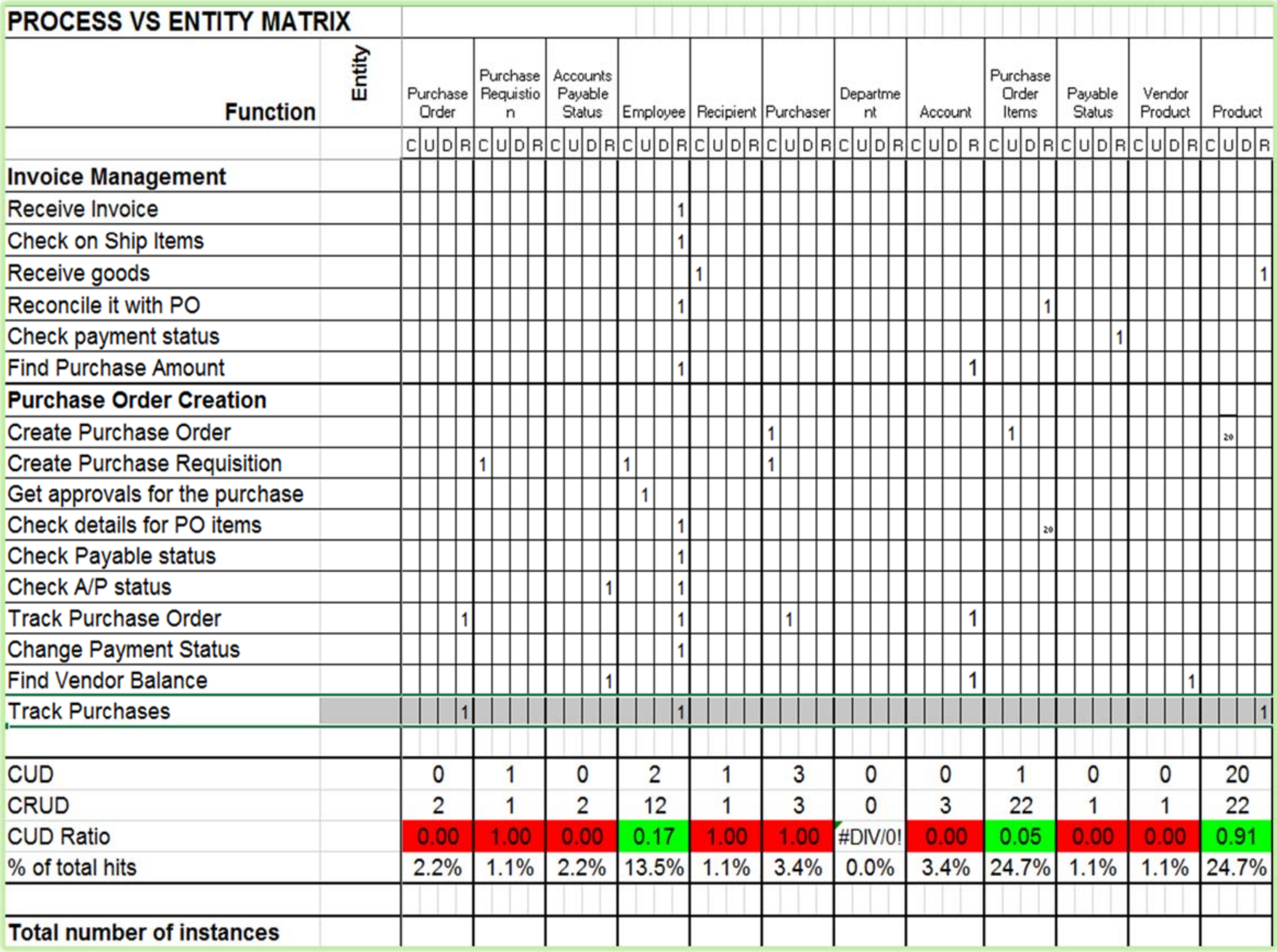
Changes to the CDM



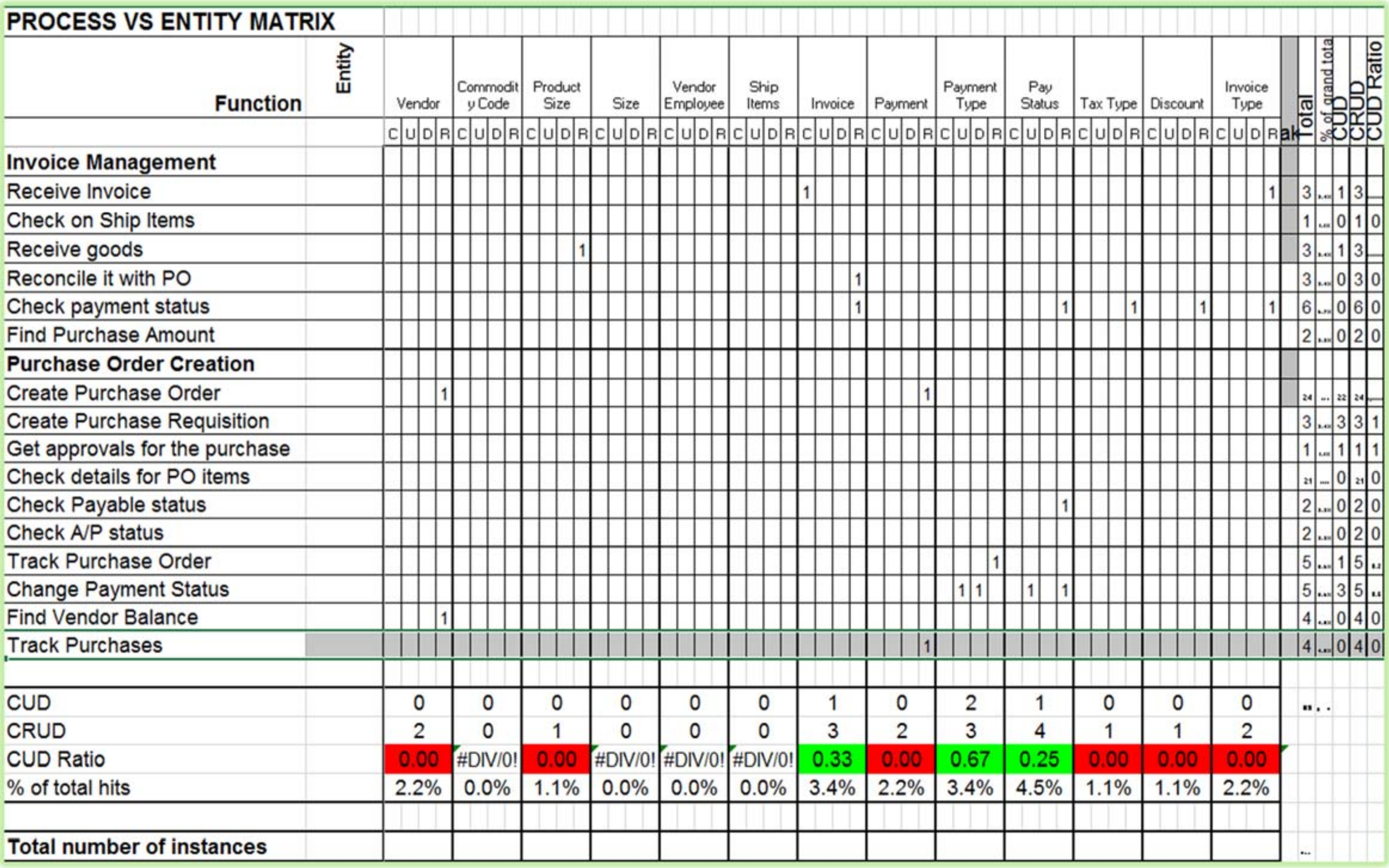
To accommodate our triggers, we added audit tables for invoice and payment. We also added a zip table for our address attributes.

E. Physical Design & Implementation with Relational Model

Process vs. Entity Matrix



PROCESS VS ENTITY MATRIX																																																	
Function	**Entity**	Purchase Order				Purchase Requistion				Accounts Payable Status				Employee				Recipient				Purchaser				Department				Account				Purchase Order Items				Payable Status				Vendor Product				Product			
		C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R
Invoice Management																																																	
Receive Invoice																	1																																
Check on Ship Items																	1																																
Receive goods																		1																															1
Reconcile it with PO																	1																				1												
Check payment status																																									1								
Find Purchase Amount																	1																1																
Purchase Order Creation																																																	
Create Purchase Order																						1													1												20		
Create Purchase Requisition						1								1								1																											
Get approvals for the purchase															1																																		
Check details for PO items																	1																				20												
Check Payable status																	1																																
Check A/P status													1				1																																
Track Purchase Order					1												1						1										1																
Change Payment Status																	1																																
Find Vendor Balance													1																				1												1				
Track Purchases					1												1																																1
CUD		0				1				0				2				1				3				0				0				1				0				0				20			
CRUD		2				1				2				12				1				3				0				3				22				1				1				22			
CUD Ratio		0.00				1.00				0.00				0.17				1.00				1.00				#DIV/0!				0.00				0.05				0.00				0.00				0.91			
% of total hits		2.2%				1.1%				2.2%				13.5%				1.1%				3.4%				0.0%				3.4%				24.7%				1.1%				1.1%				24.7%			
Total number of instances																																																	



PROCESS VS ENTITY MATRIX

Function	Entity	Vendor				Commodity Code				Product Size				Size				Vendor Employee				Ship Items				Invoice			
		C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R
Invoice Management																													
Receive Invoice																										1			
Check on Ship Items																													
Receive goods													1																
Reconcile it with PO																													1
Check payment status																													1
Find Purchase Amount																													
Purchase Order Creation																													
Create Purchase Order					1																								
Create Purchase Requisition																													
Get approvals for the purchase																													
Check details for PO items																													
Check Payable status																													
Check A/P status																													
Track Purchase Order																													
Change Payment Status																													
Find Vendor Balance					1																								
Track Purchases																													
CUD		0				0				0				0				0				0				1			
CRUD		2				0				1				0				0				0				3			
CUD Ratio		0.00				#DIV/0!				0.00				#DIV/0!				#DIV/0!				#DIV/0!				0.33			
% of total hits		2.2%				0.0%				1.1%				0.0%				0.0%				0.0%				3.4%			
Total number of instances																													

Function	Payment				Payment Type				Pay Status				Tax Type				Discount				Invoice Type					Total	% of grand tota	CUD	CRUD	CUD Ratio
	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	C	U	D	R	ak					
Invoice Management																														
Receive Invoice																								1		3	…	1	3	[illegible]
Check on Ship Items																										1	…	0	1	0
Receive goods																										3	…	1	3	[illegible]
Reconcile it with PO																										3	…	0	3	0
Check payment status												1				1				1				1		6	…	0	6	0
Find Purchase Amount																										2	…	0	2	0
Purchase Order Creation																														
Create Purchase Order				1																						24	…	22	24	[illegible]
Create Purchase Requisition																										3	…	3	3	1
Get approvals for the purchase																										1	…	1	1	1
Check details for PO items																										21	…	0	21	0
Check Payable status												1														2	…	0	2	0
Check A/P status																										2	…	0	2	0
Track Purchase Order								1																		5	…	1	5	0.2
Change Payment Status						1	1			1		1														5	…	3	5	0.6
Find Vendor Balance																										4	…	0	4	0
Track Purchases				1																						4	…	0	4	0
CUD	0				2				1				0				0				0					…				
CRUD	2				3				4				1				1				2									
CUD Ratio	0.00				0.67				0.25				0.00				0.00				0.00									
% of total hits	2.2%				3.4%				4.5%				1.1%				1.1%				2.2%									
Total number of instances																										…				

Transaction Analysis Forms

TRANSACTION ANALYSIS FORM (TAF)

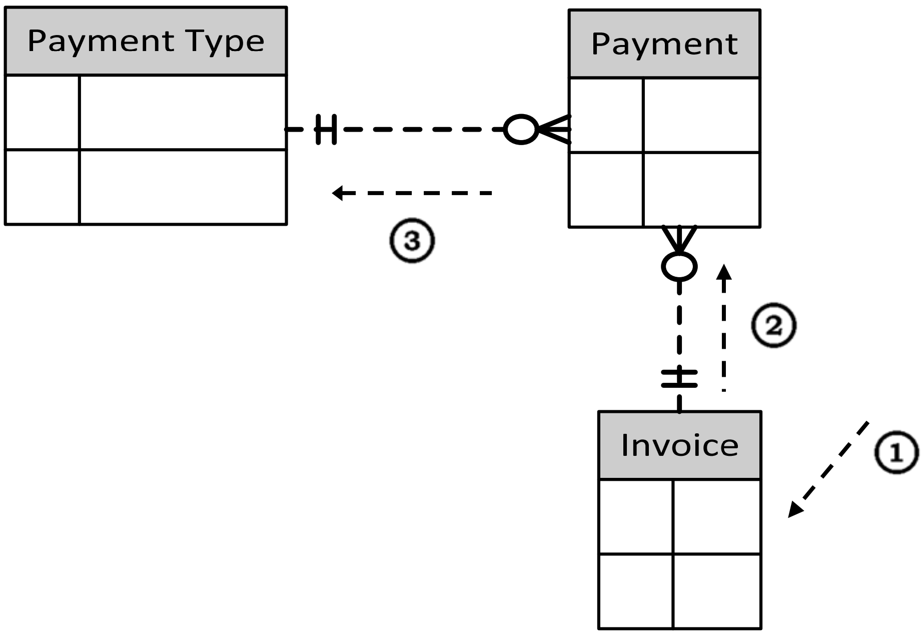
TRANSACTION NO. XJT3D17D9 DATE: 10/12/2015

TRANSACTION NAME: PAYING INVOICE

TRANSACTION VOLUME:

AVERAGE 1/hr PEAK 100/hr

TRANSACTION MAP:



NO.	NAME	TYPE OF ACCESS	No. of Reference	
			Per Tran.	Per period
1	Receiving Invoice	R	1	100
2	Making Payment	R	1	100
3	Selecting Payment Type	R	1	100
TOTAL NUMBER OF REFERENCES			3	300

TRANSACTION ANALYSIS FORM (TAF)

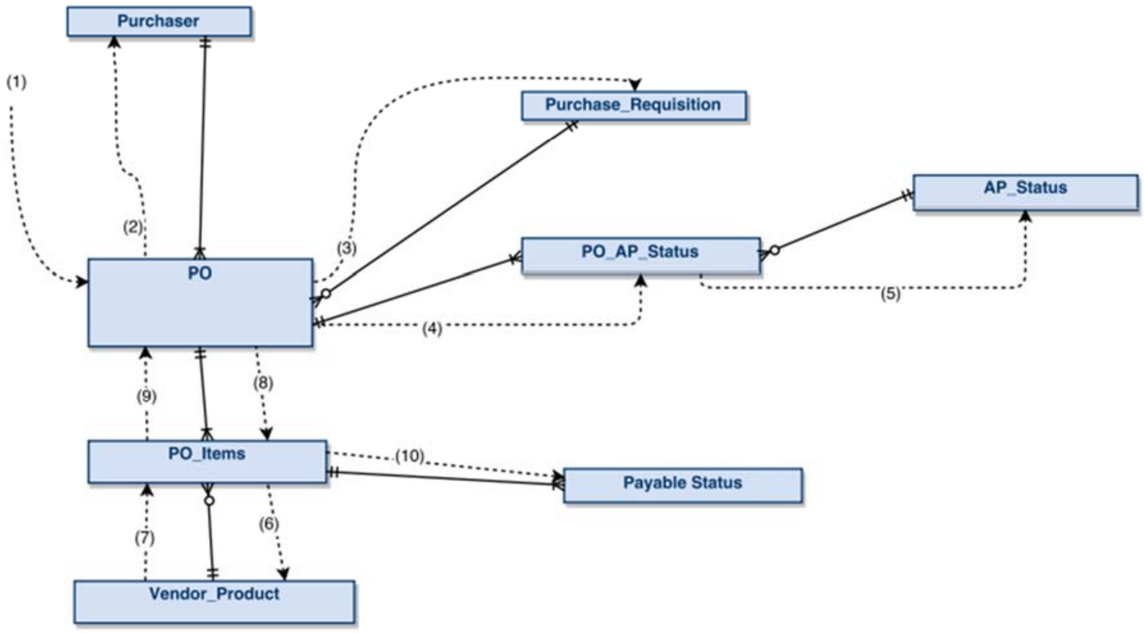
TRANSACTION NO.__XJT3D17D10____ DATE: __10/15/2015______

TRANSACTION NAME: __CREATING PO___________

TRANSACTION VOLUME:

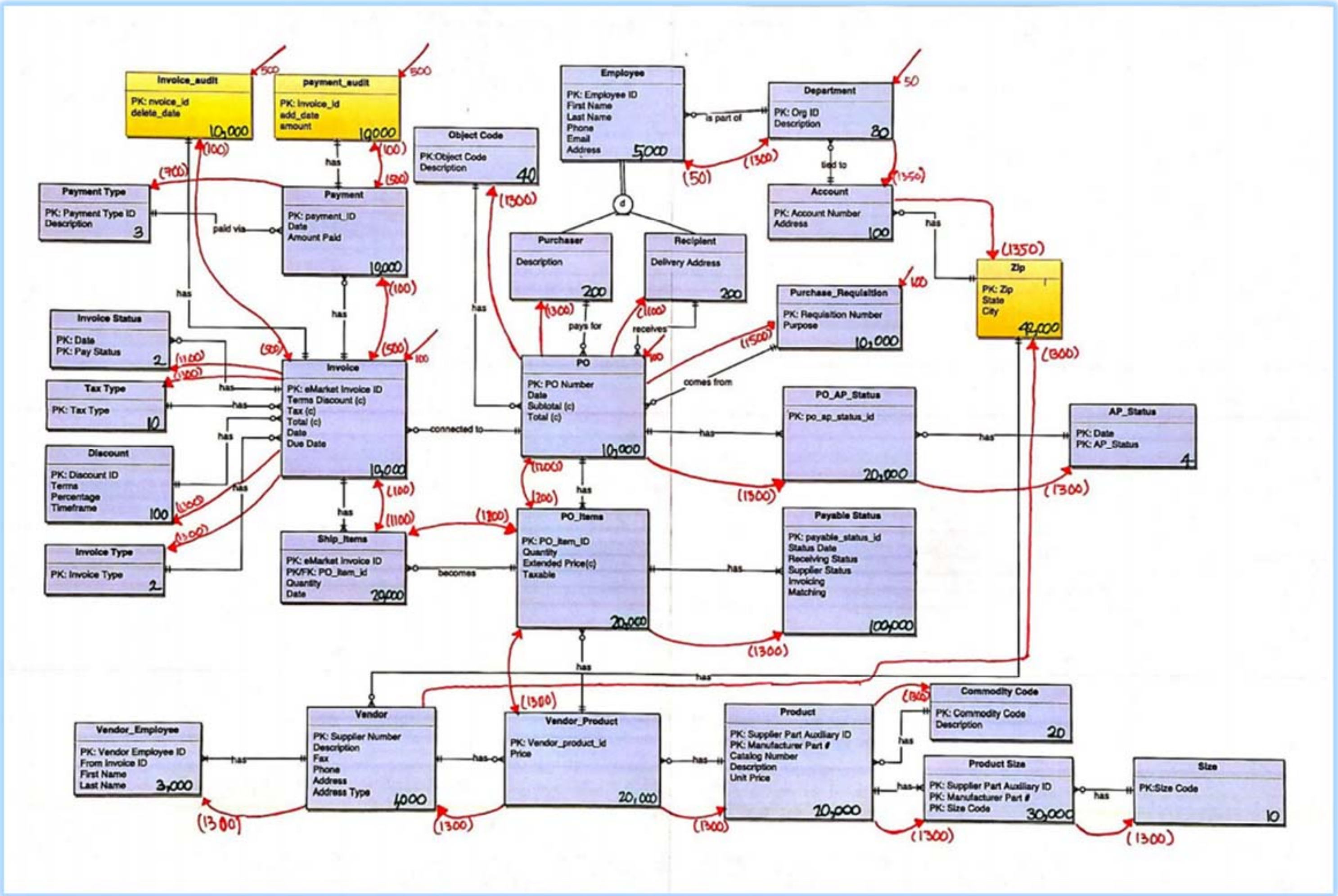
AVERAGE ___1/hr____ PEAK __100/hr___

TRANSACTION MAP:



NO.	NAME	TYPE OF ACCESS	No. of Reference	
			Per Tran.	Per Period
1	Flyin- PO	C	1	100
2	Get Purchaser	R	1	100
3	Get Purchase Req.	R	2	200
4	Create PO_AP_Status	C	1	100
5	Get AP_Status	R	1	100
6	Get Vendor_Product	R	1	100
7	Add to PO_Items	C	1	100
8	Get PO_Items	R	1	100
9	Update PO Total	U	1	100
10	Add to Payable_Status	C	1	100
TOTAL NUMBER OF REFERENCES			3	2800

Composite Usage Map

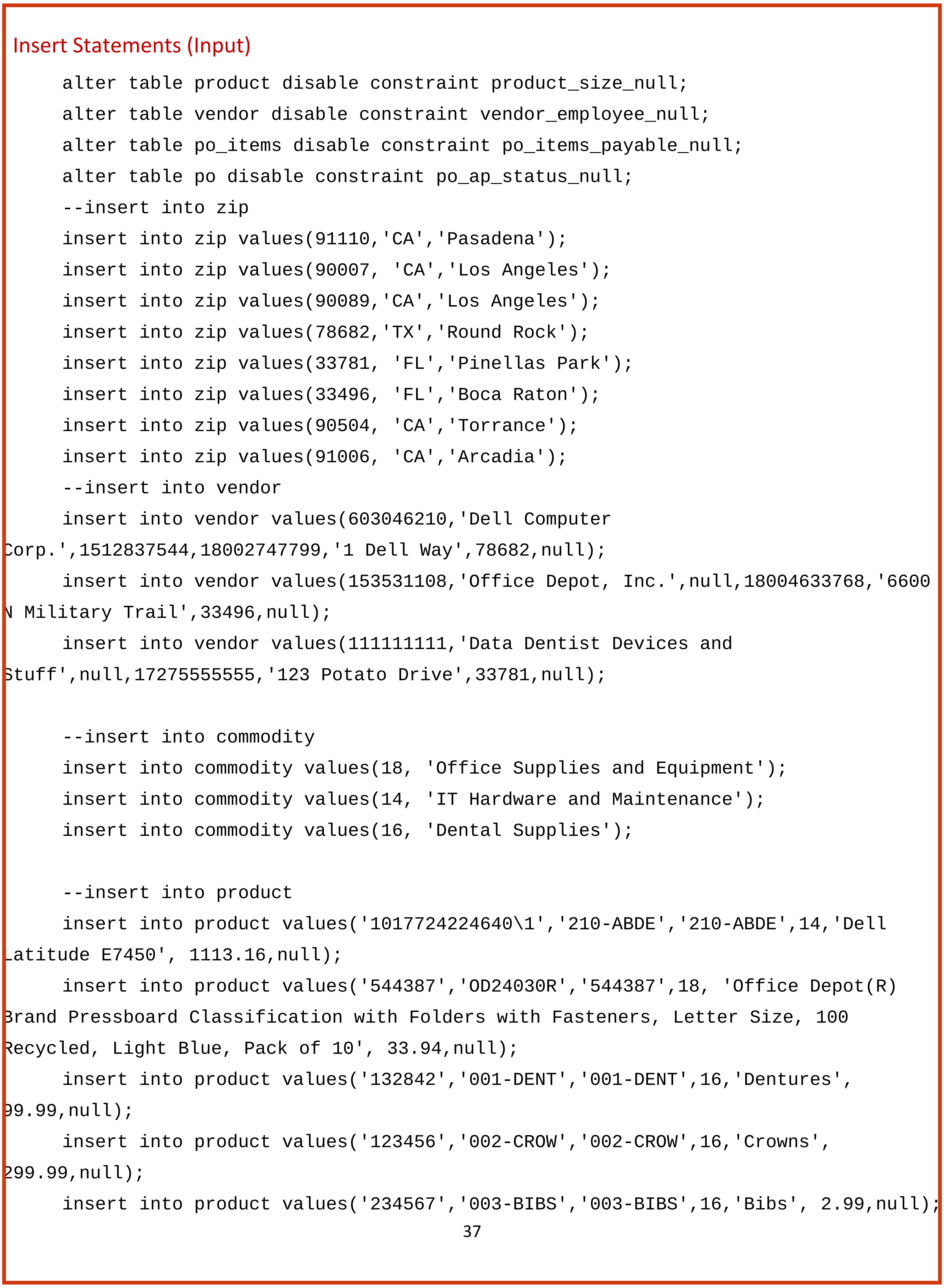


Create Statements (Input)

```
set pagesize 60;
set linesize 100;
purge recyclebin;
set autocommit on;

alter session set nls_date_format = 'mm/dd/yyyy';

drop index to_po_cluster_idx;
drop index payment_audit_idx;
drop index payment_date;
drop index employee_name;
drop index employee_phone;
drop index employee_email;
drop index vendor_emp_name;
drop index po_date_idx;
drop index obj_code_desc;
drop index invoice_date_idx;
drop index product_catalog_num;
drop index vendor_product_price;
drop index product_unit_price;
drop index vendor_name;
drop index vendor_phone;
drop index payable_statuses;
drop index payable_status_dates;
drop index taxable_bool;

drop table invoice_audit cascade constraints;
drop table payment_audit cascade constraints;
DROP TABLE invoice_type CASCADE CONSTRAINTS;
DROP TABLE tax_type CASCADE CONSTRAINTS;
DROP TABLE discount CASCADE CONSTRAINTS;
DROP TABLE invoice CASCADE CONSTRAINTS;
DROP TABLE invoice_status CASCADE CONSTRAINTS;
DROP TABLE payment_type CASCADE CONSTRAINTS;
DROP TABLE payment CASCADE CONSTRAINTS;
DROP TABLE ship_items CASCADE CONSTRAINTS;
Drop table payable_status cascade constraints;
Drop table po_items cascade constraints;
Drop table po_ap_status cascade constraints;
Drop table ap_status cascade constraints;
drop table object_code cascade constraints;
Drop table po cascade constraints;
Drop table purchase_requisition cascade constraints;
Drop table purchaser cascade constraints;
Drop table recipient cascade constraints;
Drop table employee cascade constraints;
Drop table department cascade constraints;
Drop table accounts cascade constraints;
drop table vendor cascade constraints;
drop table product cascade constraints;
drop table commodity cascade constraints;
drop table product_size cascade constraints;
drop table size_codes cascade constraints;
drop table vendor_product cascade constraints;
drop table vendor_emp cascade constraints;
drop table zip cascade constraints;
drop cluster to_po including tables;

create cluster to_po(bob number);
create index to_po_cluster_idx on cluster to_po;

create table zip(
zip number(5) constraint zip_pk primary key,
state varchar2(2),
city varchar2(30)
);

--dropped address_type attribute
create table vendor(
supplier_number number constraint vendor_pk primary key,
description varchar2(30) constraint vendor_desc_null not null,
fax number,
phone number constraint vendor_phone_null not null,
address varchar2(30) constraint vendor_address_null not null,
zip_code number(5) constraint vendor_zip_null not null constraint vendor_zip_fk
references zip
);

create table commodity(
commodity_code number constraint commodity_pk primary key,
description varchar2(50) constraint commodity_desc_null not null
);

create table product(
supplier_part_aux_id varchar2(20) constraint product_pk1_null unique not null,
manu_part_number varchar2(8) constraint product_pk2_null unique not null,
catalog_number varchar(20) constraint catalog_null not null,
commodity_code number constraint product_commodity_fk references commodity,
description varchar2(150) constraint product_desc_null not null,
unit_price number constraint unit_price_null not null constraint unit_price_zero
check(unit_price>0),
constraint product_pk primary key(supplier_part_aux_id, manu_part_number)
);

create table size_codes(
size_code varchar2(2) constraint size_codes_pk primary key
);

create table product_size(
supplier_part_aux_id varchar2(20) constraint supplier_part_fk references
product(supplier_part_aux_id) not null,
manu_part_number varchar2(8) constraint manu_part_fk references
product(manu_part_number) not null,
size_code varchar2(2) constraint size_code_fk references size_codes,
constraint product_size_pk primary
key(supplier_part_aux_id,manu_part_number,size_code)
);

alter table product add (size_code varchar2(2) constraint product_size_fk
references size_codes
                                              constraint product_size_null not
null);
```

```
create table vendor_product(
vendor_product_id number constraint vendor_product_pk primary key,
supplier_number number constraint supplier_num_fk references vendor,
supplier_part_aux_id varchar2(20) constraint supp_part_fk references
product(supplier_part_aux_id),
manu_part_number varchar2(8) constraint manu_part_number_fk references
product(manu_part_number),
price number constraint vendor_product_price_null not null
             constraint check_price check(price>0)
);

create table vendor_emp(
vendor_emp_id number constraint vendor_emp_pk primary key,
supplier_number number constraint vendor_emp_supplier_no_fk references vendor,
from_invoice_id number constraint invoice_id_null not null,
first_name varchar2(15) constraint vendor_emp_first_null not null,
last_name varchar2(15) constraint vendor_emp_last_null not null
);

alter table vendor add(vendor_employee number constraint vendor_employee_fk
references vendor_emp
                                              constraint vendor_employee_null
not null);

create table accounts(
account_number number(12) constraint accounts_pk primary key,
address varchar2(30),
zip number(5) constraint account_zip_fk references zip
);

Create table department(
org_id number(10) constraint department_pk primary key,
description varchar2(30),
account_number number(12) references accounts
);

create table employee(
employee_id number(10) constraint employee_id_pk primary key,
first_name varchar2(10) constraint employee_first_nn not null,
last_name varchar2(10) constraint employee_last_nn not null,
phone number constraint employee_phone_nn not null,
email varchar2(30) constraint employee_email_nn not null,
address varchar2(30),
zip_code number(5) constraint employee_zip_fk references zip,
org_id number(10) constraint employee_org_fk references department
);

create table recipient(
employee_id number constraint recipient_pk primary key constraint recipient_fk
references employee(employee_id),
delivery_address varchar2(30)
);

create table purchaser(
employee_id number constraint purchaser_pk primary key constraint purchaser_fk
references employee(employee_id),
description varchar2(30)
);

create table purchase_requisition(
requisition_number number constraint pur_req_pk primary key,
purpose varchar2(60)
);

create table object_code(
object_code number constraint object_code_pk primary key,
description varchar2(30) constraint object_code_null not null
);

create table po(
po_number number constraint po_pk primary key,
po_date date,
object_code number constraint po_object_code_fk references object_code,
purchaser_id number(10) constraint po_purchaser_fk references employee,
recipient_id number(10) constraint po_recipient_fk references employee,
purchase_req_id number constraint po_pur_req_fk references purchase_requisition
);

Create table ap_status(
ap_status_id number constraint ap_status_pk primary key,
description varchar2(30)
);

Create table po_ap_status(
po_ap_status_id number constraint po_ap_pk primary key,
po_number number constraint po_ap_fk references po,
ap_status_id number constraint ap_status_fk references ap_status
)cluster to_po(po_number);

alter table po add(po_ap_status_id number constraint po_ap_status_fk references
po_ap_status
                                          constraint po_ap_status_null not
null);

Create table po_items(
po_item_id number(5) constraint po_items_pk primary key,
po_number number constraint po_num_item_fk references po,
quantity number constraint quantity_null not null constraint quantity_zero
check(quantity>0),
taxable number
)cluster to_po(po_number);

create table payable_status(
payable_status_id number(5) constraint payable_status_pk primary key,
status_date date constraint payable_status_date_null not null,
po_number number constraint payable_status_po_fk references po,
receiving_status varchar2(30) constraint receiving_status_null not null
                             constraint receiving_status_opts
check(receiving_status = 'None' or receiving_status='Received'),
supplier_status varchar2(30) constraint supplier_status_null not null
                            constraint supplier_status_opts
check(supplier_status='Sent to Supplier' or supplier_status='Shipped'),
invoicing varchar2(30) constraint invoicing_null not null
```

```
                       constraint invoicing_opts check(invoicing='Fully
Invoiced'),
matching varchar2(30) constraint matching_null not null
                      constraint matching_opts check(matching='Fully Matched' or
matching='Not Matched'),
po_item_id number(5) constraint payable_po_item_fk references po_items
)cluster to_po(po_number);

alter table po_items add(payable_status_id number constraint po_items_payable_fk
references payable_status
                                                  constraint
po_items_payable_null not null);

CREATE TABLE invoice_type (
invoice_type_id     number(3)       CONSTRAINT invoice_type_PK PRIMARY KEY,
description         varchar2(20)    CONSTRAINT invoice_type_desc_NN NOT NULL);

CREATE TABLE tax_type (
tax_type_id         number(3)       CONSTRAINT tax_type_PK PRIMARY KEY,
description         varchar(10)     CONSTRAINT tax_type_desc_NN NOT NULL
);

CREATE TABLE discount (
discount_id         number(3)       CONSTRAINT discount_PK PRIMARY KEY,
terms               varchar2(20)    CONSTRAINT discount_desc_NN NOT NULL,
percentage          number(5,3)     CONSTRAINT discount_percentage_NN NOT NULL,
timeframe           number(5)      CONSTRAINT discount_timeframe_NN NOT NULL
);

CREATE TABLE invoice (
eMarket_invoice_id  number(10)      CONSTRAINT invoice_PK PRIMARY KEY,
po_number           number          constraint po_num_fk references po,
invoice_date        date            CONSTRAINT invoide_date_NN NOT NULL,
due_date            date            constraint invoice_due_NN not null,
tax_type_id         number(3,2)     CONSTRAINT invoice_tax_FK references
tax_type
                                    CONSTRAINT invoice_tax_NN NOT NULL,
discount_id         number(3,2)     CONSTRAINT invoice_discount_FK references
discount
                                    CONSTRAINT invoice_discount_NN NOT NULL,
invoice_type_id     number(3)       CONSTRAINT invoice_itype_FK references
invoice_type
                                    CONSTRAINT invoice_itype_NN NOT NULL
)cluster to_po(po_number);

CREATE TABLE invoice_status (
status_date        date,
pay_status         varchar2(20),
eMarket_invoice_id number(10)       CONSTRAINT invoice_status_invoice_FK
references invoice
                                    CONSTRAINT invoice_status_invoice_NN NOT
NULL,
CONSTRAINT invoice_status_PK PRIMARY KEY (status_date,
pay_status,eMarket_invoice_id)
);

CREATE TABLE payment_type(
payment_type_id     number(3)       CONSTRAINT payment_type_PK PRIMARY KEY,
description         varchar2(20)    CONSTRAINT payment_type_desc_NN NOT NULL
);

CREATE TABLE payment(
payment_id number constraint payment_pk primary key,
payment_type_id     number(3)       CONSTRAINT payment_ptype_FK references
payment_type
                                    CONSTRAINT payment_ptype_NN NOT NULL,
pay_date            date,
eMarket_invoice_id  number(10)      CONSTRAINT payment_invoice_FK references
invoice
                                    CONSTRAINT payment_invoice_NN NOT NULL,
amount_paid number constraint amount_paid_null not null constraint
amount_paid_zero check(amount_paid>0)
);

CREATE TABLE ship_items(
ship_items_id       number(5)       CONSTRAINT ship_PK PRIMARY KEY,
quantity            number(20)      CONSTRAINT ship_quantity_NN NOT NULL,
eMarket_invoice_id  number(10)      CONSTRAINT ship_invoice_FK references
invoice
                                    CONSTRAINT ship_invoice_NN NOT NULL,
po_item_id  number(10)              CONSTRAINT ship_poitem_FK references
po_items
                                    CONSTRAINT ship_poitem_NN NOT NULL
);

create table invoice_audit(
invoice_id number constraint invoice_audit_pk primary key,
delete_date date constraint invoice_audit_null not null
);

create table payment_audit(
invoice_id number constraint payment_audit_pk primary key,
add_date date constraint payment_audit_null not null,
amount number constraint amount_null not null
);

create index employee_name on employee(first_name,last_name);
create index employee_phone on employee(phone);
create index employee_email on employee(email);
create index vendor_emp_name on vendor_emp(first_name,last_name);
create index po_date_idx on po(po_date);
create index obj_code_desc on object_code(description);
create index invoice_date_idx on invoice(invoice_date,due_date);
create index product_catalog_num on product(catalog_number);
create index vendor_product_price on vendor_product(price);
create index product_unit_price on product(unit_price);
create index vendor_name on vendor(description);
create index vendor_phone on vendor(phone);
create index payment_date on payment(pay_date);
create index payable_statuses on
payable_status(invoicing,matching,receiving_status,supplier_status);
create index payable_status_dates on payable_status(status_date);
create index taxable_bool on po_items(taxable);
create index payment_audit_idx on payment_audit(invoice_id,add_date,amount);
```



Insert Statements (Input)

```
     alter table product disable constraint product_size_null;
     alter table vendor disable constraint vendor_employee_null;
     alter table po_items disable constraint po_items_payable_null;
     alter table po disable constraint po_ap_status_null;
     --insert into zip
     insert into zip values(91110,'CA','Pasadena');
     insert into zip values(90007, 'CA','Los Angeles');
     insert into zip values(90089,'CA','Los Angeles');
     insert into zip values(78682,'TX','Round Rock');
     insert into zip values(33781, 'FL','Pinellas Park');
     insert into zip values(33496, 'FL','Boca Raton');
     insert into zip values(90504, 'CA','Torrance');
     insert into zip values(91006, 'CA','Arcadia');
     --insert into vendor
     insert into vendor values(603046210,'Dell Computer
Corp.',1512837544,18002747799,'1 Dell Way',78682,null);
     insert into vendor values(153531108,'Office Depot, Inc.',null,18004633768,'6600
N Military Trail',33496,null);
     insert into vendor values(111111111,'Data Dentist Devices and
Stuff',null,17275555555,'123 Potato Drive',33781,null);

     --insert into commodity
     insert into commodity values(18, 'Office Supplies and Equipment');
     insert into commodity values(14, 'IT Hardware and Maintenance');
     insert into commodity values(16, 'Dental Supplies');

     --insert into product
     insert into product values('1017724224640\1','210-ABDE','210-ABDE',14,'Dell
Latitude E7450', 1113.16,null);
     insert into product values('544387','OD24030R','544387',18, 'Office Depot(R)
Brand Pressboard Classification with Folders with Fasteners, Letter Size, 100
Recycled, Light Blue, Pack of 10', 33.94,null);
     insert into product values('132842','001-DENT','001-DENT',16,'Dentures',
99.99,null);
     insert into product values('123456','002-CROW','002-CROW',16,'Crowns',
299.99,null);
     insert into product values('234567','003-BIBS','003-BIBS',16,'Bibs', 2.99,null);
     --insert into size_codes
     insert into size_codes values('EA');
     insert into size_codes values('BX');

     --insert into product_size
     insert into product_size values('1017724224640\1','210-ABDE','EA');
     insert into product_size values('123456','002-CROW','EA');
     insert into product_size values('234567','003-BIBS','EA');
     insert into product_size values('544387','OD24030R','BX');
     insert into product_size values('132842','001-DENT','EA');

     update product set size_code='EA' where supplier_part_aux_id='1017724224640\1'
or supplier_part_aux_id='123456'
     or supplier_part_aux_id='234567' or supplier_part_aux_id='132842';
     update product set size_code='BX' where supplier_part_aux_id='544387';

     --insert into vendor_product
     insert into vendor_product values(1,603046210,'1017724224640\1','210-
ABDE',1113.16);
     insert into vendor_product values(2,111111111,'123456','002-CROW',299.99);
     insert into vendor_product values(3,153531108,'544387','OD24030R',33.94);
     insert into vendor_product values(4,111111111,'234567','003-BIBS',2.99);
     insert into vendor_product values(5,111111111,'132842','001-DENT',99.99);

     --insert into vendor_emp
     insert into vendor_emp values(1,603046210,7426166805,'Jeff','Dahlman');
     insert into vendor_emp values(2,153531108,798851330,'Huy','Ngo');
     insert into vendor_emp values(3,111111111,472942793,'Douglas','Shook');

     update vendor set vendor_employee=1 where supplier_number=603046210;
     update vendor set vendor_employee=2 where supplier_number=153531108;
     update vendor set vendor_employee=3 where supplier_number=111111111;

     --insert into accounts
     insert into accounts values(1213050004,'PO Box 77967',90007);
     insert into accounts values(1234567890,'PO Box 17380',33781);
     --insert into department
     insert into department values(2060603000,'Information Technology',1213050004);
     insert into department values(2123109112,'Meat Selection',1234567890);

     --insert into employee
     insert into employee
values(1234567890,'Brandon','Crabtree',12137406708,'bcrabtre@usc.edu','925 West 34th
St.',90089,2060603000);
     insert into employee values(2345678901, 'Huy','Ngo',17273486341,
'huyngo@usc.edu','1151 W. 36th Pl.',90007,2123109112);
     insert into employee values(3456789012, 'Jane','Oh',17273486341,
'janeoh@usc.edu','Somewhere in Torrance',90504,2123109112);
     insert into employee values(4567890123, 'Wilson','Lin',17273486341,
'wilsonlin@usc.edu','Cardinal Gardens',91006,2123109112);
     insert into employee values(4567890124, 'Douglas','Shook',12135555555,
'shook@usc.edu','Cal Building',90007,2060603000);
     insert into employee values(4567890125, 'Vicky','Dy',12135555554,
'dy@usc.edu','Dental School',90007,2060603000);

     --insert into recipient
     insert into recipient values(4567890124,'Cal Building');
     insert into recipient values(4567890125,'Dental School');

     --insert into purchaser
     insert into purchaser values(3456789012,'Jane Oh''s Account');
     insert into purchaser values(4567890123,'Wilson Lin''s Account');
     insert into purchaser values(2345678901,'1151 W. 36th Pl. Rm. B');
     insert into purchaser values(1234567890,'925 West 34th St.');

     --insert into purchase_requisition
     insert into purchase_requisition values(66710697,'New Laptop for Tele-Dentistry
Clinic per Linda Brookman');
     insert into purchase_requisition values(67052957,'Folders for IT Dept.');
     insert into purchase_requisition values(67052958,'Dental Supplies');
     insert into purchase_requisition values(67052959,'New TV for Bossman');
     insert into purchase_requisition values(67052960,'Sour Gummy Worms');

     --insert into object_code
     insert into object_code values(15102,'General/Project Supplies');
     insert into object_code values(15307,'Office Supplies');

     --insert into po
     insert into po
values(10356325,'10/07/2015',15307,1234567890,4567890125,66710697,null);
     insert into po
values(10353635,'9/30/2015',15102,1234567890,4567890125,67052957,null);
     insert into po
values(10353636,'12/01/2015',15102,2345678901,4567890124,67052958,null);

     --insert ap_status
     insert into ap_status values(1,'Soft Closed');
     insert into ap_status values(2,'Hard Closed');
     insert into ap_status values(3,'Open');
     insert into ap_status values(4,'Pending');

     --insert into po_ap_status
     insert into po_ap_status values(1,10356325,3);
     insert into po_ap_status values(2,10353635,1);
     insert into po_ap_status values(3,10353636,3);

     update po set PO_AP_STATUS_ID=1 where po_number=10356325;
     update po set PO_AP_STATUS_ID=2 where po_number=10353635;
     update po set PO_AP_STATUS_ID=3 where po_number=10353636;

     --insert into po_items
     insert into po_items values(2,10356325,2,1,null);
     insert into po_items values(1,10353635,1,1,null);
     insert into po_items values(3,10353636,1,1,null);
     insert into po_items values(4,10353636,1,1,null);

     --insert into payable_status
     insert into payable_status values(1,'9/30/2015',10353635,'None','Sent to
Supplier','Fully Invoiced','Fully Matched',1);
     insert into payable_status values(2,'8/30/2015',10356325,'None','Sent to
Supplier','Fully Invoiced','Not Matched',2);
```

```
     insert into payable_status values(3,'12/01/2015',10353636,'None','Sent to
Supplier','Fully Invoiced','Not Matched',3);
     insert into payable_status values(4,'12/01/2015',10353636,'None','Sent to
Supplier','Fully Invoiced','Not Matched',4);
     update po_items set payable_status_id=1 where po_item_id=1;
     update po_items set payable_status_id=2 where po_item_id=2;
     update po_items set payable_status_id=3 where po_item_id=3;
     update po_items set payable_status_id=4 where po_item_id=4;

     --insert into invoice_type
     insert into invoice_type values(001,'Invoice');

     --insert into tax_type
     insert into tax_type values(001,'Sales');
     insert into tax_type values(002,'Use');

     --insert into discount
     insert into discount values(1,'0% 30, Net 30',0,30);
     insert into discount values(2,'0% 10, Net 10',0,10);

     --insert into invoice
     insert into invoice
values(100779134,10356325,'10/08/2015','11/07/2015',001,2,001);
     insert into invoice
values(100780781,10353635,'10/07/2015','11/06/2015',001,1,001);
     insert into invoice
values(100780782,10353636,'12/07/2015','01/06/2016',001,1,001);
     insert into invoice
values(100780783,10353636,'12/07/2015','01/06/2016',001,1,001);

     --insert into invoice_status
     insert into invoice_status values('10/07/2015','Payable',100780781);
     insert into invoice_status values('10/08/2015','Payable',100779134);
     insert into invoice_status values('12/07/2015','Payable',100780782);
     insert into invoice_status values('12/07/2015','Payable',100780783);

     --insert into payment_type
     insert into payment_type values(1,'Unknown');
     insert into payment_type values(2,'Employee Card');
     insert into payment_type values(3,'Procurement Card');

     --insert into payment
     insert into payment values(1,2,'12/05/2015',100780782,3.26);
     insert into payment values(2,2,'12/05/2015',100780783,326.99);

     --insert into ship_items
     insert into ship_items values(1,1,100780781,1);
     insert into ship_items values(2,1,100780782,3);
     insert into ship_items values(3,1,100780783,4);
     update invoice_status set pay_status='Paid' where emarket_invoice_id='100780781'
or emarket_invoice_id='100780782' or emarket_invoice_id='100780783';

     select * from payment;
     select * from product;
     select * from product_size;
     select * from vendor;
     select * from vendor_product;
     select * from employee;
     select * from invoice_status;
     select * from invoice;
     select * from po_items;
     select * from po;
     select * from invoice_status;
     select * from ap_status;
     select * from po_ap_status;
     select * from purchaser;
     select * from ship_items;

     alter table product enable constraint product_size_null;
     alter table vendor enable constraint vendor_employee_null;
     alter table po_items enable constraint po_items_payable_null;
     alter table po enable constraint po_ap_status_null;
```

Create Triggers (Input)

```
create or replace trigger invoice_delete_audit
before delete on invoice
for each row
begin
insert into invoice_audit values
(:old.emarket_invoice_id,:old.invoice_date);
end;

create or replace trigger payment_insert_audit
before insert on payment
for each row
begin
insert into payment_audit values
(:new.emarket_invoice_id,:new.pay_date,:new.amount_paid);
end;
```

Queries (Input)

```
    /* show all vendors and the names of the employees that are located in FL */
    select distinct description, vendor_emp.first_name, vendor_emp.last_name
    from zip, vendor, vendor_emp
    where zip.state = 'FL' and zip.zip=vendor.ZIP_CODE and
vendor_emp.vendor_emp_id=vendor.vendor_employee;

    /* show the amount paid and pay statuses from invoices that occurred in December
2015 */
    select PAYMENT.EMARKET_INVOICE_ID, sum(amount_paid) as total, pay_status
    from po,PAYMENT,invoice_status,invoice
    where po.po_date >= '12/01/2015' and po.po_number=invoice.po_number and
pay_status ='Paid' and
    invoice.emarket_invoice_id=PAYMENT.EMARKET_INVOICE_ID and
PAYMENT.EMARKET_INVOICE_ID=invoice_status.emarket_invoice_id
    group by payment.emarket_invoice_id, pay_status;

    /* show the purchase requisitions that did not get fulfilled */
    select distinct requisition_number, purpose
    from purchase_requisition, po
    minus
    select distinct po.PURCHASE_REQ_ID, purpose
    from PURCHASE_REQUISITION,PO
    where po.purchase_req_id=purchase_requisition.requisition_number;

    /* show the invoices paid by 'Employee Card' and is over $100 */
    select emarket_invoice_id, amount_paid
    from payment, payment_type
    where amount_paid>100 and payment_type.description='Employee Card';

    /* show the employee responsible for the most POs past June 2015*/
    with po_count as(select count(po_number)as max_count,purchaser_id from po where
po_date>='06/01/2015' group by PURCHASER_ID)
    select po_count.purchaser_id, po_count.max_count
    from po_count
    where rownum<=1
    group by po_count.purchaser_id, po_count.max_count
    order by po_count.max_count desc;
```

```
     /* show the recipient and PO number for the POs for which the purchaser was
either Huy Ngo or Jane Oh */
     with desired_names as (select distinct employee.employee_id from purchaser,
employee where first_name='Jane' or first_name='Huy'
     and purchaser.employee_id=employee.employee_id)
     select distinct recipient_id, po.PO_NUMBER
     from po, recipient,purchaser,employee,desired_names
     where recipient.employee_id = employee.employee_id and
desired_names.employee_id=po.purchaser_id;
```

Create Statements (Output)

```
Commit Complete
SQL> alter session set nls_date_format = 'mm/dd/yyyy';

Session altered.

Commit Complete
SQL> drop index to_po_cluster_idx;

Index TO_PO_CLUSTER_IDX dropped.

Commit Complete
SQL> drop index payment_audit_idx;

Index PAYMENT_AUDIT_IDX dropped.

Commit Complete
SQL> drop index payment_date;

Index PAYMENT_DATE dropped.

Commit Complete
SQL> drop index employee_name;

Index EMPLOYEE_NAME dropped.

Commit Complete
SQL> drop index employee_phone;

Index EMPLOYEE_PHONE dropped.

Commit Complete
SQL> drop index employee_email;

Index EMPLOYEE_EMAIL dropped.

Commit Complete
SQL> drop index vendor_emp_name;
```

```
Index VENDOR_EMP_NAME dropped.

Commit Complete
SQL> drop index po_date_idx;

Index PO_DATE_IDX dropped.

Commit Complete
SQL> drop index obj_code_desc;

Index OBJ_CODE_DESC dropped.

Commit Complete
SQL> drop index invoice_date_idx;

Index INVOICE_DATE_IDX dropped.

Commit Complete
SQL> drop index product_catalog_num;

Index PRODUCT_CATALOG_NUM dropped.

Commit Complete
SQL> drop index vendor_product_price;

Index VENDOR_PRODUCT_PRICE dropped.

Commit Complete
SQL> drop index product_unit_price;

Index PRODUCT_UNIT_PRICE dropped.

Commit Complete
SQL> drop index vendor_name;

Index VENDOR_NAME dropped.
```

```
Commit Complete
SQL> drop index vendor_phone;

Index VENDOR_PHONE dropped.

Commit Complete
SQL> drop index payable_statuses;

Index PAYABLE_STATUSES dropped.

Commit Complete
SQL> drop index payable_status_dates;

Index PAYABLE_STATUS_DATES dropped.

Commit Complete
SQL> drop index taxable_bool;

Index TAXABLE_BOOL dropped.

Commit Complete
SQL> drop table invoice_audit cascade constraints;

Table INVOICE_AUDIT dropped.

Commit Complete
SQL> drop table payment_audit cascade constraints;

Table PAYMENT_AUDIT dropped.

Commit Complete
SQL> DROP TABLE invoice_type CASCADE CONSTRAINTS;

Table INVOICE_TYPE dropped.

Commit Complete
SQL> DROP TABLE tax_type CASCADE CONSTRAINTS;
```

```
Table TAX_TYPE dropped.

Commit Complete
SQL> DROP TABLE discount CASCADE CONSTRAINTS;

Table DISCOUNT dropped.

Commit Complete
SQL> DROP TABLE invoice CASCADE CONSTRAINTS;

Table INVOICE dropped.

Commit Complete
SQL> DROP TABLE invoice_status CASCADE CONSTRAINTS;

Table INVOICE_STATUS dropped.

Commit Complete
SQL> DROP TABLE payment_type CASCADE CONSTRAINTS;

Table PAYMENT_TYPE dropped.

Commit Complete
SQL> DROP TABLE payment CASCADE CONSTRAINTS;

Table PAYMENT dropped.

Commit Complete
SQL> DROP TABLE ship_items CASCADE CONSTRAINTS;

Table SHIP_ITEMS dropped.

Commit Complete
SQL> Drop table payable_status cascade constraints;

Table PAYABLE_STATUS dropped.

Commit Complete
```

```
SQL> Drop table po_items cascade constraints;

Table PO_ITEMS dropped.

Commit Complete
SQL> Drop table po_ap_status cascade constraints;

Table PO_AP_STATUS dropped.

Commit Complete
SQL> Drop table ap_status cascade constraints;

Table AP_STATUS dropped.

Commit Complete
SQL> drop table object_code cascade constraints;

Table OBJECT_CODE dropped.

Commit Complete
SQL> Drop table po cascade constraints;

Table PO dropped.

Commit Complete
SQL> Drop table purchase_requisition cascade constraints;

Table PURCHASE_REQUISITION dropped.

Commit Complete
SQL> Drop table purchaser cascade constraints;

Table PURCHASER dropped.

Commit Complete
SQL> Drop table recipient cascade constraints;

Table RECIPIENT dropped.
```

```
Commit Complete
SQL> Drop table employee cascade constraints;

Table EMPLOYEE dropped.

Commit Complete
SQL> Drop table department cascade constraints;

Table DEPARTMENT dropped.

Commit Complete
SQL> Drop table accounts cascade constraints;

Table ACCOUNTS dropped.

Commit Complete
SQL> drop table vendor cascade constraints;

Table VENDOR dropped.

Commit Complete
SQL> drop table product cascade constraints;

Table PRODUCT dropped.

Commit Complete
SQL> drop table commodity cascade constraints;

Table COMMODITY dropped.

Commit Complete
SQL> drop table product_size cascade constraints;

Table PRODUCT_SIZE dropped.

Commit Complete
SQL> drop table size_codes cascade constraints;
```

```
Table SIZE_CODES dropped.

Commit Complete
SQL> drop table vendor_product cascade constraints;

Table VENDOR_PRODUCT dropped.

Commit Complete
SQL> drop table vendor_emp cascade constraints;

Table VENDOR_EMP dropped.

Commit Complete
SQL> drop table zip cascade constraints;

Table ZIP dropped.

Commit Complete
SQL> drop cluster to_po including tables;

Cluster TO_PO dropped.

Commit Complete
SQL> create cluster to_po(bob number);

Cluster TO_PO created.

Commit Complete
SQL> create index to_po_cluster_idx on cluster to_po;

Index TO_PO_CLUSTER_IDX created.

Commit Complete
SQL> create table zip(
zip number(5) constraint zip_pk primary key,
state varchar2(2),
city varchar2(30)
```

```
    );

    Table ZIP created.

    Commit Complete
    SQL> --dropped address_type attribute
    Commit Complete
    SQL> create table vendor(
    supplier_number number constraint vendor_pk primary key,
    description varchar2(30) constraint vendor_desc_null not null,
    fax number,
    phone number constraint vendor_phone_null not null,
    address varchar2(30) constraint vendor_address_null not null,
    zip_code number(5) constraint vendor_zip_null not null constraint vendor_zip_fk
references zip
    );

    Table VENDOR created.

    Commit Complete
    SQL> create table commodity(
    commodity_code number constraint commodity_pk primary key,
    description varchar2(50) constraint commodity_desc_null not null
    );

    Table COMMODITY created.

    Commit Complete
    SQL> create table product(
    supplier_part_aux_id varchar2(20) constraint product_pk1_null unique not null,
    manu_part_number varchar2(8) constraint product_pk2_null unique not null,
    catalog_number varchar(20) constraint catalog_null not null,
    commodity_code number constraint product_commodity_fk references commodity,
    description varchar2(150) constraint product_desc_null not null,
    unit_price number constraint unit_price_null not null constraint unit_price_zero
check(unit_price>0),
    constraint product_pk primary key(supplier_part_aux_id, manu_part_number)
    );
```

```
     Table PRODUCT created.

     Commit Complete
     SQL> create table size_codes(
     size_code varchar2(2) constraint size_codes_pk primary key
     );

     Table SIZE_CODES created.

     Commit Complete
     SQL> create table product_size(
     supplier_part_aux_id varchar2(20) constraint supplier_part_fk references
product(supplier_part_aux_id) not null,
     manu_part_number varchar2(8) constraint manu_part_fk references
product(manu_part_number) not null,
     size_code varchar2(2) constraint size_code_fk references size_codes,
     constraint product_size_pk primary
key(supplier_part_aux_id,manu_part_number,size_code)
     );

     Table PRODUCT_SIZE created.

     Commit Complete
     SQL> alter table product add (size_code varchar2(2) constraint product_size_fk
references size_codes
                                                    constraint product_size_null not
null);

     Table PRODUCT altered.

     Commit Complete
     SQL> create table vendor_product(
     vendor_product_id number constraint vendor_product_pk primary key,
     supplier_number number constraint supplier_num_fk references vendor,
     supplier_part_aux_id varchar2(20) constraint supp_part_fk references
product(supplier_part_aux_id),
     manu_part_number varchar2(8) constraint manu_part_number_fk references
product(manu_part_number),
     price number constraint vendor_product_price_null not null
                  constraint check_price check(price>0)
     );

     Table VENDOR_PRODUCT created.

     Commit Complete
     SQL> create table vendor_emp(
     vendor_emp_id number constraint vendor_emp_pk primary key,
     supplier_number number constraint vendor_emp_supplier_no_fk references vendor,
     from_invoice_id number constraint invoice_id_null not null,
     first_name varchar2(15) constraint vendor_emp_first_null not null,
     last_name varchar2(15) constraint vendor_emp_last_null not null
     );

     Table VENDOR_EMP created.

     Commit Complete
     SQL> alter table vendor add(vendor_employee number constraint vendor_employee_fk
references vendor_emp
                                                   constraint vendor_employee_null
not null);

     Table VENDOR altered.

     Commit Complete
     SQL> create table accounts(
     account_number number(12) constraint accounts_pk primary key,
     address varchar2(30),
     zip number(5) constraint account_zip_fk references zip
     );

     Table ACCOUNTS created.

     Commit Complete
     SQL> Create table department(
     org_id number(10) constraint department_pk primary key,
     description varchar2(30),
     account_number number(12) references accounts
     );

     Table DEPARTMENT created.

     Commit Complete
     SQL> create table employee(
     employee_id number(10) constraint employee_id_pk primary key,
     first_name varchar2(10) constraint employee_first_nn not null,
     last_name varchar2(10) constraint employee_last_nn not null,
     phone number constraint employee_phone_nn not null,
     email varchar2(30) constraint employee_email_nn not null,
     address varchar2(30),
     zip_code number(5) constraint employee_zip_fk references zip,
     org_id number(10) constraint employee_org_fk references department
     );

     Table EMPLOYEE created.

     Commit Complete
     SQL> create table recipient(
     employee_id number constraint recipient_pk primary key constraint recipient_fk
references employee(employee_id),
     delivery_address varchar2(30)
     );

     Table RECIPIENT created.

     Commit Complete
     SQL> create table purchaser(
     employee_id number constraint purchaser_pk primary key constraint purchaser_fk
references employee(employee_id),
     description varchar2(30)
     );

     Table PURCHASER created.
```

```
Commit Complete
SQL> create table purchase_requisition(
requisition_number number constraint pur_req_pk primary key,
purpose varchar2(60)
);

Table PURCHASE_REQUISITION created.

Commit Complete
SQL> create table object_code(
object_code number constraint object_code_pk primary key,
description varchar2(30) constraint object_code_null not null
);

Table OBJECT_CODE created.

Commit Complete
SQL> create table po(
po_number number constraint po_pk primary key,
po_date date,
object_code number constraint po_object_code_fk references object_code,
purchaser_id number(10) constraint po_purchaser_fk references employee,
recipient_id number(10) constraint po_recipient_fk references employee,
purchase_req_id number constraint po_pur_req_fk references purchase_requisition
);

Table PO created.

Commit Complete
SQL> Create table ap_status(
ap_status_id number constraint ap_status_pk primary key,
description varchar2(30)
);

Table AP_STATUS created.

Commit Complete
```

```
     SQL> Create table po_ap_status(
     po_ap_status_id number constraint po_ap_pk primary key,
     po_number number constraint po_ap_fk references po,
     ap_status_id number constraint ap_status_fk references ap_status
     )cluster to_po(po_number);

     Table PO_AP_STATUS created.

     Commit Complete
     SQL> alter table po add(po_ap_status_id number constraint po_ap_status_fk
references po_ap_status
                                                   constraint po_ap_status_null not
null);

     Table PO altered.

     Commit Complete
     SQL> Create table po_items(
     po_item_id number(5) constraint po_items_pk primary key,
     po_number number constraint po_num_item_fk references po,
     quantity number constraint quantity_null not null constraint quantity_zero
check(quantity>0),
     taxable number
     )cluster to_po(po_number);

     Table PO_ITEMS created.

     Commit Complete
     SQL> create table payable_status(
     payable_status_id number(5) constraint payable_status_pk primary key,
     status_date date constraint payable_status_date_null not null,
     po_number number constraint payable_status_po_fk references po,
     receiving_status varchar2(30) constraint receiving_status_null not null
                                  constraint receiving_status_opts
check(receiving_status = 'None' or receiving_status='Received'),
     supplier_status varchar2(30) constraint supplier_status_null not null
                                 constraint supplier_status_opts
check(supplier_status='Sent to Supplier' or supplier_status='Shipped'),
     invoicing varchar2(30) constraint invoicing_null not null
                            constraint invoicing_opts check(invoicing='Fully
Invoiced'),
     matching varchar2(30) constraint matching_null not null
                           constraint matching_opts check(matching='Fully Matched' or
matching='Not Matched'),
     po_item_id number(5) constraint payable_po_item_fk references po_items
     )cluster to_po(po_number);

     Table PAYABLE_STATUS created.

     Commit Complete
     SQL> alter table po_items add(payable_status_id number constraint
po_items_payable_fk references payable_status
                                                        constraint
po_items_payable_null not null);

     Table PO_ITEMS altered.

     Commit Complete
     SQL> CREATE TABLE invoice_type (
     invoice_type_id     number(3)       CONSTRAINT invoice_type_PK PRIMARY KEY,
     description         varchar2(20)    CONSTRAINT invoice_type_desc_NN NOT NULL);

     Table INVOICE_TYPE created.

     Commit Complete
     SQL> CREATE TABLE tax_type (
     tax_type_id         number(3)       CONSTRAINT tax_type_PK PRIMARY KEY,
     description         varchar(10)     CONSTRAINT tax_type_desc_NN NOT NULL
     );

     Table TAX_TYPE created.

     Commit Complete
     SQL> CREATE TABLE discount (
     discount_id         number(3)       CONSTRAINT discount_PK PRIMARY KEY,
     terms               varchar2(20)    CONSTRAINT discount_desc_NN NOT NULL,
     percentage          number(5,3)     CONSTRAINT discount_percentage_NN NOT NULL,
     timeframe           number(5)      CONSTRAINT discount_timeframe_NN NOT NULL
     );

     Table DISCOUNT created.

     Commit Complete
     SQL> CREATE TABLE invoice (
     eMarket_invoice_id  number(10)      CONSTRAINT invoice_PK PRIMARY KEY,
     po_number           number          constraint po_num_fk references po,
     invoice_date        date            CONSTRAINT invoide_date_NN NOT NULL,
     due_date            date            constraint invoice_due_NN not null,
     tax_type_id         number(3,2)     CONSTRAINT invoice_tax_FK references
tax_type
                                         CONSTRAINT invoice_tax_NN NOT NULL,
     discount_id         number(3,2)     CONSTRAINT invoice_discount_FK references
discount
                                         CONSTRAINT invoice_discount_NN NOT NULL,
     invoice_type_id     number(3)       CONSTRAINT invoice_itype_FK references
invoice_type
                                         CONSTRAINT invoice_itype_NN NOT NULL
     )cluster to_po(po_number);

     Table INVOICE created.

     Commit Complete
     SQL> CREATE TABLE invoice_status (
     status_date        date,
     pay_status         varchar2(20),
     eMarket_invoice_id number(10)       CONSTRAINT invoice_status_invoice_FK
references invoice
                                         CONSTRAINT invoice_status_invoice_NN NOT
NULL,
     CONSTRAINT invoice_status_PK PRIMARY KEY (status_date,
pay_status,eMarket_invoice_id)
     );

     Table INVOICE_STATUS created.
     Commit Complete
     SQL> CREATE TABLE payment_type(
     payment_type_id     number(3)       CONSTRAINT payment_type_PK PRIMARY KEY,
     description         varchar2(20)    CONSTRAINT payment_type_desc_NN NOT NULL
     );

     Table PAYMENT_TYPE created.

     Commit Complete
     SQL> CREATE TABLE payment(
     payment_id number constraint payment_pk primary key,
     payment_type_id     number(3)       CONSTRAINT payment_ptype_FK references
payment_type
                                         CONSTRAINT payment_ptype_NN NOT NULL,
     pay_date            date,
     eMarket_invoice_id  number(10)      CONSTRAINT payment_invoice_FK references
invoice
                                         CONSTRAINT payment_invoice_NN NOT NULL,
     amount_paid number constraint amount_paid_null not null constraint
amount_paid_zero check(amount_paid>0)
     );

     Table PAYMENT created.

     Commit Complete
     SQL> CREATE TABLE ship_items(
     ship_items_id       number(5)       CONSTRAINT ship_PK PRIMARY KEY,
     quantity            number(20)      CONSTRAINT ship_quantity_NN NOT NULL,
     eMarket_invoice_id  number(10)      CONSTRAINT ship_invoice_FK references
invoice
                                         CONSTRAINT ship_invoice_NN NOT NULL,
     po_item_id  number(10)              CONSTRAINT ship_poitem_FK references
po_items
                                         CONSTRAINT ship_poitem_NN NOT NULL
     );

     Table SHIP_ITEMS created.
```

```
Commit Complete
SQL> create table invoice_audit(
invoice_id number constraint invoice_audit_pk primary key,
delete_date date constraint invoice_audit_null not null
);

Table INVOICE_AUDIT created.

Commit Complete
SQL> create table payment_audit(
invoice_id number constraint payment_audit_null not null,
add_date date constraint payment_date_null not null,
amount number constraint amount_null not null,
constraint payment_audit_pk primary key(invoice_id,add_date)
);

Table PAYMENT_AUDIT created.

Commit Complete
SQL> create index employee_name on employee(first_name,last_name);

Index EMPLOYEE_NAME created.

Commit Complete
SQL> create index employee_phone on employee(phone);

Index EMPLOYEE_PHONE created.

Commit Complete
SQL> create index employee_email on employee(email);

Index EMPLOYEE_EMAIL created.

Commit Complete
SQL> create index vendor_emp_name on vendor_emp(first_name,last_name);

Index VENDOR_EMP_NAME created.
```

```
Commit Complete
SQL> create index po_date_idx on po(po_date);

Index PO_DATE_IDX created.

Commit Complete
SQL> create index obj_code_desc on object_code(description);

Index OBJ_CODE_DESC created.

Commit Complete
SQL> create index invoice_date_idx on invoice(invoice_date,due_date);

Index INVOICE_DATE_IDX created.

Commit Complete
SQL> create index product_catalog_num on product(catalog_number);

Index PRODUCT_CATALOG_NUM created.

Commit Complete
SQL> create index vendor_product_price on vendor_product(price);

Index VENDOR_PRODUCT_PRICE created.

Commit Complete
SQL> create index product_unit_price on product(unit_price);

Index PRODUCT_UNIT_PRICE created.

Commit Complete
SQL> create index vendor_name on vendor(description);

Index VENDOR_NAME created.

Commit Complete
SQL> create index vendor_phone on vendor(phone);
```

```
     Index VENDOR_PHONE created.

     Commit Complete
     SQL> create index payment_date on payment(pay_date);

     Index PAYMENT_DATE created.

     Commit Complete
     SQL> create index payable_statuses on
payable_status(invoicing,matching,receiving_status,supplier_status);

     Index PAYABLE_STATUSES created.

     Commit Complete
     SQL> create index payable_status_dates on payable_status(status_date);

     Index PAYABLE_STATUS_DATES created.

     Commit Complete
     SQL> create index taxable_bool on po_items(taxable);

     Index TAXABLE_BOOL created.

     Commit Complete
     SQL> create index payment_audit_idx on
payment_audit(invoice_id,add_date,amount);

     Index PAYMENT_AUDIT_IDX created.

     Commit Complete
     SQL> spool off
```

Insert Statements and Select * From Tables (Output)

```
SQL> alter table product disable constraint product_size_null;

Table PRODUCT altered.

SQL> alter table vendor disable constraint vendor_employee_null;

Table VENDOR altered.

SQL> alter table po_items disable constraint po_items_payable_null;

Table PO_ITEMS altered.

SQL> alter table po disable constraint po_ap_status_null;

Table PO altered.

SQL> --insert into zip
SQL> insert into zip values(91110,'CA','Pasadena');

1 row inserted.

SQL> insert into zip values(90007, 'CA','Los Angeles');

1 row inserted.

SQL> insert into zip values(90089,'CA','Los Angeles');

1 row inserted.

SQL> insert into zip values(78682,'TX','Round Rock');

1 row inserted.

SQL> insert into zip values(33781, 'FL','Pinellas Park');

1 row inserted.
```

```
     SQL> insert into zip values(33496, 'FL','Boca Raton');

     1 row inserted.

     SQL> insert into zip values(90504, 'CA','Torrance');

     1 row inserted.

     SQL> insert into zip values(91006, 'CA','Arcadia');

     1 row inserted.

     SQL> --insert into vendor
     SQL> insert into vendor values(603046210,'Dell Computer
Corp.',1512837544,18002747799,'1 Dell Way',78682,null);

     1 row inserted.

     SQL> insert into vendor values(153531108,'Office Depot,
Inc.',null,18004633768,'6600 N Military Trail',33496,null);

     1 row inserted.

     SQL> insert into vendor values(111111111,'Data Dentist Devices and
Stuff',null,17275555555,'123 Potato Drive',33781,null);

     1 row inserted.

     SQL> --insert into commodity
     SQL> insert into commodity values(18, 'Office Supplies and Equipment');

     1 row inserted.

     SQL> insert into commodity values(14, 'IT Hardware and Maintenance');

     1 row inserted.

     SQL> insert into commodity values(16, 'Dental Supplies');
     1 row inserted.

     SQL> --insert into product
     SQL> insert into product values('1017724224640\1','210-ABDE','210-ABDE',14,'Dell
Latitude E7450', 1113.16,null);

     1 row inserted.

     SQL> insert into product values('544387','OD24030R','544387',18, 'Office
Depot(R) Brand Pressboard Classification with Folders with Fasteners, Letter Size,
100 Recycled, Light Blue, Pack of 10', 33.94,null);

     1 row inserted.

     SQL> insert into product values('132842','001-DENT','001-DENT',16,'Dentures',
99.99,null);

     1 row inserted.

     SQL> insert into product values('123456','002-CROW','002-CROW',16,'Crowns',
299.99,null);

     1 row inserted.

     SQL> insert into product values('234567','003-BIBS','003-BIBS',16,'Bibs',
2.99,null);

     1 row inserted.

     SQL> --insert into size_codes
     SQL> insert into size_codes values('EA');

     1 row inserted.

     SQL> insert into size_codes values('BX');

     1 row inserted.
     SQL> --insert into product_size
     SQL> insert into product_size values('1017724224640\1','210-ABDE','EA');

     1 row inserted.

     SQL> insert into product_size values('123456','002-CROW','EA');

     1 row inserted.

     SQL> insert into product_size values('234567','003-BIBS','EA');

     1 row inserted.

     SQL> insert into product_size values('544387','OD24030R','BX');

     1 row inserted.

     SQL> insert into product_size values('132842','001-DENT','EA');

     1 row inserted.

     SQL> update product set size_code='EA' where
supplier_part_aux_id='1017724224640\1' or supplier_part_aux_id='123456'
     or supplier_part_aux_id='234567' or supplier_part_aux_id='132842';

     4 rows updated.

     SQL> update product set size_code='BX' where supplier_part_aux_id='544387';

     1 row updated.

     SQL> --insert into vendor_product
     SQL> insert into vendor_product values(1,603046210,'1017724224640\1','210-
ABDE',1113.16);

     1 row inserted.
```

```
SQL> insert into vendor_product values(2,111111111,'123456','002-CROW',299.99);

1 row inserted.

SQL> insert into vendor_product values(3,153531108,'544387','OD24030R',33.94);

1 row inserted.

SQL> insert into vendor_product values(4,111111111,'234567','003-BIBS',2.99);

1 row inserted.

SQL> insert into vendor_product values(5,111111111,'132842','001-DENT',99.99);

1 row inserted.

SQL> --insert into vendor_emp
SQL> insert into vendor_emp values(1,603046210,7426166805,'Jeff','Dahlman');

1 row inserted.

SQL> insert into vendor_emp values(2,153531108,798851330,'Huy','Ngo');

1 row inserted.

SQL> insert into vendor_emp values(3,111111111,472942793,'Douglas','Shook');

1 row inserted.

SQL> update vendor set vendor_employee=1 where supplier_number=603046210;

1 row updated.

SQL> update vendor set vendor_employee=2 where supplier_number=153531108;

1 row updated.

SQL> update vendor set vendor_employee=3 where supplier_number=111111111;
```

```
     1 row updated.

     SQL> --insert into accounts
     SQL> insert into accounts values(1213050004,'PO Box 77967',90007);

     1 row inserted.

     SQL> insert into accounts values(1234567890,'PO Box 17380',33781);

     1 row inserted.

     SQL> --insert into department
     SQL> insert into department values(2060603000,'Information
Technology',1213050004);

     1 row inserted.

     SQL> insert into department values(2123109112,'Meat Selection',1234567890);

     1 row inserted.

     SQL> --insert into employee
     SQL> insert into employee
values(1234567890,'Brandon','Crabtree',12137406708,'bcrabtre@usc.edu','925 West 34th
St.',90089,2060603000);

     1 row inserted.

     SQL> insert into employee values(2345678901, 'Huy','Ngo',17273486341,
'huyngo@usc.edu','1151 W. 36th Pl.',90007,2123109112);

     1 row inserted.

     SQL> insert into employee values(3456789012, 'Jane','Oh',17273486341,
'janeoh@usc.edu','Somewhere in Torrance',90504,2123109112);

     1 row inserted.
     SQL> insert into employee values(4567890123, 'Wilson','Lin',17273486341,
'wilsonlin@usc.edu','Cardinal Gardens',91006,2123109112);

     1 row inserted.

     SQL> insert into employee values(4567890124, 'Douglas','Shook',12135555555,
'shook@usc.edu','Cal Building',90007,2060603000);

     1 row inserted.

     SQL> insert into employee values(4567890125, 'Vicky','Dy',12135555554,
'dy@usc.edu','Dental School',90007,2060603000);

     1 row inserted.

     SQL> --insert into recipient
     SQL> insert into recipient values(4567890124,'Cal Building');

     1 row inserted.

     SQL> insert into recipient values(4567890125,'Dental School');

     1 row inserted.

     SQL> --insert into purchaser
     SQL> insert into purchaser values(3456789012,'Jane Oh''s Account');

     1 row inserted.

     SQL> insert into purchaser values(4567890123,'Wilson Lin''s Account');

     1 row inserted.

     SQL> insert into purchaser values(2345678901,'1151 W. 36th Pl. Rm. B');

     1 row inserted.
     SQL> insert into purchaser values(1234567890,'925 West 34th St.');

     1 row inserted.

     SQL> --insert into purchase_requisition
     SQL> insert into purchase_requisition values(66710697,'New Laptop for Tele-
Dentistry Clinic per Linda Brookman');

     1 row inserted.

     SQL> insert into purchase_requisition values(67052957,'Folders for IT Dept.');

     1 row inserted.

     SQL> insert into purchase_requisition values(67052958,'Dental Supplies');

     1 row inserted.

     SQL> insert into purchase_requisition values(67052959,'New TV for Bossman');

     1 row inserted.

     SQL> insert into purchase_requisition values(67052960,'Sour Gummy Worms');

     1 row inserted.

     SQL> --insert into object_code
     SQL> insert into object_code values(15102,'General/Project Supplies');

     1 row inserted.

     SQL> insert into object_code values(15307,'Office Supplies');

     1 row inserted.

     SQL> --insert into po
     SQL> insert into po
values(10356325,'10/07/2015',15307,1234567890,4567890125,66710697,null);
```

```
     1 row inserted.

     SQL> insert into po
values(10353635,'9/30/2015',15102,1234567890,4567890125,67052957,null);

     1 row inserted.

     SQL> insert into po
values(10353636,'12/01/2015',15102,2345678901,4567890124,67052958,null);

     1 row inserted.

     SQL> --insert ap_status
     SQL> insert into ap_status values(1,'Soft Closed');

     1 row inserted.

     SQL> insert into ap_status values(2,'Hard Closed');

     1 row inserted.

     SQL> insert into ap_status values(3,'Open');

     1 row inserted.

     SQL> insert into ap_status values(4,'Pending');

     1 row inserted.

     SQL> --insert into po_ap_status
     SQL> insert into po_ap_status values(1,10356325,3);

     1 row inserted.

     SQL> insert into po_ap_status values(2,10353635,1);

     1 row inserted.
```

```
SQL> insert into po_ap_status values(3,10353636,3);

1 row inserted.

SQL> update po set PO_AP_STATUS_ID=1 where po_number=10356325;

1 row updated.

SQL> update po set PO_AP_STATUS_ID=2 where po_number=10353635;

1 row updated.

SQL> update po set PO_AP_STATUS_ID=3 where po_number=10353636;

1 row updated.

SQL> --insert into po_items
SQL> insert into po_items values(2,10356325,2,1,null);

1 row inserted.

SQL> insert into po_items values(1,10353635,1,1,null);

1 row inserted.

SQL> insert into po_items values(3,10353636,1,1,null);

1 row inserted.

SQL> insert into po_items values(4,10353636,1,1,null);

1 row inserted.

SQL> --insert into payable_status
SQL> insert into payable_status values(1,'9/30/2015',10353635,'None','Sent to
Supplier','Fully Invoiced','Fully Matched',1);
```

```
     1 row inserted.

     SQL> insert into payable_status values(2,'8/30/2015',10356325,'None','Sent to
Supplier','Fully Invoiced','Not Matched',2);

     1 row inserted.

     SQL> insert into payable_status values(3,'12/01/2015',10353636,'None','Sent to
Supplier','Fully Invoiced','Not Matched',3);

     1 row inserted.

     SQL> insert into payable_status values(4,'12/01/2015',10353636,'None','Sent to
Supplier','Fully Invoiced','Not Matched',4);

     1 row inserted.

     SQL> update po_items set payable_status_id=1 where po_item_id=1;

     1 row updated.

     SQL> update po_items set payable_status_id=2 where po_item_id=2;

     1 row updated.

     SQL> update po_items set payable_status_id=3 where po_item_id=3;

     1 row updated.

     SQL> update po_items set payable_status_id=4 where po_item_id=4;

     1 row updated.

     SQL> --insert into invoice_type
     SQL> insert into invoice_type values(001,'Invoice');

     1 row inserted.
     SQL> --insert into tax_type
     SQL> insert into tax_type values(001,'Sales');

     1 row inserted.

     SQL> insert into tax_type values(002,'Use');

     1 row inserted.

     SQL> --insert into discount
     SQL> insert into discount values(1,'0% 30, Net 30',0,30);

     1 row inserted.

     SQL> insert into discount values(2,'0% 10, Net 10',0,10);

     1 row inserted.

     SQL> --insert into invoice
     SQL> insert into invoice
values(100779134,10356325,'10/08/2015','11/07/2015',001,2,001);

     1 row inserted.

     SQL> insert into invoice
values(100780781,10353635,'10/07/2015','11/06/2015',001,1,001);

     1 row inserted.

     SQL> insert into invoice
values(100780782,10353636,'12/07/2015','01/06/2016',001,1,001);

     1 row inserted.

     SQL> insert into invoice
values(100780783,10353636,'12/07/2015','01/06/2016',001,1,001);

     1 row inserted.
```

```
SQL> --insert into invoice_status
SQL> insert into invoice_status values('10/07/2015','Payable',100780781);

1 row inserted.

SQL> insert into invoice_status values('10/08/2015','Payable',100779134);

1 row inserted.

SQL> insert into invoice_status values('12/07/2015','Payable',100780782);

1 row inserted.

SQL> insert into invoice_status values('12/07/2015','Payable',100780783);

1 row inserted.

SQL> --insert into payment_type
SQL> insert into payment_type values(1,'Unknown');

1 row inserted.

SQL> insert into payment_type values(2,'Employee Card');

1 row inserted.

SQL> insert into payment_type values(3,'Procurement Card');

1 row inserted.

SQL> --insert into payment
SQL> insert into payment values(1,2,'12/05/2015',100780782,3.26);

1 row inserted.

SQL> insert into payment values(2,2,'12/05/2015',100780783,326.99);
```

```
     1 row inserted.

     SQL> --insert into ship_items
     SQL> insert into ship_items values(1,1,100780781,1);

     1 row inserted.

     SQL> insert into ship_items values(2,1,100780782,3);

     1 row inserted.

     SQL> insert into ship_items values(3,1,100780783,4);

     1 row inserted.

     SQL> update invoice_status set pay_status='Paid' where
emarket_invoice_id='100780781' or emarket_invoice_id='100780782' or
emarket_invoice_id='100780783';

     3 rows updated.

     SQL> select * from invoice_audit;

     no rows selected


     SQL> select * from payment_audit;

     INVOICE_ID ADD_DATE       AMOUNT
     ---------- ---------- ----------
      100780782 12/05/2015       3.26
      100780783 12/05/2015     326.99

     SQL> select * from invoice_type;

     INVOICE_TYPE_ID DESCRIPTION
     --------------- --------------------
                   1 Invoice
```

```
SQL> select * from tax_type;

TAX_TYPE_ID DESCRIPTION
----------- ----------
          1 Sales
          2 Use

SQL> select * from discount;

DISCOUNT_ID TERMS                PERCENTAGE  TIMEFRAME
----------- -------------------- ---------- ----------
          1 0% 30, Net 30                 0         30
          2 0% 10, Net 10                 0         10

SQL> select * from invoice;

EMARKET_INV  PO_NUMBER INVOICE_DATE DUE_DATE   TAX_TYPE_ID DISCOUNT_ID
----------- ---------- ------------ ---------- ----------- -----------
INVOICE_TYPE_ID
---------------
  100779134   10356325 10/08/2015   11/07/2015           1           2
              1

  100780781   10353635 10/07/2015   11/06/2015           1           1
              1

  100780782   10353636 12/07/2015   01/06/2016           1           1
              1


EMARKET_INV  PO_NUMBER INVOICE_DATE DUE_DATE   TAX_TYPE_ID DISCOUNT_ID
----------- ---------- ------------ ---------- ----------- -----------
INVOICE_TYPE_ID
---------------
  100780783   10353636 12/07/2015   01/06/2016           1           1
              1
```

```
SQL> select * from invoice_status;

STATUS_DATE PAY_STATUS           EMARKET_INV
----------- -------------------- -----------
10/07/2015  Paid                   100780781
10/08/2015  Payable                100779134
12/07/2015  Paid                   100780782
12/07/2015  Paid                   100780783

SQL> select * from payment_type;

PAYMENT_TYPE_ID DESCRIPTION
--------------- --------------------
              1 Unknown
              2 Employee Card
              3 Procurement Card

SQL> select * from payment;

PAYMENT_ID PAYMENT_TYPE_ID PAY_DATE   EMARKET_INV
---------- --------------- ---------- -----------
                            AMOUNT_PAID
---------------------------------------
         1               2 12/05/2015   100780782
                                   3.26

         2               2 12/05/2015   100780783
                                 326.99


SQL> select * from ship_items;

SHIP_ITEMS_ID   QUANTITY EMARKET_INV PO_ITEM_ID
------------- ---------- ----------- ----------
            1          1   100780781          1
            2          1   100780782          3
            3          1   100780783          4
```

```
SQL> select * from payable_status;

PAYABLE_STATUS_ID STATUS_DATE  PO_NUMBER RECEIVING_STATUS
----------------- ----------- ---------- ------------------------------
SUPPLIER_STATUS                INVOICING
------------------------------ ------------------------------
MATCHING                       PO_ITEM_ID
------------------------------ ----------
                2 08/30/2015    10356325 None
Sent to Supplier               Fully Invoiced
Not Matched                             2

                1 09/30/2015    10353635 None
Sent to Supplier               Fully Invoiced
Fully Matched                           1

PAYABLE_STATUS_ID STATUS_DATE  PO_NUMBER RECEIVING_STATUS
----------------- ----------- ---------- ------------------------------
SUPPLIER_STATUS                INVOICING
------------------------------ ------------------------------
MATCHING                       PO_ITEM_ID
------------------------------ ----------
                3 12/01/2015    10353636 None
Sent to Supplier               Fully Invoiced
Not Matched                             3

                4 12/01/2015    10353636 None
Sent to Supplier               Fully Invoiced
Not Matched                             4

SQL> select * from po_items;

PO_ITEM_ID  PO_NUMBER   QUANTITY    TAXABLE
---------- ---------- ---------- ----------
                      PAYABLE_STATUS_ID
---------------------------------------
         2   10356325          2          1
```

```
                                      2


         1   10353635          1          1
                                      1


         3   10353636          1          1
                                      3



PO_ITEM_ID  PO_NUMBER   QUANTITY    TAXABLE
---------- ---------- ---------- ----------
                      PAYABLE_STATUS_ID
---------------------------------------
         4   10353636          1          1
                                      4



SQL> select * from po_ap_status;

                        PO_AP_STATUS_ID  PO_NUMBER
--------------------------------------- ----------
                           AP_STATUS_ID
---------------------------------------
                                      1   10356325
                                      3

                                      2   10353635
                                      1

                                      3   10353636
                                      3



SQL> select * from ap_status;

                           AP_STATUS_ID DESCRIPTION
--------------------------------------- ------------------------------
                                      1 Soft Closed
                                      2 Hard Closed
                                      3 Open
                                      4 Pending

SQL> select * from object_code;

                            OBJECT_CODE DESCRIPTION
--------------------------------------- ------------------------------
                                  15102 General/Project Supplies
                                  15307 Office Supplies

SQL> select * from po;

 PO_NUMBER PO_DATE                                OBJECT_CODE PURCHASER_I
---------- ---------- --------------------------------------- -----------
RECIPIENT_I                          PURCHASE_REQ_ID
----------- ---------------------------------------
                        PO_AP_STATUS_ID
---------------------------------------
  10356325 10/07/2015                                   15307  1234567890
 4567890125                                 66710697
                                      1

  10353635 09/30/2015                                   15102  1234567890
 4567890125                                 67052957
                                      2

 PO_NUMBER PO_DATE                                OBJECT_CODE PURCHASER_I
---------- ---------- --------------------------------------- -----------
RECIPIENT_I                          PURCHASE_REQ_ID
----------- ---------------------------------------
                        PO_AP_STATUS_ID
---------------------------------------
  10353636 12/01/2015                                   15102  2345678901
 4567890124                                 67052958
                                      3
```

```
SQL> select * from purchase_requisition;

                     REQUISITION_NUMBER
---------------------------------------
PURPOSE
------------------------------------------------------------
                               66710697
New Laptop for Tele-Dentistry Clinic per Linda Brookman

                               67052957
Folders for IT Dept.

                               67052958
Dental Supplies


                     REQUISITION_NUMBER
---------------------------------------
PURPOSE
------------------------------------------------------------
                               67052959
New TV for Bossman

                               67052960
Sour Gummy Worms


SQL> select * from purchaser;

                            EMPLOYEE_ID DESCRIPTION
--------------------------------------- ------------------------------
                             3456789012 Jane Oh's Account
                             4567890123 Wilson Lin's Account
                             2345678901 1151 W. 36th Pl. Rm. B
                             1234567890 925 West 34th St.

SQL> select * from recipient;
```

```
                            EMPLOYEE_ID DELIVERY_ADDRESS
--------------------------------------- ------------------------------
                             4567890124 Cal Building
                             4567890125 Dental School

SQL> select * from employee;

EMPLOYEE_ID FIRST_NAME LAST_NAME       PHONE EMAIL
----------- ---------- ---------- ---------- ------------------------------
ADDRESS                          ZIP_CODE     ORG_ID
------------------------------ ---------- ----------
 1234567890 Brandon    Crabtree      1.2E+10 bcrabtre@usc.edu
925 West 34th St.                   90089 2060603000

 2345678901 Huy        Ngo           1.7E+10 huyngo@usc.edu
1151 W. 36th Pl.                    90007 2123109112

 3456789012 Jane       Oh            1.7E+10 janeoh@usc.edu
Somewhere in Torrance               90504 2123109112


EMPLOYEE_ID FIRST_NAME LAST_NAME       PHONE EMAIL
----------- ---------- ---------- ---------- ------------------------------
ADDRESS                          ZIP_CODE     ORG_ID
------------------------------ ---------- ----------
 4567890123 Wilson     Lin           1.7E+10 wilsonlin@usc.edu
Cardinal Gardens                    91006 2123109112

 4567890124 Douglas    Shook         1.2E+10 shook@usc.edu
Cal Building                        90007 2060603000

 4567890125 Vicky      Dy            1.2E+10 dy@usc.edu
Dental School                       90007 2060603000


 6 rows selected

SQL> select * from department;
```

```
    ORG_ID DESCRIPTION                    ACCOUNT_NUMBE
---------- ------------------------------ -------------
2060603000 Information Technology            1213050004
2123109112 Meat Selection                    1234567890

SQL> select * from accounts;

ACCOUNT_NUMBE ADDRESS                               ZIP
------------- ------------------------------ ----------
   1213050004 PO Box 77967                        90007
   1234567890 PO Box 17380                        33781

SQL> select * from vendor;

                        SUPPLIER_NUMBER DESCRIPTION
--------------------------------------- ------------------------------
       FAX      PHONE ADDRESS                          ZIP_CODE
---------- ---------- ------------------------------ ----------
                        VENDOR_EMPLOYEE
---------------------------------------
                              603046210 Dell Computer Corp.
1512837544    1.8E+10 1 Dell Way                          78682
                                      1

                              153531108 Office Depot, Inc.
              1.8E+10 6600 N Military Trail               33496
                                      2

                        SUPPLIER_NUMBER DESCRIPTION
--------------------------------------- ------------------------------
       FAX      PHONE ADDRESS                          ZIP_CODE
---------- ---------- ------------------------------ ----------
                        VENDOR_EMPLOYEE
---------------------------------------
                              111111111 Data Dentist Devices and Stuff
              1.7E+10 123 Potato Drive                    33781
                                      3
```

```
SQL> select * from product;

SUPPLIER_PART_AUX_ID MANU_PAR CATALOG_NUMBER
-------------------- -------- --------------------
                         COMMODITY_CODE
---------------------------------------
DESCRIPTION
--------------------------------------------------------------------------------
UNIT_PRICE SI
---------- --
1017724224640\1      210-ABDE 210-ABDE
                                     14
Dell Latitude E7450
   1113.16 EA


SUPPLIER_PART_AUX_ID MANU_PAR CATALOG_NUMBER
-------------------- -------- --------------------
                         COMMODITY_CODE
---------------------------------------
DESCRIPTION
--------------------------------------------------------------------------------
UNIT_PRICE SI
---------- --
544387               OD24030R 544387
                                     18
Office Depot(R) Brand Pressboard Classification with Folders with Fasteners, Let
ter Size, 100 Recycled, Light Blue, Pack of 10
     33.94 BX

SUPPLIER_PART_AUX_ID MANU_PAR CATALOG_NUMBER
-------------------- -------- --------------------
                         COMMODITY_CODE
---------------------------------------
DESCRIPTION
--------------------------------------------------------------------------------
```

```
UNIT_PRICE SI
---------- --
132842               001-DENT 001-DENT
                                     16
Dentures
     99.99 EA


SUPPLIER_PART_AUX_ID MANU_PAR CATALOG_NUMBER
-------------------- -------- --------------------
                         COMMODITY_CODE
---------------------------------------
DESCRIPTION
--------------------------------------------------------------------------------
UNIT_PRICE SI
---------- --
123456               002-CROW 002-CROW
                                     16
Crowns
    299.99 EA


SUPPLIER_PART_AUX_ID MANU_PAR CATALOG_NUMBER
-------------------- -------- --------------------
                         COMMODITY_CODE
---------------------------------------
DESCRIPTION
--------------------------------------------------------------------------------
UNIT_PRICE SI
---------- --
234567               003-BIBS 003-BIBS
                                     16
Bibs
      2.99 EA


SQL> select * from commodity;
```

```
                         COMMODITY_CODE
---------------------------------------
DESCRIPTION
--------------------------------------------------
                                     18
Office Supplies and Equipment

                                     14
IT Hardware and Maintenance

                                     16
Dental Supplies


SQL> select * from product_size;

SUPPLIER_PART_AUX_ID MANU_PAR SI
-------------------- -------- --
1017724224640\1      210-ABDE EA
123456               002-CROW EA
132842               001-DENT EA
234567               003-BIBS EA
544387               OD24030R BX

SQL> select * from size_codes;

SI
--
BX
EA

SQL> select * from vendor_product;

                      VENDOR_PRODUCT_ID                         SUPPLIER_NUMBER
--------------------------------------- ---------------------------------------
SUPPLIER_PART_AUX_ID MANU_PAR      PRICE
-------------------- -------- ----------
                                      1                               603046210
```

```
1017724224640\1      210-ABDE    1113.16

                                      2                               111111111
123456               002-CROW     299.99

                                      3                               153531108
544387               OD24030R      33.94


                      VENDOR_PRODUCT_ID                         SUPPLIER_NUMBER
--------------------------------------- ---------------------------------------
SUPPLIER_PART_AUX_ID MANU_PAR      PRICE
-------------------- -------- ----------
                                      4                               111111111
234567               003-BIBS       2.99

                                      5                               111111111
132842               001-DENT      99.99


SQL> select * from vendor_emp;

                          VENDOR_EMP_ID                         SUPPLIER_NUMBER
--------------------------------------- ---------------------------------------
                        FROM_INVOICE_ID FIRST_NAME      LAST_NAME
--------------------------------------- --------------- ---------------
                                      1                               603046210
                             7426166805 Jeff            Dahlman

                                      2                               153531108
                              798851330 Huy             Ngo

                                      3                               111111111
                              472942793 Douglas         Shook


SQL> select * from zip;
```

```
       ZIP ST CITY
---------- -- ------------------------------
     91110 CA Pasadena
     90007 CA Los Angeles
     90089 CA Los Angeles
     78682 TX Round Rock
     33781 FL Pinellas Park
     33496 FL Boca Raton
     90504 CA Torrance
     91006 CA Arcadia

 8 rows selected

SQL> alter table product enable constraint product_size_null;

Table PRODUCT altered.

SQL> alter table vendor enable constraint vendor_employee_null;

Table VENDOR altered.

SQL> alter table po_items enable constraint po_items_payable_null;

Table PO_ITEMS altered.

SQL> alter table po enable constraint po_ap_status_null;

Table PO altered.

SQL> spool off
```

Queries (Output)

```
     SQL> /* show all vendors and the names of the employees that are located in FL
*/
     SQL> select distinct description, vendor_emp.first_name, vendor_emp.last_name
     from zip, vendor, vendor_emp
     where zip.state = 'FL' and zip.zip=vendor.ZIP_CODE and
vendor_emp.vendor_emp_id=vendor.vendor_employee;

     DESCRIPTION                    FIRST_NAME      LAST_NAME
     ------------------------------ --------------- ---------------
     Office Depot, Inc.             Huy             Ngo
     Data Dentist Devices and Stuff Douglas         Shook

     SQL> /* show the amount paid and pay statuses from invoices that occurred in
December 2015 */
     SQL> select PAYMENT.EMARKET_INVOICE_ID, sum(amount_paid) as total, pay_status
     from po,PAYMENT,invoice_status,invoice
     where po.po_date >= '12/01/2015' and po.po_number=invoice.po_number and
pay_status ='Paid' and
     invoice.emarket_invoice_id=PAYMENT.EMARKET_INVOICE_ID and
PAYMENT.EMARKET_INVOICE_ID=invoice_status.emarket_invoice_id
     group by payment.emarket_invoice_id, pay_status;

     EMARKET_INV      TOTAL PAY_STATUS
     ----------- ---------- --------------------
       100780782       3.26 Paid
       100780783     326.99 Paid

     SQL> /* show the purchase requisitions that did not get fulfilled */
     SQL> select distinct requisition_number, purpose
     from purchase_requisition, po
     minus
     select distinct po.PURCHASE_REQ_ID, purpose
     from PURCHASE_REQUISITION,PO
     where po.purchase_req_id=purchase_requisition.requisition_number;

                         REQUISITION_NUMBER
     ---------------------------------------
```

```
PURPOSE
------------------------------------------------------------
                               67052959
New TV for Bossman

                               67052960
Sour Gummy Worms



SQL> /* show the invoices paid by 'Employee Card' and is over $100 */
SQL> select emarket_invoice_id, amount_paid
from payment, payment_type
where amount_paid>100 and payment_type.description='Employee Card';

EMARKET_INV                             AMOUNT_PAID
----------- ---------------------------------------
  100780783                                  326.99

SQL> /* show the employee responsible for the most POs past June 2015*/
SQL> with po_count as(select count(po_number)as max_count,purchaser_id from po
where po_date>='06/01/2015' group by PURCHASER_ID)
select po_count.purchaser_id, po_count.max_count
from po_count
where rownum<=1
group by po_count.purchaser_id, po_count.max_count
order by po_count.max_count desc;

PURCHASER_I  MAX_COUNT
----------- ----------
 1234567890          2

SQL> /* show the recipient and PO number for the POs for which the purchaser was
either Huy Ngo or Jane Oh */
SQL> with desired_names as (select distinct employee.employee_id from purchaser,
employee where first_name='Jane' or first_name='Huy'
and purchaser.employee_id=employee.employee_id)
select distinct recipient_id, po.PO_NUMBER
from po, recipient,purchaser,employee,desired_names
```

```
     where recipient.employee_id = employee.employee_id and
desired_names.employee_id=po.purchaser_id;

     RECIPIENT_I  PO_NUMBER
     ----------- ----------
      4567890124   10353636

     SQL> spool off
```

Triggers (Output)

```
Trigger INVOICE_DELETE_AUDIT compiled


Trigger PAYMENT_INSERT_AUDIT compiled
```

Physical Design Summary

The implementation of indexes was used primarily on attributes that we felt, would be queried frequently. For example, this would mean first names, last names, dates, emails, and so on. This would be helpful, as seen in our Transaction Analysis Forms, for the creation of a purchase order and the management of cash accounts. The creation of a purchase order requires reading from tables such as Recipient, Purchaser, and Vendor; indexing these tables would be favorable because they have a low create, update, delete to create, read, update, delete ratio even though they do not have a high volume. The management of cash accounts involves extensive reading from Payment and Invoice to calculate totals for a fiscal period, so indexing Payment dates and Invoice dates would be wise for quick access. It was also part of our strategy to establish 1:n relationships among tables and composite primary keys whenever appropriate so that we could have foreign keys indexed for free.

Our triggers worked to establish some obvious business rules that is required for the dental school for carry out its auditing processes. One trigger, “invoice_audit” to archived deleted invoices that can be referred to later on. Another trigger tracks the “payment_audit” to establish a running total of payments, their dates, and what invoices to which they are connected so that a spending total can be calculated based on this for any range of dates. As an additional benefit, this allows for volume control of both the Invoice and Payment tables so that query speeds can increase; querying from our trigger tables will be less frequent than the regular Invoice and Payment tables. In all, amortized runtime will be much lower because of these triggers. There were a few ways to enforce business rules without the use of triggers by using check statements to enforce, for example, insert quantities and prices to be greater than 0. Our user views had a way to track statuses of purchase orders and invoices, so we used check statements in order to enforce consistent insert formats for statuses.

As for our clusters, we chose to cluster the purchase order number (po_number) because so many other tables require it as a foreign key in order to be accurate in querying information regarding an invoice (as it is connected to a purchase order), accounts payable status, and so on. The strategy for clusters was more conservative considering the fact that clusters can optimize joins from purchase order to another table, but can slow down other accesses. This way, we concluded would allow for one single array of access for po_numbers rather than having it scattered between tables containing the same po_numbers repeatedly.

F. Summary and Conclusion

Overall, our project was a successful and meaningful experience. We learned a lot through collaborative discussion and problem solving and implementing the physical design. We agreed that this was one of the more complex projects we've encountered while at USC. Being able to integrate all our work together was rewarding and very valuable.

Difficulties

We initially encountered issues with our client contact. They did not fully understand the concept of a database, and so they were unsure how they could help us with the project. To remediate this, we made a presentation to them about what databases are, how they are different from spreadsheets, and the benefits of databases. From there, our contact and our team were able to better scope the project, and consequently received the complex user views that we used for the project.

Understanding the complex user views also presented was also challenging. As the user views spanned multiple pages, it was initially difficult to track all the different attributes and business rules being put in place by the system. Some attributes also confused us, such as commodity codes, acronyms like UOM (unit of material), and accounting codes.

Enforcing the complex business rules in our database was also a challenge. As we modified our SQL code to better represent the constraints, we had ripple effects occurring in our 3rd normal forms, domain definitions, and CDM.

Problem Solving

Our team had great synergy. Whenever we encountered problems, we shared our questions with the team so we could collaboratively figure out solutions. This helped us grow as a group for we bounced our different perspectives with each other. To enforce business rules, such as the implicit differentiation among employees, we had subclass Employee into Recipient and Purchaser, who are responsible for different parts of the business process including initial creation, receiving, and payment. We faced a small consideration when it came to the standard relationship between purchase requisition and purchase order. We decided to enforce a business rule, to improve model integrity, that required a purchase order to be created only after source purchase requisition(s) are in place. It was a notable accomplishment for us in modeling the business rule that a purchase order can result in multiple invoices and shipments. This was done by creating tables such as ship_items to be connected to invoice so that staggered shipments can still be invoiced properly from the same purchase order.

Likes/Dislikes about the Database

We like that our database represents a common business situation, but in a complex format. We liked that we had to dig deep and represent many of the business rules and data that was being stored in every transactions. Looking at our final product, we do not like that our database needs another semester of work to work out more details so that we can enforce more business rules and have better functionality for common queries.

What you learned

As engineering students, Huy and Wilson learned a lot about the accounting process, including the accounting codes and ways to identify products. We all also learned about a collaborative flow chart website called draw.io, which proved to be immensely helpful in creating our EDM, ERD, CDM, and Composite Usage Map.

Our main take-away from the project was being able to appreciate the complexity of databases. We previously thought it would simply be an organized way to store information. However, through the class and the project, we understood how complex relationships are and how one needs to creatively think to be true data mavens.