Oracle Applications Data Structure

by Jenny Chan, Red Rock Consulting

Introduction

Oracle Applications has a very flexible architecture containing valuable information but with flexibility comes complexity in the underlying data structure. Understanding the basic Oracle Applications structure is critical in retrieving important and relevant content. This information is an asset and when used appropriately can be used as a competitive advantage.

This article provides a high level overview of the Oracle Applications data structure and some business examples. However, before we get understand the data structure there is some technical terminology that needs explanation.

The business examples will be illustrated in Oracle SQL*Plus but you can use the same table joins in your preferred reporting writing tool such as Oracle Querybuilder and Oracle Discoverer.

Please refer to the Oracle Technical Reference manuals for more detail on the entity data models, table descriptions and functional decomposition.

Also note that the following information is based on Oracle Applications Release 11.03 data structures.

Definitions

Table

The basic entity of storage in a relational database management system.

Consist of one or more units of information (records) each of which contains a number of data elements or fields (or columns).

Record/Row

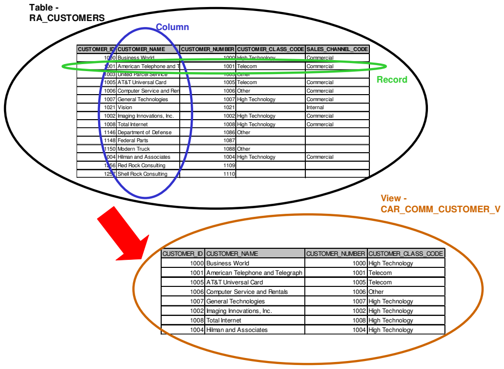
One set of related information contained in a table.

Column/Field

A subdivision of a table, with a column name and a specific datatype.

View

A datasource made up of columns from one or more database tables combined into one logical table or object.



CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_NUMBER	CUSTOMER_CLASS_CODE	SALES_CHANNEL_CODE
1000	Business World	1000	High Technology	Commercial
1001	American Telephone and T	1001	Telecom	Commercial
1003	United Parcel Service	1003	Other	
1005	AT&T Universal Card	1005	Telecom	Commercial
1006	Computer Service and Ren	1006	Other	Commercial
1007	General Technologies	1007	High Technology	Commercial
1021	Vision	1021		Internal
1002	Imaging Innovations, Inc.	1002	High Technology	Commercial
1008	Total Internet	1008	High Technology	Commercial
1146	Department of Defense	1086	Other	
1148	Federal Parts	1087		
1150	Modern Truck	1088	Other	
1004	Hilman and Associates	1004	High Technology	Commercial
1256	Red Rock Consulting	1109		
1257	Shell Rock Consulting	1110		

CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_NUMBER	CUSTOMER_CLASS_CODE
1000	Business World	1000	High Technology
1001	American Telephone and Telegraph	1001	Telecom
1005	AT&T Universal Card	1005	Telecom
1006	Computer Service and Rentals	1006	Other
1007	General Technologies	1007	High Technology
1002	Imaging Innovations, Inc.	1002	High Technology
1008	Total Internet	1008	High Technology
1004	Hilman and Associates	1004	High Technology

Synonym

Another name assigned to a table for easy identification and used for data classification between Oracle Application modules.

Primary Key

The column or columns that uniquely identify each row of a table.

Foreign Key

One of more columns in one table whose values refer to the primary key values in another table.

Query

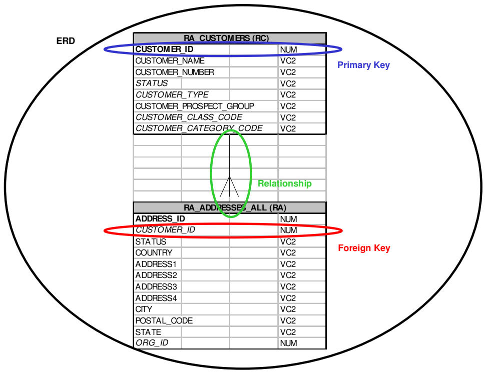
Series of command to instruct the database to retrieve the data you have specified.

Relationship

A connection between two or more tables.

Entity Relationship Diagram (ERD)

Diagram that display the relationships between tables.



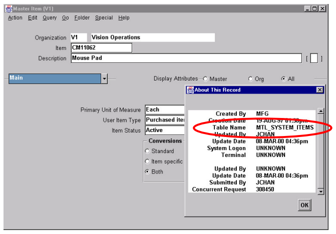
RA_CUSTOMERS (RC)	
CUSTOMER_ID	NUM
CUSTOMER_NAME	VC2
CUSTOMER_NUMBER	VC2
STATUS	VC2
CUSTOMER_TYPE	VC2
CUSTOMER_PROSPECT_GROUP	VC2
CUSTOMER_CLASS_CODE	VC2
CUSTOMER_CATEGORY_CODE	VC2

RA_ADDRESSES_ALL (RA)	
ADDRESS_ID	NUM
CUSTOMER_ID	NUM
STATUS	VC2
COUNTRY	VC2
ADDRESS1	VC2
ADDRESS2	VC2
ADDRESS3	VC2
ADDRESS4	VC2
CITY	VC2
POSTAL_CODE	VC2
STATE	VC2
ORG_ID	NUM

Tips

- There is a screen in Oracle Applications where you would like to report the information.

Click on Help > About This Record from the Menu.



Organization V1 Vision Operations
Item CM11062
Description Mouse Pad

Display Attributes ○ Master ○ Org ● All

Primary Unit of Measure Each
User Item Type Purchased ite
Item Status Active

Conversions
○ Standard
○ Item specific
● Both

Created By	MFG
Creation Date	19-AUG-97 01:38pm
Table Name	MTL_SYSTEM_ITEMS
Updated By	JCHAN
Update Date	08-MAR-00 04:36pm
System Logon	UNKNOWN
Terminal	UNKNOWN
Updated By	UNKNOWN
Update Date	08-MAR-00 04:36pm
Submitted By	JCHAN
Concurrent Request	308450

- There is a table name (e.g. ITEM) you would like to use but unsure of the exact name.

Login to SQL*Plus and type

```
SELECT table_name
FROM all_tables
WHERE table_name LIKE '%ITEM%'
```

- There is a column (e.g. CUSTOMER_NAME) that you would like to use but unsure which table the column belongs to.

Login to SQL*Plus and type

```
SELECT table_name, column_name
FROM all_tab_columns
WHERE column_name = 'CUSTOMER_NAME'
```

- The columns org_id and organization_id are different. Organisation_id refers to the inventory organisations in Oracle Inventory and org_id refers to the operating unit.

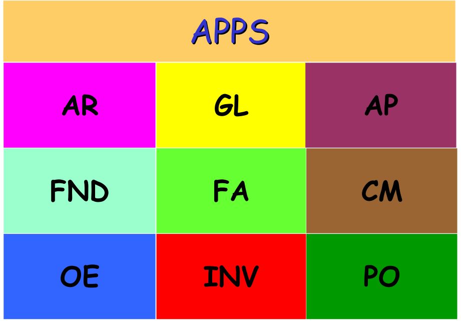
- Table names that ends with _all has the column org_id included. The org_id signifies the operating unit. Therefore, if you have multi-organisation functionality, you will need to use the _all tables. Note that not all tables will have _all suffix.

- The majority of the entities describes in the following are tables but you may find that there is a view that combines all your requirements. These views will be based on non _all tables. You need to type the following in SQL*Plus to see the data in these views.

  exec dbms_application_info.set_client_info(org_id)

  Org_id is the operating unit in hr_organization_units.

- The following illustrates a simplified view of the Oracle Applications data schema. Each data schema is mapped to an application module. The apps schema has a number of synonyms and views of all the Oracle Application modules. The fnd has all the schema all the Oracle Application foundation information such as user profiles, responsibility and value sets.
  You need to login to the appropriate schema when using query tools on a table. Therefore, consult your Database Administrator regarding security to the schema/schemas you need access.



APPS		
AR	GL	AP
FND	FA	CM
OE	INV	PO

Oracle Applications Data Structure

Oracle General Ledger

Gl_code_combinations

✚ Setup > Accounts > Combinations

This table stores the valid account combinations.

The value in your chart of account segments are stored in the columns segment1 to segment30 depending on your application configuration.

For example, say your chart of accounts is
Company – Cost Centre – Account
then segment1 = company, segment 2 = cost centre and segment3 = account.
However, this sequencing of segments is not guaranteed therefore, its best to check your configuration.

Another important column is the account_type which signifies your account is an Asset, Liability, Revenue, Expense or Owners Equity account.

Gl_je_batches

✚ Journals > Enter

This table stores the journal entry batches. Journal entries are batched in General Ledger.

Some columns of interest includes :

- Name
- Set_of_books_id (when you have more than one set of book, you'll also need to link to gl_sets_of_books)
- Status
- Default_period_name
- Posted_date
- Posting_run_id

Gl_je_headers

✚ Journals > Enter

This table stores the journal entry headers. There is always two journal lines for each journal header.

Some columns of interest includes :

- Je_category

- Period_name
- Set_of_books_id (when you have more than one set of book, you'll also need to link to gl_sets_of_books)
- Posted_flag
- Je_source
- Name
- Status

Gl_je_lines

✚ Journals > Enter

This table stores the journal entry lines.

The entered_dr and entered_cr stores the amount in the entered currency whereas the accounted_dr and accounted_cr stores the amount in the functional currency.

Other columns of interest includes :

- Set_of_books_id (when you have more than one set of book, you'll also need to link to gl_sets_of_books)
- Period_name
- Status
- Description
- Reference_1..reference10 (these columns links back to your Subledgers)

  For example, for Purchasing transactions

  Reference_1 = 'PO'
  Reference_2 = po_headers_all.po_header_id
  Reference_3 = po_distributions_all.po_distribution_id
  Reference_4 = po_headers_all.segment? (this is the purchase order number

Oracle Payables

Ap_invoices_all

✚ Invoices > Entry > Invoices

This table stores all the invoices you enter. For an invoice to be approved, the total invoice amount must be stored in ap_invoice_distributions_all and ap_payment_schedules_all.

Some columns of interest includes :

- Invoice_num
- Invoice_date
- Amount_paid
- Invoice_currency_code
- Invoice_type_lookup_code
- Payment_status_flag

Ap_invoice_distributions_all

✚ Invoices > Entry > Invoices

This table stores the accounting information for the invoice you have entered. There is one row for each invoice disribution, that is this table corresponds to the Distributions window.

Some columns of interest includes :

- Line_type_lookup_code
- Dist_code_combination_id (credit entry)
- Accts_pay_code_combination_id (debit_entry)
- Base_amount (in functional currency)

Ap_checks_all

✚ Payments > Entry > Payments

This table stores payments to suppliers.

Some columns of interest includes :

- Amount (in functional currency)
- Check_date
- Bank_account_name
- Check_number
- Payment_method_lookup_code
- Payment_type_flag

Ap_invoice_payments_all

✚ Payments > Entry > Payments

This table stores invoice payments to suppliers. This table is updated when you confirm an automatic payment batch, enter a manual payment or process a Quick Payment. Void payments are represented as a negative of the original payment line.

Some columns of interest includes :

- Accounting_date
- Period_name
- Amount
- Payment_num

Ap_payment_distributions_all

✚ Payments > Entry > Payments

This table stores accounting information for payments. There is at least one CASH payment distribution for each invoice payment. Additional rows may include DISCOUNT, GAIN and LOSS distributions where appropriate.

Some columns of interest includes :

- Line_type_lookup_code (CASH/DISCOUNT/GAIN/LOSS)
- Base_amount

Oracle Purchasing

Po_vendors

✚ Supply Base > Suppliers

This table stores supplier information.

Some columns of interest includes :

- Segment1 (supplier number)
- Vendor_name
- Terms_id
- Vendor_type
- Ship_to_location (link to hr_locations for location information)
- Bill_to_location (link to hr_locations for location information)

Po_vendor_sites_all

✚ Supply Base > Suppliers

This table stores supplier sites information.

Some columns of interest includes :

- Pay_site_flag
- Purchasing_site_flag
- Address_line1 to address_line3
- City
- State
- Area_code
- Zip

Po_headers_all

✚ Purchase Orders > Purchase Orders

This table stores the seven types of purchasing documents such as Purchase Order and Blanket Agreement.

Segment1 is the document number (i.e. purchase order number)

Some columns of interest includes :

- Agent_id (link to per_people_f for the buyer)
- Type_lookup_code

Po_lines_all

✚ Purchase Orders > Purchase Orders

This table stores purchasing document lines.

Some columns of interest includes :

- Line_num
- Item_description
- Unit_price
- Unit_meas_lookup_code (unit of measure)
- Quantity
- Item_id (link to mtl_system_items for the item number)
- Category_id (link to mtl_categories for the category name)

Po_line_locations_all

✚ Purchase Orders > Purchase Orders

This table stores purchase order shipment schedules and blanket agreement price breaks. A purchase order is closed when QUANTITY is equal to QUANTITY_RECEIVED.

Some columns of interest includes :

- Quantity
- Quantity_accepted
- Quantity_received
- Quantity_cancelled
- Need_by_date
- Ship_to_organization_id (link to org_organization_definitions for the organization code)

Po_distributions_all

✚ Purchase Orders > Purchase Orders

This table stores the accounting information on a purchase order shipment. This table is used for Standard and Planned Purchase Orders and Planned and Blanket Purchase Order Release.

Some columns of interest includes :

- Quantity_ordered
- Quantity_billed
- Amount_billed
- Quantity_delivered
- Quantity_cancelled
- Destination_organization_id (link to org_organization_definitions for the organization code)
- Destination_subinventory

Rcv_shipment_headers

✚ Receiving > Receipts

This table stores the receiving information. The three receipt sources are Supplier, Inventory and Internal Order. There is one receipt header per receipt source.

Some columns of interest includes :

- Receipt_num
- Shipment_num
- Receipt_source_code
- Shipped_date
- Ship_to_org_id

Rcv_shipment_lines

✚ Receiving > Receipts

This table stores information about items that have been shipped and/or received from a receipt source.

Some columns of interest includes :

- Line_num
- Quantity_shipped
- Unit_of_measure
- Item_id (link to mtl_system_items for item number)
- To_organization_id (link to org_organization_definitions for organization code)
- To_subinventory
- Shipment_line_status_code (EXPECTED, FULLY RECEIVED, PARTIALLY RECEIVED)
- Quantity_received
- Quantity_shipped

Oracle Inventory

Org_organization_definitions

✚ Setup > Organizations > Parameters

This view contains basic information on all inventory organisations.

Some columns of interest includes :

- Organization_code
- Organization_name
- Set_of_books_id (when you have more than one set of book, you'll also need to link to gl_sets_of_books)
- Inventory_enabled_flag

Mtl_secondary_inventories

✚ Setup > Organizations > Subinventories

This table stores all subinventory information for an inventory organisation.

Some columns of interest includes :

- Secondary_inventory_name
- Description

Mtl_material_transactions

✚ Transactions > Material Transactions (Inquiry)

This table stores all inventory transactions including cost updates.

Some columns of interest includes :

- Transaction_quantity
- Transaction_type_id
- Transaction_source_type_id
- Transaction_source_name

Mtl_transaction_accounts

✚ Transactions > Material Distributions (Inquiry)

This table stores the inventory accounting information. There are two rows in this table for each transaction in mtl_material_transactions.

Some columns of interest includes :

- Transaction_date
- Gl_batch_id
- Accounting_line_type
- Base_transaction_value

Mtl_system_items

✚ Items > Master Items or Items > Organization Items

This table stores the item definition. An item must exist in an inventory organisation.

Your item number is stored in the columns segment1 to segment20 depending on your application configuration. If you have configured your items to have to segments then you may be using segment1 and segment2

Some columns of interest includes :

- Segment1 to segment20
- Description
- Invetory_item_flag
- Purchasing_item_flag
- Inventory_asset_flag
- Stock_enabled_flag
- Invoiceable_item_flag
- Shippable_item_flag
- So_transaction_flag
- Mtl_transactions_enabled_flag
- Primary_unit_of_measure

Mtl_onhand_quantities

✚ On-hand, Availability > On-hand Quantities

This table stores quantity on hand in a location for each item.

Some columns of interest includes :

- Date_received
- Transaction_quantity
- Subinventory_code

Cst_item_costs

✚ Costs > Item Costs

This table stores the item cost information. Note that there can be multiple costs per item and the actual cost is where the cost type is Frozen.

Some columns of interest includes :

- Cost_type_id (link to cst_cost_types)
- Item_cost

Oracle Receivables

Ra_customers

✚ Customers > Standard

This table stores customer information.

Some columns of interest includes :

- Customer_name
- Customer_number
- Status
- Customer_prospect_code
- Customer_type
- Orig_system_reference (for imported customers from an external source)

Ra_addresses_all

✚ Customers > Standard

This table stores customer address information and your remit-to addresses.

Some columns of interest includes :

- Status
- Orig_system_reference (for imported customer addresses from an external source)
- Address1 to address4
- City
- State
- Postal_code

Ra_site_uses_all

✚ Customers > Standard

This table stores the customer's site and site purpose. You must have one row for each address. A customer must have one bill to address for Receivables. A customer must have one ship to address and one bill to address for Order Entry.

Some columns of interest includes :

- Site_use_code (BILL_TO, SHIP_TO, STMTS, DUN/LEGAL)
- Primary_flag
- Status
- Location

Ra_customer_trx_all

✚ Transactions > Transactions

This table stores invoice, debit memo, chargeback, commitment and credit memo header information.

Some columns of interest includes :

- Cust_trx_type_id (link to ra_cust_trx_types_all)
- Set_of_books_id (when you have more than one set of book, you'll also need to link to gl_sets_of_books)
- Terms_id (link to ra_terms)
- Trx_number (invoice number)
- Trx_date (invoice date)

Ra_customer_trx_lines_all

✚ Transactions > Transactions

This table stores the invoice, debit memo, chargeback, commitment and credit memo line information.

Some columns of interest includes :

- Line_number
- Description
- Quantity_ordered
- Quantity_credited
- Quantity_invoiced
- Unit_standard_price
- Unit_selling_price
- Line_type
- Extended_amount
- Revenue_amount

Ra_cust_trx_line_gl_dist_all

✚ Transactions > Transactions

This table stores the accounting information for revenue, unearned revenue, unbilled receivables, receivables, charges, freight and tax for each invoice or credit memo line.

Some columns of interest includes :

- Amount_gl_date
- Gl_posted_date
- Account_class (CHARGES/FREIGHT/TAX/REC/REV/UNBILL/UNEARN)
- Acctd_amount (functional currency)

Ar_cash_receipts

✚ Receipts > Receipts

This table stores the payment information.

Some columns of interest includes :

- Set_of_books_id (when you have more than one set of book, you'll also need to link to gl_sets_of_books)
- Status (APP, UNAPP, UNID, NSF, STOP, REV)
- Type (CASH, MISC)
- Receipt_number
- Amount
- Currency_code
- Pay_from_customer

- Receipt_date

Ar_receivable_applications

✚ Receipts > Receipts

This table stores accounting entries for cash and credit memo applications.

Some columns of interest includes :

- Amount_applied
- Line_applied
- Tax_applied
- Application_type
- Display
- Gl_date
- Set_of_books_id (when you have more than one set of book, you'll also need to link to gl_sets_of_books)

Ar_payment_schedules

✚ Transactions > Transactions and Receipts > Receipts

This table stores all transactions except adjustments and miscellaneous cash receipts. This table is updated when a transaction occurs against an invoice, debit memo, chargeback, credit memo, on-account credit, or receipt.

Some columns of interest includes :

- Amount_due_original
- Status
- Class (DEP, DM, PMT, GUAR, CM, CB, INV)
- Due_date
- Amount_due_remaining
- Invoice_currency_code
- Amount_applied
- Anmount_credited
- Amount_adjusted

Oracle Order Entry

So_headers_all

✚ Orders, Returns > Orders, Returns

This table stores the orders and returns information.

The s1 to s30 and s1_date to s30_date relates to the order cycle status.

Some columns of interest includes :

- Order_category (I, P. S, R, RMA)
- Order_number
- Purchase_order_num
- Original_system_source_code (for imported orders from an external source)
- Original_system_reference (for imported orders from an external source)
- Order_type_id (link to so_order_types_all for order type)
- Date_ordered
- S1 to s30
- S1_date to s30_date

So_lines_all

✚ Orders, Returns > Orders, Returns

This table stores the orders and returns line information.

The s1 to s30 and s1_date to s30_date relates to the order line cycle status.

Some columns of interest includes :

- Line_type_code (DETAIL, PARENT, REGULAR, RETURN)
- Ordered_quantity
- Cancelled_quantity
- Selling_price
- Price_list_id (links to so_price_lists for price list)
- Schedule_date
- Promise_date

So_line_details

✚ Orders, Returns > Orders, Returns

This table stores order scheduling information.

Some columns of interest includes :

- Released_flag
- Quantity
- Schedule_date
- Delivery (link to mtl_demand for reservation details)

So_picking_batches_all

✚ Shipping > Release Sales Orders or Shipping > Release Sales Orders, SRS

This table stores the batch of orders that have been pick released.

The header_count column indicates the number of picking headers are contained in a picking batch.

So_picking_headers_all

✚ Shipping > Release Sales Orders or Shipping > Release Sales Orders, SRS

This table stores the picking headers within a picking batch.

Some columns of interest includes :

- Status_code (BACKORDERED, BACKORDER RELEASE, CLOSED, OPEN, PENDING, IN PROGRESS)
- Pick_slip_number

- Picked_by (link to per_people_f for picked by user)
- Date_released
- Date_confirmed
- Date_shipped

So_picking_lines_all

✚ Shipping > Release Sales Orders or Shipping > Release Sales Orders, SRS

This table stores the picking lines for a picking header.

Some columns of interest includes :

- Requested_quantity
- shipped_quantity
- Date_requested
- Cancelled_quantity

So_picking_line_details

✚ Shipping > Release Sales Orders or Shipping > Release Sales Orders, SRS

This table stores the location for the picking lines that have been reserved.

Some columns of interest includes :

- Requested_quantity
- Serial_number

Wsh_departure

✚ Shipping > Departure Planning> Departure Planning or Shipping > Departure Planning > New Departures

This table stores departure information for departure planning.

Some columns of interest includes :

- Name
- Planned_departure_date
- Actual_depature_date

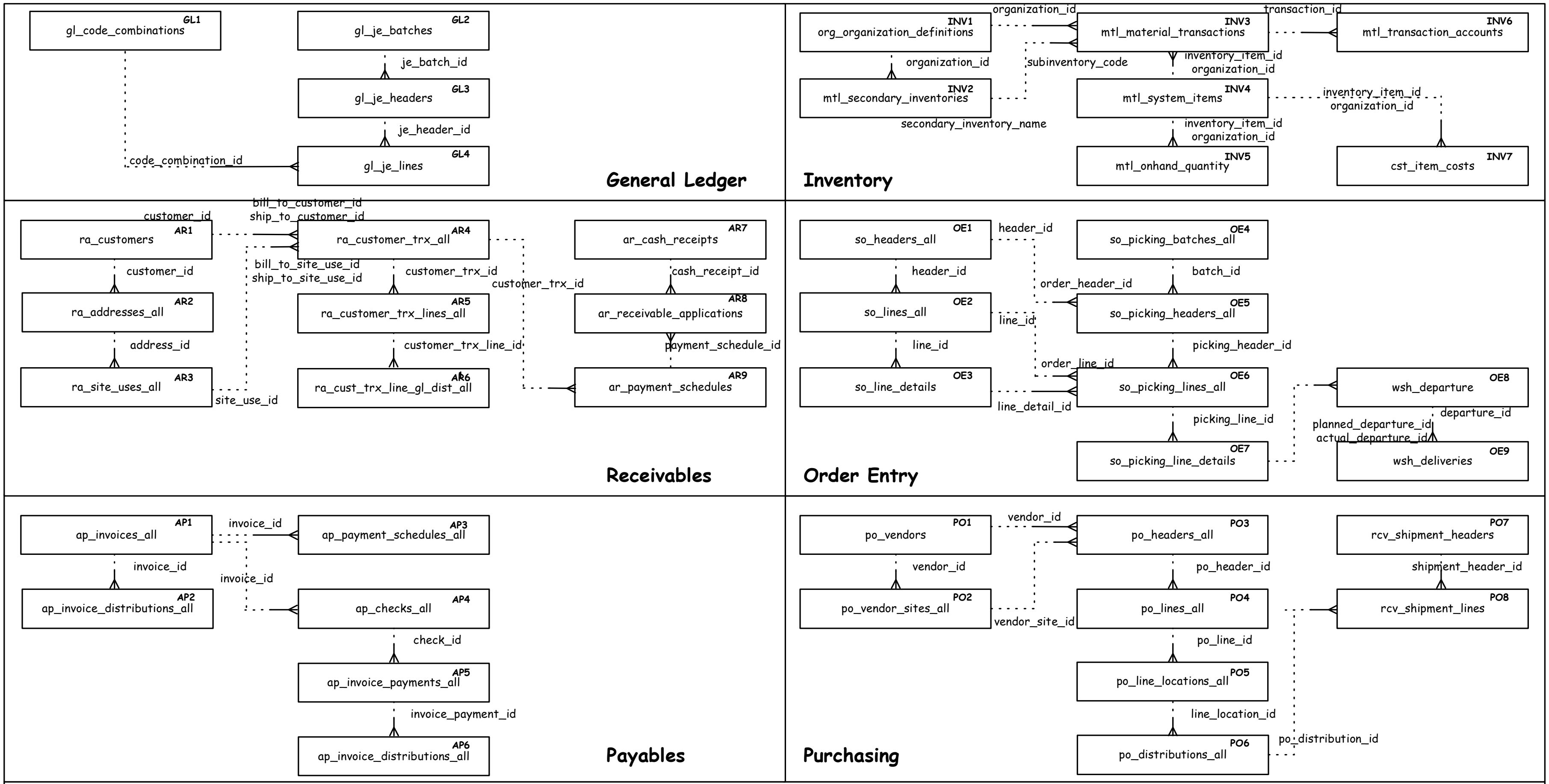
Wsh_delivery

✚ Shipping > Departure Planning> Departure Planning or Shipping > Departure Planning > New Deliveries

This table stores delivery information within a departure.

Some columns of interest includes :

- Name
- Planned_departure_date
- Actual_depature_date
- Waybill
- Date_closed



Cross Application Joins

GL1	code_combination_id	=	AR6, AR9 PO6	code_combination_id
GL1	code_combination_id	=	PO6	budget_account_id, accrual_account_id, variance_account_id
GL1	code_combination_id	=	INV6	reference_accounts
GL1	code_combination_id	=	AP2, AP6	accts_pay_code_combination_id
GL1	code_combination_id	=	AP6	dist_code_combination_id
INV3	source_line_id and source_code = 'ORDER ENTRY'	=	OE3	line_detail_id
INV3	picking_line_id and source_code = 'ORDER ENTRY'	=	OE6	picking_line_id
INV3	shipment_number and source_code = 'RCV'	=	PO7	shipment_num
INV3	transaction_source_id and source_code = 'RCV'	=	PO8	shipment_line_id
INV4	organization_id	=	OE2	organization_id
INV4	inventory_item_id	=	OE2, PO4, AR5	inventory_item_id

AP1	po_header_id	=	PO3	po_header_id
AP6	po_distribution_id	=	PO6	po_distribution_id
AP1	vendor_id	=	PO1	vendor_id
AP1	vendor_site_id	=	PO2	vendor_site_id
OE1	customer_id	=	AR1	customer_id
OE1	ship_to_site_use_id	=	AR2	site_use_id and site_use_code = 'SHIP_TO'
OE1	bill_to_site_use_id	=	AR2	site_ise_id and site_use_code = 'BILL_TO'
AR5	interface_line_attribute1 and interface_context = 'ORDER ENTRY'	=	OE1	order_number
AR5	interface_line_attribute6 and interface_context = 'ORDER ENTRY'	=	OE2	line_id
AR5	interface_line_attribute7 and interface_context = 'ORDER ENTRY'	=	OE6	picking_line_id

Business Examples

To illustrate how important knowing the Oracle Applications data structure, I have included some business examples.

Reconciliation

Are you having problem reconciling your General Ledger to your subledger ? This process can take a long time to resolve if you don't know where to look.

The following SQL*Plus example displays the sum of the Trade Receipts from your Receivables subledger for a given period name (i.e. 'Apr-00') by the General Ledger batch name. This example also assumes that you are entering receipts in batches.

You can even go further and look at the accounts for which these trade receipts have debited and credited by joining ar_receivable_applications to the gl_code_combinations table via code_combination_id.

The gl_import_references table links the General Ledger journal tables to Receivables.

```
SELECT
        Gb.name gl_batch_name,
        B.name ar_batch,
        NVL(SUM(DECODE(ra.status,'APP',  ra.amount_applied,0)),0) +
        NVL(SUM(DECODE(ra.status,'ACC',  ra.amount_applied,0)),0) +
        NVL(SUM(DECODE(ra.status,'UNAPP',ra.amount_applied,0)),0) +
        NVL(SUM(DECODE(ra.status,'UNID', ra.amount_applied,0)),0) rec_amount
FROM    ar_receivable_applications_all       ra,
        ar_cash_receipts_all                 cr,
        ar_batches_all                       b,
        ar_cash_receipt_history_all          crh,
        gl_je_batches                        gb
WHERE   ra.cash_receipt_id = cr.cash_receipt_id
AND     cr.cash_receipt_id = crh.cash_receipt_id
AND     crh.batch_id       = b.batch_id
AND     (crh.batch_id, crh.cash_receipt_id, gb.je_batch_id) IN
        (SELECT DISTINCT
                ab.batch_id,
                ac.cash_receipt_id,
                b.je_batch_id
        FROM
                ar_batches_all ab,
                ar_cash_receipt_history_all ah,
                ar_cash_receipts_all ac,
                gl_import_references i,
                gl_je_headers h,
                gl_je_batches b
        WHERE   h.je_header_id = i.je_header_id
        AND     b.je_batch_id  = i.je_batch_id
        AND     i.reference_2  = ah.cash_receipt_id
        AND     ah.cash_receipt_id = ac.cash_receipt_id
        AND     ab.batch_id(+)    = ah.batch_id
        AND     h.je_source    = 'Receivables'
        AND     h.je_category  = 'Trade Receipts'
        AND     h.je_batch_id  = b.je_batch_id
        AND     h.period_name  = 'Apr-00')
GROUP BY gb.name, b.name
ORDER BY 1, 2
/
```

Information Management

You schedule your orders but you are running low in stock for some items. The following example displays the reservation information for the item parameter. Based on this information, you can then decide which orders can be rescheduled.

The table mtl_demand stores the reservation information and fnd_user contains user information. The majority of the tables have the columns created_by, last_updated_by that represents the user who entered the transaction and the last user to have updated the transaction, respectively.

```
SELECT DECODE(d.reservation_type, 1, 'On Demand', 2, 'Reserved') Res_type,
       SUBSTR(u.user_name, 1, 20) User_name,
       SUBSTR(i.segment1, 1, 10) Item,
       c.customer_name cust_name,
       c.customer_number cust_number,
       h.order_number Ord_no,
       substr(l.shipment_priority_code, 1, 10) ship_code,
       h.date_ordered Date_ordered,
       r.organization_code Org,
       d.line_item_quantity demand_qty
FROM   so_headers_all h,
       mtl_demand d,
       mtl_sales_orders s,
       mtl_system_items i,
       org_organization_definitions r,
       fnd_user u,
       ra_customers c,
       so_lines_all l
WHERE   h.created_by           = u.user_id
AND     d.inventory_item_id    = i.inventory_item_id
AND     d.organization_id      = i.organization_id
AND     h.order_number         = s.segment1
AND     r.organization_id      = d.organization_id
AND     c.customer_id          = h.customer_id
AND     d.demand_source_line   = l.line_id
AND     l.header_id            = h.header_id
AND     d.line_item_quantity > 0
AND     i.segment1 BETWEEN '&&item_from' AND '&&item_to'
AND     d.demand_id IN
        (SELECT MAX(d1.demand_id)
        FROM  mtl_demand d1
        WHERE d1.line_item_quantity > 0
        AND   d1.demand_source_type in (2, 8)
        AND   d1.parent_demand_id IS NOT NULL
        GROUP BY d1.reservation_type, d1.parent_demand_id)
ORDER BY
DECODE(d.reservation_type, 1, 'Demand', 2, 'Reserved'),
SUBSTR(u.user_name, 1, 20),
SUBSTR(i.segment1, 1, 10)

/
```

Your customers have been returning products, but why ? Is it the product quality, is it the delivery ? How can the business improve ?

You enter return reasons on Return Material Authorisations (RMAs) and the following example reviews the returns for a given date range (where &&from_date and &&to_date are parameters in SQL*Plus) for when the RMAs have been created.

```
SELECT
        l.transaction_reason_code reason_code,
        c.customer_name cust_name,
        SUBSTR(s.segment1, 1, 10) item,
        h.order_number ord_no,
        t.trx_number inv_no,
        0 - NVL(l.ordered_quantity, 0) - NVL(l.cancelled_quantity, 0) rma_quantity,
        NVL(tl.quantity_credited, 0) qty_credited,
        NVL(tl.quantity_credited, 0) * NVL(unit_selling_price, 0) total_cr,
        u.user_name creator_name,
        h.creation_date creation_date
FROM    fnd_user u,
        mtl_system_items s,
        ra_customers c,
        ra_customer_trx_all t,
        ra_customer_trx_lines_all tl,
        so_headers_all h,
        so_lines_all l
WHERE   h.order_category                    = 'RMA'
AND     h.cancelled_flag is null
AND     tl.customer_trx_id                  = t.customer_trx_id(+)
AND     tl.interface_line_attribute6(+)     = l.line_id
AND     c.customer_id                       = h.customer_id
AND     u.user_id                           = h.created_by
AND     l.inventory_item_id                 = s.inventory_item_id
AND     l.warehouse_id                      = s.organization_id
AND     h.header_id                         = l.header_id
AND     h.creation_date between to_date('&&from_date', 'DD-MON-RRRR')
AND     to_date('&&to_date', 'DD-MON-RRRR')
ORDER BY 1, 2
/
```

Alerts

You need to ensure that all orders entered for the day is booked in order to progress to the next step. Therefore, you may setup an Oracle Alert based on the following SQL*Plus and email the output to the Sales Order Administrator. The following can also highlight training issues if one particular user is not booking orders on a regular basis as indicated by the Created_by column.

```
SELECT
        o.organization_code Org_name,
        SUBSTR(msi.segment1, 1, 10) Item,
        SUBSTR(msi.description, 1, 45) Item_desc,
        sh.order_number Order_number,
        sh.date_ordered Date_ordered,
        SUBSTR(fu.user_name, 1, 12) Created_by,
        NVL(sl.ordered_quantity, 0) - NVL(sl.cancelled_quantity, 0) Qty_ordered,
        sl.selling_price*sl.ordered_quantity Ext_amount
FROM    so_headers_all sh,
        mtl_system_items msi,
        so_lines_all sl,
        fnd_user fu,
        org_organization_definitions o
WHERE   sh.header_id          = sl.header_id
AND     o.organization_id     = msi.organization_id
AND    sh.order_category IN ('R', 'RMA')
AND     sl.inventory_item_id  = msi.inventory_item_id
AND     sl.warehouse_id       = msi.organization_id
AND     sh.created_by         = fu.user_id
AND     (sh.s1 = 5 or sh.s1 = 15)
AND      sh.cancelled_flag is null
AND      sh.creation_date BETWEEN TO_DATE('&&from_date, 'DD-MON-RR')
                               AND TO_DATE('&&to_date', 'DD-MON-RR')
ORDER BY
        o.organization_code,
        SUBSTR(msi.segment1, 1, 10),
        SUBSTR(msi.description, 1, 45),
        sh.order_number
/
```

About the Author

Jenny Chan is an Applications Consultant with Red Rock Consulting based in Sydney, Australia. She has 9 years experience within the Oracle environment, 5 of which have been working with Oracle Applications. You can reach her by email at chanj@redrock.net.au and there are web pages of articles and tips at http://www.redrock.net.au.