Oracle Contracts - An Introduction

1. What is a contract?

A contract is an agreement entered into voluntarily by two parties or more with the intention of creating a legal obligation in which there is a promise to do something in return for a valuable benefit known as consideration.

Since the law of contracts is at the heart of most business dealings, it is one of the significant areas of legal concern and can involve variations on circumstances and complexities. The existence of a contract requires finding the following factual elements

- An offer.
- An acceptance of the offer which results in understanding of two parties.
- A promise to perform and a valuable.
- A time or event when performance must be made.
- Terms and conditions for performance, including fulfilling promises.

2. Oracle EBS Contracts

In real life every organization, every business is closely working within the boundaries of contract laid between two parties. It can be a manufacturer and a dealer or an independent worker and an organization. In this case all happens within Oracle E-Business Suite.

In fact, contracts govern most businesstransactions. Companies rely on contracts to define:

Specific products or services that they sell or buy.

Terms and conditions that govern pricing, shipment, payment, quality, and othermutual business and legal obligations of the involved parties.

Oracle contracts provide common infrastructure components that will be uses across all the different modules that deal with contracts.

- Contract Terms Library
- Contract Terms Authoring and Printing
- Contract Document Management
- Contract Repository
- Contracts Workbench

2.1. Contracts Terms Library

Every organization deal with contracts and have a separate administrative and legal team that negotiates the contract and author them. Every year they work on a number of such legal documents and it will be a time consuming task over a period of time. Oracle's Contracts Term Library provides feature that simplifies the task of documenting and maintaining these terms.

Organization having their presence globally uses such libraries and enforces them to be applied all over the globe. It also has feature of accommodate local regulations as well that give local authorities to amend as per country standards.

Contracts terms library provides following features:-

- **Create / Manage Standard Clause**–Legal department and contract administrators create clauses that contain standard business terms used in the contracts. Legal people can create clauses as per business needs and route them for approval. Once approved it can be published for use across the organizations. Clauses can be modified by creating a new version.

- **Create / Manage Contract Expert Rule** – Contract expert is a rule based contract authoring tool that helps organizations define their policies to select additional clauses automatically during authoring and report how contracts deviate from their standard process. We have two rules –

o Clause selection rules
o Policy deviation rules

- **Create / Manage template** – Organizations implement best practices by creating standard contract template based on their day to day contracting needs. It facilitates easy creation of contract authoring that minimizes the overhead in authoring, legal review and approval.

2.2. Contract Term Authoring and Printing

Oracle contracts core facilitates comprehensive authoring capabilities integrated with service, procurement or sales business flows.

For example Oracle Sales Contracts, provides contract term authoring available via Quoting and Order Management. We can also publish our contract term over our iStore. Similarly with Oracle Procurement contracts we can author contracts term in Sourcing and Purchasing. Also we can negotiate contract terms with our supplier using iSupplier portal.

Some of the standard features are as follows:-

- We can use preapproved templates defined in the library to quickly author initial versions.
- We can make offline changes to the contract term by downloading it into a word document and uploading after modifications.
- Review contract deviations from standard predefined contract terms.
- Preview and print contact terms.

2.3. Contract Document Management

Creating and maintaining contract version is time consuming and tedious task even its stored electronically. Oracle contracts provide a central repository that can be accessed by users across the globe. Users can also load signed copies and other supporting stuffs into the repository. Since its easily accessible user can find any version of the contract.

2.4. Contract Repository

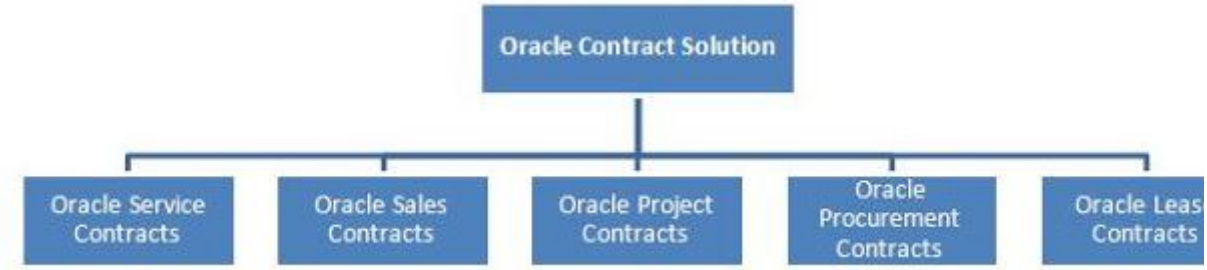
Contract repository enables us to create miscellaneous contract like license agreements, merger agreement which are not specifically falls under Oracle contracts application. We can also create sales of purchase agreement which do not fall under standard sales or purchasing flow that does not require the full execution capabilities provided by application.

2.5. Contract Workbench

Contract workbench provides single and unique view of all the organizations contracts for effective management. It gives us access and visibility to all the contractual agreements in the organization.

Contract workbench provides extensive search capabilities, provides details of which contract requires attention and has link to common task like creating a copy of contract, renewing contract and approving it. It also comes with security features which restrictuser's only view contracts.

3. Oracle E-Business Suite's contract management solution



Modules in Service Contracts

3.1. Oracle Service Contracts

Oracle Service Contracts provides a complete contract authoring execution solution to manage warranties, extended warranties, usage, subscription services etc from its authoring, approval, billing or termination or renewal of entire contract or specific part of the contract.

- With Oracle Service Contracts we can:
- Sell different types of service to organization or end consumers.
- Define pricing and billing schedules for flexible billing and payment.
- Automate renewals for recurring revenue opportunities
- Simplify change management for existing contracts.

3.2. Oracle Sales Contracts

Oracle Sales contracts is an essential entity for businesses where the terms and conditions of business transactions are defined and are legally binding on both parties. The Sales Contracts functionality is a vital part of business processes like quoting, ordering, and negotiating long-term agreements.

Oracle Sales Contracts enhances the ability of sales organizations to manage their contracts by adding sophisticated contract management and compliance features to Oracle Quoting, Oracle iStore, and Oracle Order Management.

In a typical Sales contract business flow it goes through following steps:

- Establishing contract term that govern the standard business policies and procedures.
- Authoring and Negotiating contact with customer.
- Approval of the contracts upon successful negotiation.

3.3. Oracle Project Contracts

Oracle Project Contracts supports the contract management needs of project centric organizations those rely on comprehensive project management capabilities to sustain in business which includes commercial and government contractors, agencies, and subcontractors. These organizations operate in a project driven environment characterized by:

- Changing contract specifications.
- Volatile demand and long lead-times.
- High percentage of procure-to-contract components and services.
- Contractual requirements for billing.
- Compliance of government regulations.

Executive and operational management constantly face issues of budgetary constraints, contract margins, risk management, flowdown of contract information to subcontractors, and prioritization of contract deliverables. Oracle Project Contracts is designed to address these business issues by providing a comprehensive contract management

3.4. Oracle Procurement Contract

Most of organizations rely on procurement of goods or services to run their business smoothly. Procurement contracts play a critical role in all businesses. Companies rely on procurement contracts to define:

- Specific goods or services that they buy.
- Contract terms governing their price, shipment, payment, quality, and other mutual obligations of the involved parties.

Oracle Procurement supports the basic business processes of the procurement contract life cycle. Oracle Procurement Contracts enhances the ability of buying organizations to manage these stages by adding sophisticated contract management and compliance features to Oracle Purchasing, Oracle Sourcing, and Oracle iSupplier Portal.

3.5. Oracle Lease Contracts

Oracle Lease contracts have been renames to Oracle Lease and Finance Management that provides end-to-end automation and a foundation for growth of your lease and loan portfolio. Designed for asset-based finance companies spanning the entire lease life cycle, Oracle Lease and Finance Management is a comprehensive application that leverages the power of the Oracle E-Business Suite to help equipment financers increase revenues through using IT as a strategic differentiator, enabling them to implement and accomplish strategies to:

- Expand business in existing and new markets
- Focus on business relationships with customers, vendors and investors
- Manage risk-based returns

- Improve asset management capabilities

Oracle Service Contracts - Part-I (An Introduction)

Oracle Service Contracts

Oracle Service Contracts (also referred to as Service Agreements) are contracts that are usually sold to customers to support, repair and/or maintain some product or service that the customer owns. Oracle Service Contracts provides a complete contract authoring execution solution to manage warranties, extended warranties, usage, subscription services etc from its authoring, approval, billing or termination or renewal of entire contract or specific part of the contract.

With Oracle Service Contracts we can

- Sell different types of service to organization or end consumers.
- Define pricing and billing schedules for flexible billing and payment.
- Automate renewals for recurring revenue opportunities
- Simplify change management for existing contracts.

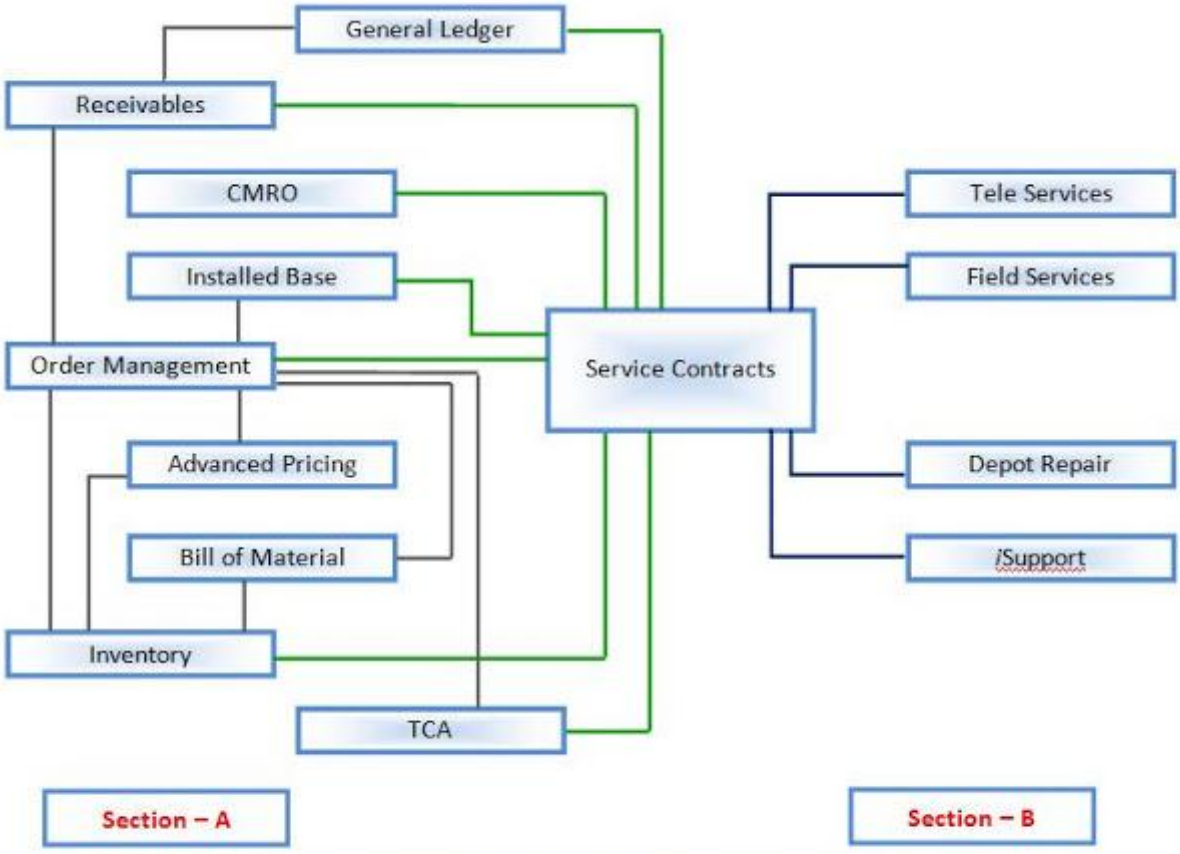


Fig - Service Contracts Integrations in E-Business

I have divided Service Contracts integration's into two Sections:-

Section – A Lists modules of Oracle Application which provide information to Service Contract. In other terms we can say that these module provides input to Service Contracts

Section – B List outs module of Oracle Application which retrieves information from Service Contracts. In other terms we can say that these module get the input from Service Contracts.

Please find the details of features used by Services Contract from different modules.

Section – A

Inventory

- Manages items associated with Oracle Service Contracts agreements and contracts: services, usages, warranties, subscriptions, and serviceable items

Order Management

- Raises sales orders from which warranties and extended warranties are created.
- Inventory is used to define serviceable products and service items (warranties and extended warranties).

Receivables

- Creates invoices and credit memos from the billing information generated by Oracle Service Contracts.

Bills of Material

- Associates warranties with serviceable items.

Advanced Pricing

- Supplies the information to price the services and usages sold on the contract.

Installed Base

- Provides details of trackable items and the tracking of counters.

CMRO (Complex Maintenance, Repair, and Overhaul)

- Provides details of preventive maintenance programs that may be associated with service contracts.

Section - B

TeleService

- Manages service requests.
- Retrieves contract details for a customer such as account, end date, status, contract type& number on all of their contracts.

- Looks for the preferred engineers for a particular service.
- Has visibility to coverage levels & times, reaction & resolution times for coverage, billing rates and types for a specified coverage.

iSupport

- Provides customer access to service contracts to view entitlements, preferred technician, response and reaction times via self service mechanisms

Depot Repair

- Provides details for repairs, exchanges, replacements, and loaners, which may update the contract coverages.
- Has visibility to coverage details.

Field Service

- Provides details about service and repairs to be carried out in the field.
- Has visibility to coverage details.

Contd... Oracle Service Contracts - Part-II (Categorization / Structure)

Oracle Service Contracts - Part-II (Categorization / Structure)

Oracle Service Contract Categorization

Service Contracts has been categorized in 3 categories:-

Fig - Service Contracts Categories

Warranty & Extended Warranty

As the name suggests it supports two different kinds of services:-

- **Warranty**
- **Extended Warranty**

Warranty is nothing but the service offer which is offered by the manufacturer / seller to the customer. Warranty is always free of cost and comes while a product is sold from Order Management. There is no way where we can provide warranty on the demand.

Example – Patrick went to market and purchased a mobile phone. It comes with 1 Year Warranty which supports free service of device for one year and replacement of battery if shown poor performance within 6 months.

Extended Warranty is kind of services which are charged to the customer while the product is sold. We can also sell extended warranty contracts separately after the product is sold from Order Management or Oracle Service Contract Module.

Example – While Patrick was buying the mobile phone the seller offered Extended Warranty at a cost of $90 / Year. The extended warranty provide service offer in which Patrick can get the mobile device repaired for 2 Year free of cost and one time refurbishment.

These contracts often take advantage of the ability to bill on a recurring basis either monthly, quarterly or annually and can give the customer the ability to pay for their services in advance or in arrears of the period of service. But this feature is only available if the extended warranty contract is sold from service contract module. If sold from Order Management then it has to be bill in advance for entire contract duration.

Subscription Agreements are effective way to sell your services / products. It allows users to get the services / products for a period of time that often have monetary benefits involved for customers. Generally customer ends up paying discounted amount for the services.

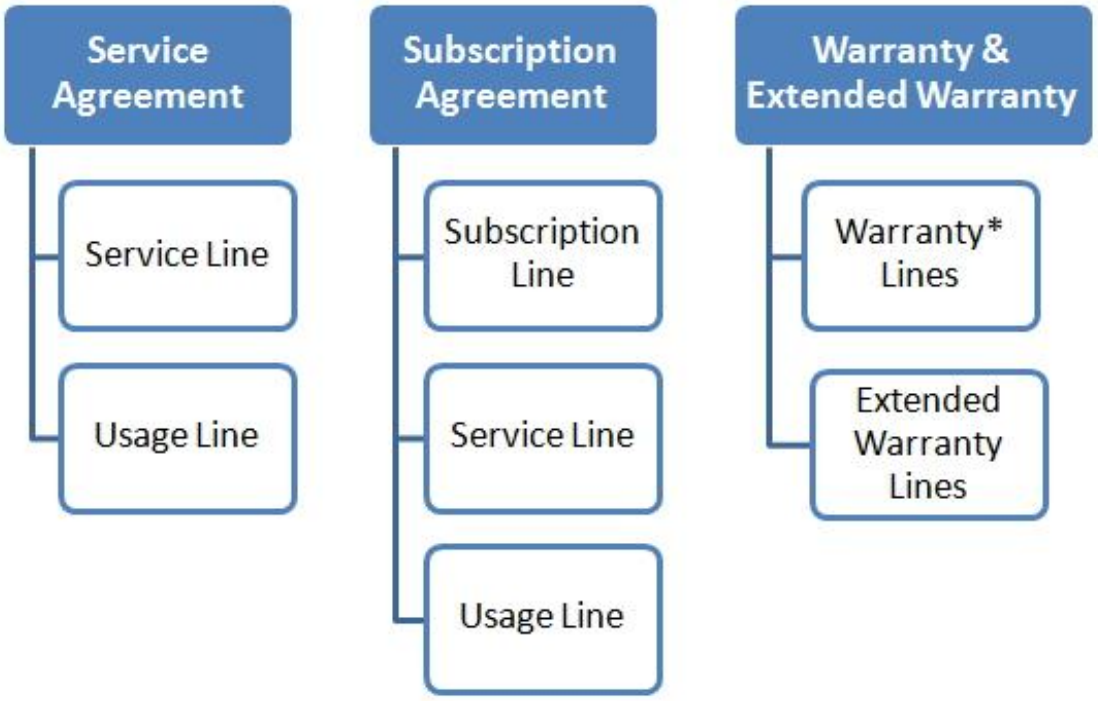
There is organizations that offer initial subscription on a very low price / free trial and then on a fixed charge if customer wants to continue. It also gives flexibility to either charge customer all at one go or billing for every subscription.

In Oracle Subscription Contracts we can sell subscriptions for both tangible and intangible items. Tangible items include magazines, collateral, or any other physical item that can be shipped through Oracle Order Management. Intangible items can be collateral sent via e-mail or permission to access a web site for a set period of time.

Service Agreements (also referred as Service Contracts) are contracts those are sold to customers to repair, support / maintain product or services that a customer has or sold by the vendor / seller. All the services agreements bind within the boundaries of terms and conditions that is associated with the contract.

Service agreement give the flexibility to bill the customer on a recurring basis like Monthly, Quarterly, Half yearly or annually and gives customer the ability to pay in advance or in arrears. Service agreements can be created manually or created automatically from Order Management. We will discuss more on that in later sections.

Oracle Service Contract Categories

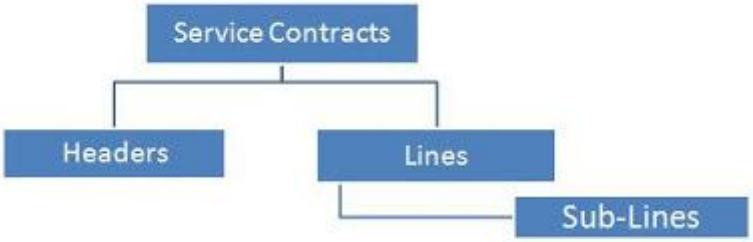


***Note:** Warranty line will not show while we create contract manually, always created automatically from OM.

- Service Contract supports different Line types – Each category will have specific line types supported to solve a business purpose
- Service Line will be used while we sell the services. For this the service item must be setup in Oracle Inventory.
- Usage Lines are used to calculate the charges where we charge to customer based on the usage of device, like counters in photo copier, electric meter.
- Subscription Lines are use where we provide subscription of products like magazines or services like software update.
- Warranty Lines will be use for warranty contracts created from Order Management. We cannot create warranty lines manually.
- Extended Warranty Lines is used where we sell extended warranty to customer. It may originate from Order Management or manually created in Service Contracts.

Oracle Service Contracts Structure

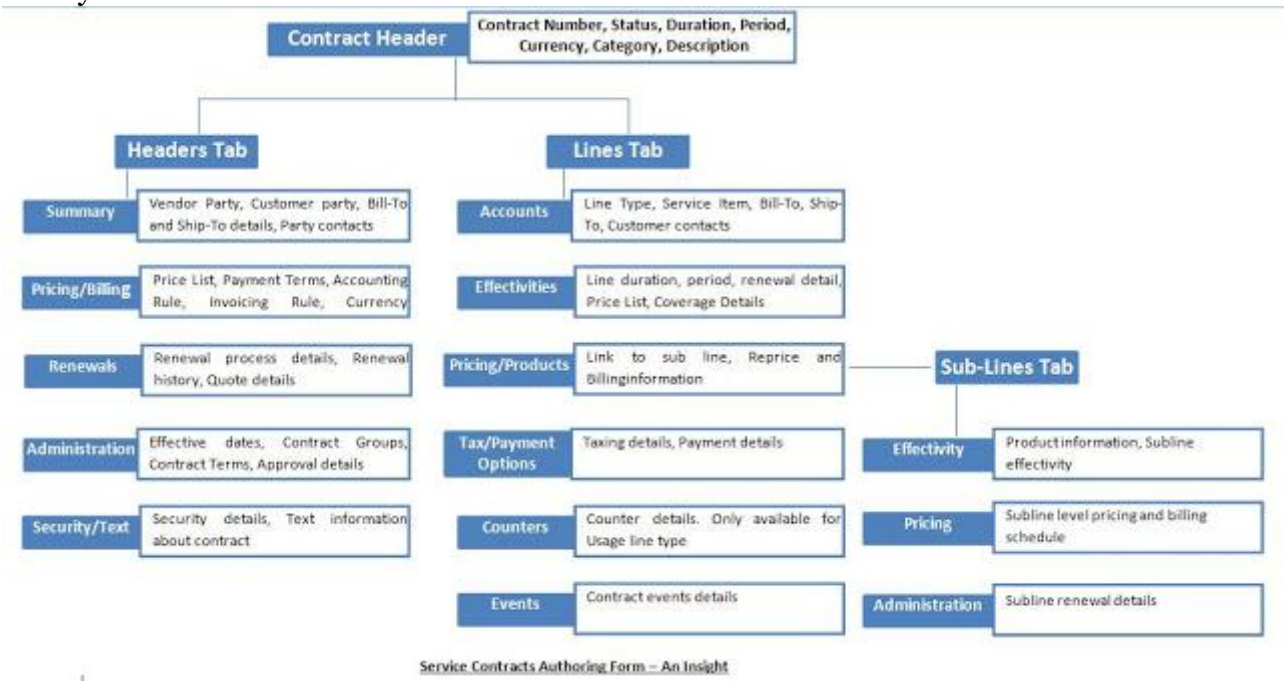
Service Contract has been categorized into 3 sections:-



Service Contracts details breakup

- Contract Header consists of very basis information that is required to create a contract e.g. Vendor and Customers, Contract duration, Currency, Party Roles, Pricing and Payment details etc.
- Contract Line consists of information about the service that organization is selling / offering e.g Gold24x7 service which will provide service around the clock etc.
- Contract Sub-Line consists of details of the actual product that will avail/get the service e.g. A TV, Laptop, and Mobiles etc.

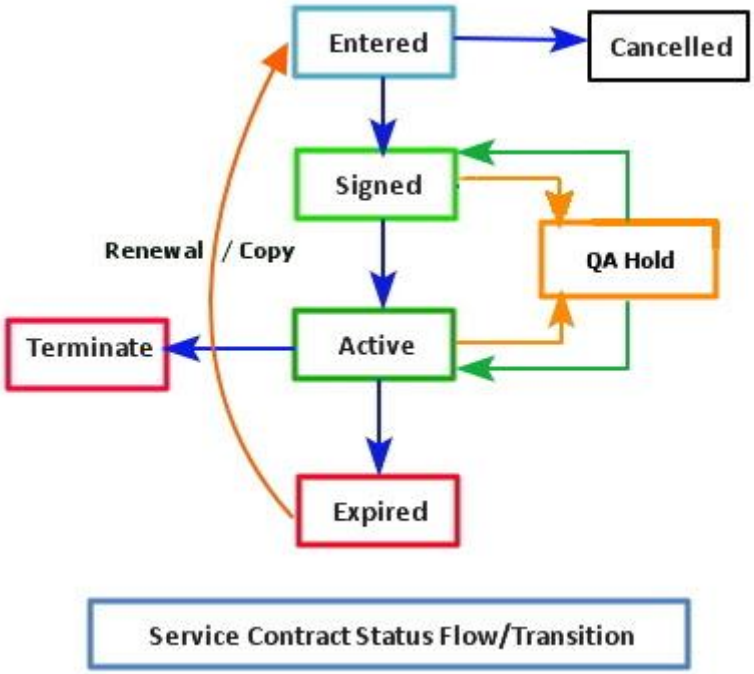
All the section has specific purpose and information related to contracts. Let us closely understand what all different data is stored at each level.



Service Contracts Authoring Form – An Insight

Contd... Oracle Service Contracts - Part-III (Status Transition / Lifecycle)

Oracle Service Contracts - Part-III (Status Transition / Lifecycle)



- **Entered** – It's the initial status when a contract is authored, it can also refer this as Draft where we are just putting details on a contract.
- **Cancelled** – A contract can be cancelled if it's not needed to work on. We can only cancel a contract when its status is Entered.
- **Signed** – Once a contract is approved and submitter signs the contract its status becomes 'Signed'. In real scenario its very rare that we see the status for a contract because it move to active as soon as its signed. If the start date of the contract is future dated, then it will be in Signed status and will move to Active once start date equal to system date.
- **Active** – Upon approval and signing a contract is moved to Active if start date of the contract is back dated or equals to system date.
- **QA Hold** – Once a contract is in Signed/Active status and if it needed some change we open the contract for update. Once we open the contract for update it will move to QA Hold status. After making the required changes perform QA check that will move it back to Signed/Active.
- **Expired** – Once contract end date passes the system date it will move to Expired status.
- **Terminated** – A contract will move to Terminated status if we terminate the contract in between its active period. Upon termination it may or may not generate credit memo that is based on the contract terms. If the contract is in QA Hold it cannot be terminated. Also once a contract / line / sub-line is terminated we cannot revert it back, so precaution needs to be taken before termination.

Contd.. How to Create a Service Contract

Oracle Service Contracts - Part-IV (Creation of Service Contract)

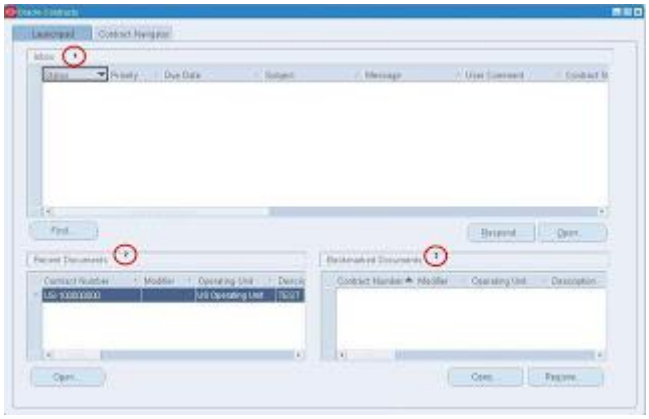
Create a Service Contract

Based on the business requirement we create different type of contracts. In this demonstration we will create a service agreement.

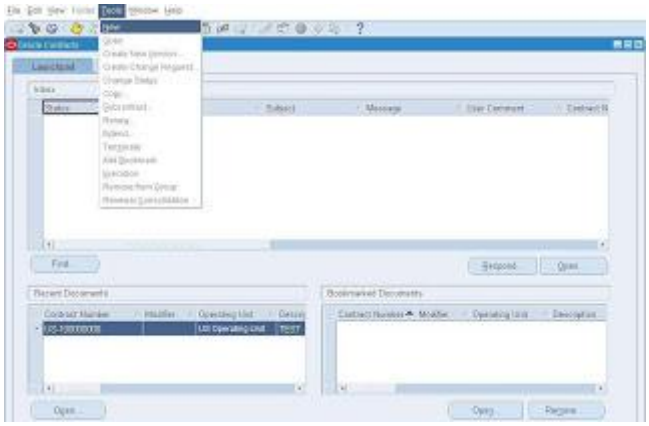
- Responsibility – Service Contracts Manager
- Navigation to Create Service Contracts i.e Launch Pad
- Contract Administration -> Launch Pad

Launch Pad has 3 sections:

1. Inbox – It gives all the notification that is sent to logged in user related to service contracts.
2. Recent Document – Shows up last 10 contracts that the user is working on.
3. Bookmarked Document – Shows any bookmarked contract that user bookmarked for quick reference.

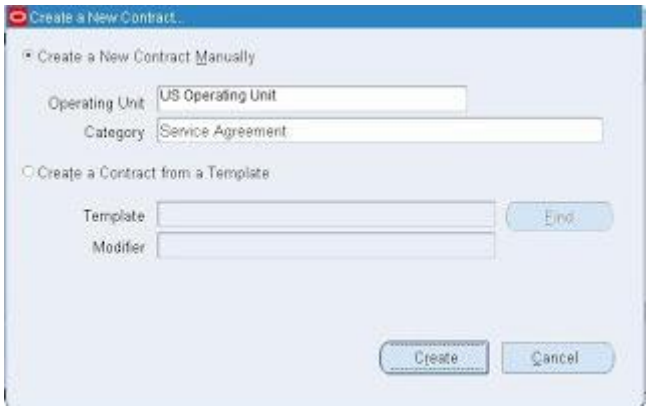


Navigate to Tools –>New

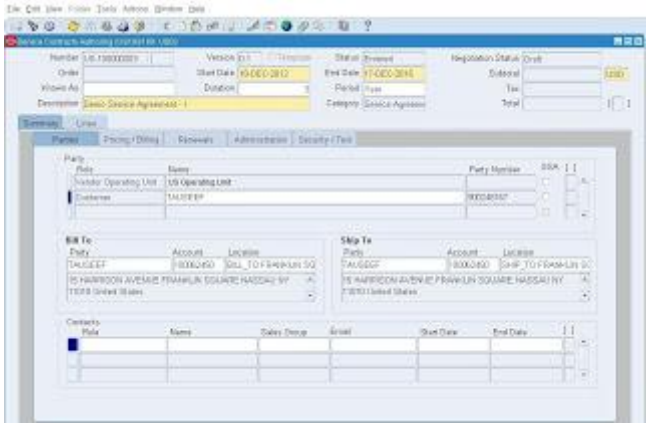


- Choose Operating Unit – This is nothing but the Vendor which will be selling the contract.

- Choose Category – **Service Agreement**
- Click **Create**

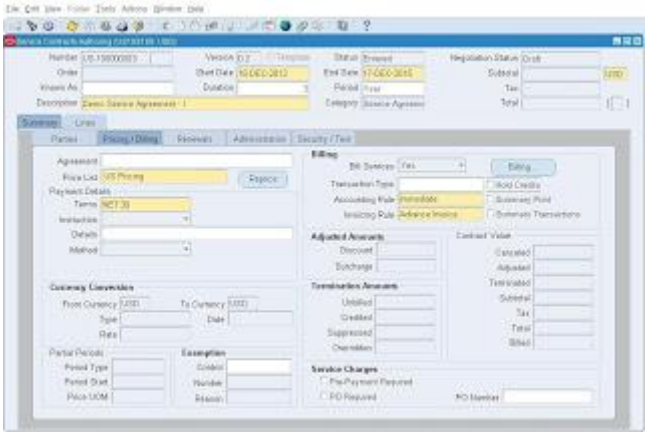


We can also use – **Create Contract from a Template**. If chosen we need to select an already create template. Choosing a template eases our work and requires less data entry.
Service Contract Authoring form opens, By default **Summary –> Parties** tab will be open.



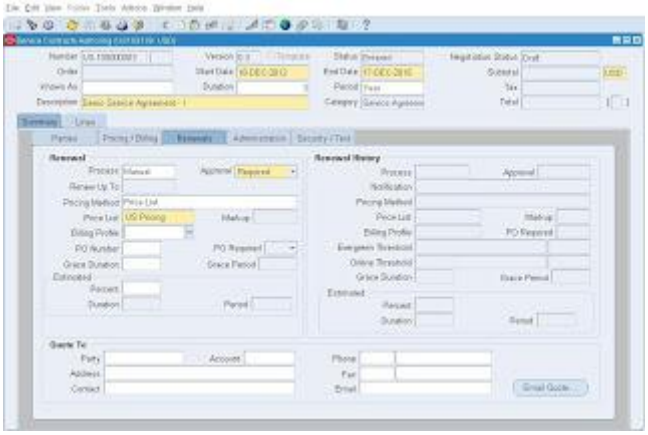
- **Number** – Enter a number. If the auto numbering setup is done then it will be grayed out and number will be populated automatically once we save the record.
- **Status** – Initial status of the contract is 'Entered' as per seeded functionality. We can change as per business requirement. Once contract is approved and signed it will move to Active status.
- **Start Date / End Date /Duration / Period** – By default start date will be system date, but if required we can create back dated contract as well. Either put an End Date or provide data for duration and period that will automatically calculate end date.
- **Currency** will be auto populated based on the ledger setup.
- Select appropriate party role and select a valid party.
- Select Bill-To and Ship-To locations
- If need add contact details for Vendor and Customer parties.

Navigate to Summary –> Pricing tab



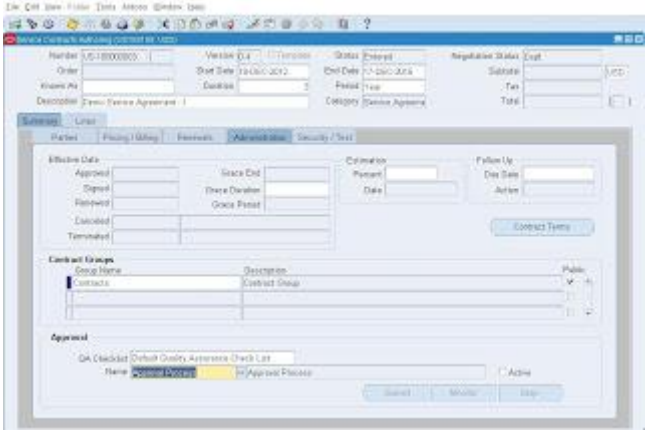
- **Price List** – Choose a price list from LOV. Price List are defined in Advanced Pricing and used to get the price of the services.
- **Accounting Rule** – Chose Accounting Rule from LOV. It determines how the accounting will happen in Receivables once contract is billed.
- **Invoicing Rule** – Choose Invoicing Rule from LOV. If we are charging in advance for the services then select 'Advance Invoice'. If services need to be charge after we provide them then select Arrear Invoice.
- **Payment Term** – Choose the payment term that will be reflected on the invoice.
- Adjusted Amounts – Shows any discounts / surcharges that has been applied on the list price of the item
- **Contract Value** –This section show detailed information about pricing. It show terminated amount, total amount and amount billed till date.
- **Currency Conversion** – Shows details related to currency conversion if there is any. If we use currency conversion we need to provide the conversion type, conversion date and conversion rate.

Navigate to Summary –> Renewals Tab



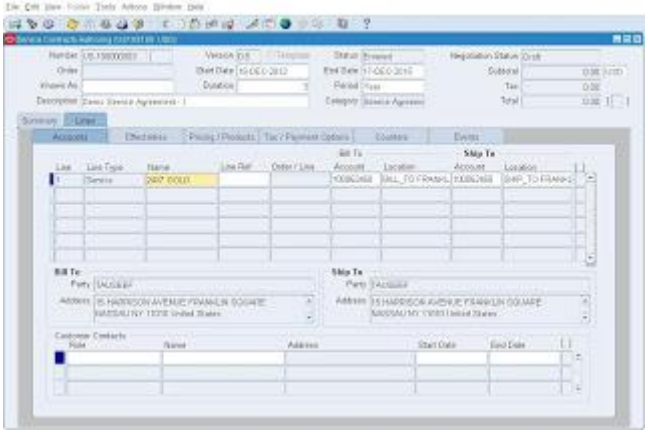
- **Renewal** – We enter the renewal details in this section. We decide what should be done with the contract upon expiration. If we want to renew we provide the Renewal Process. Choose a renewal process. If approval is required choose appropriate value.
- **Renewal History** – When we create contract for the first time it will be null. This field will have details if it's a renewed contract.

Navigate to Summary–> Administration



- **Effective Date** – Most values in this section will be automatically populated. We use this only when we need to provide a grace pried on the existing contract.
- **Contract Groups** – Choose a contract group from LOV. This is used for logical grouping of contract based on business needs.
- **Approval –>QA Checklist** – Select a Quality Assurance checklist. It has list of check list that need to be satisfied before activating the contract.
- **Approval –>Name** – Select approval workflow. By default value is 'Approval Process'.

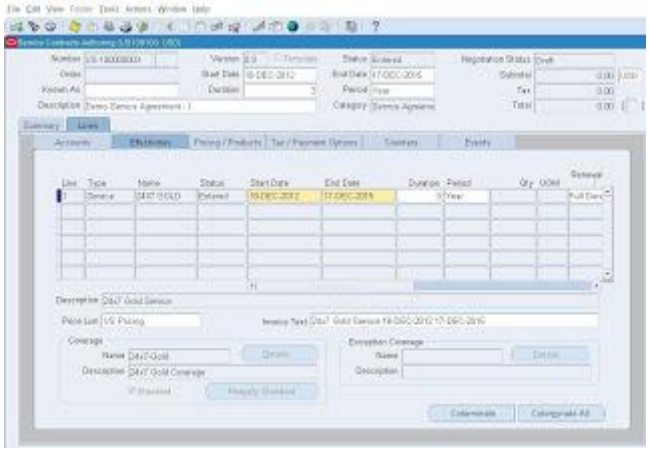
Navigate to Lines -> Accounts



- **Line Type** – Select appropriate line type. In our case we have chosen 'Service'. Other value available for Service Agreement is 'Usage'
- **Name** – Choose a valid Service Item. Let's understand the difference between service item and serviceable item

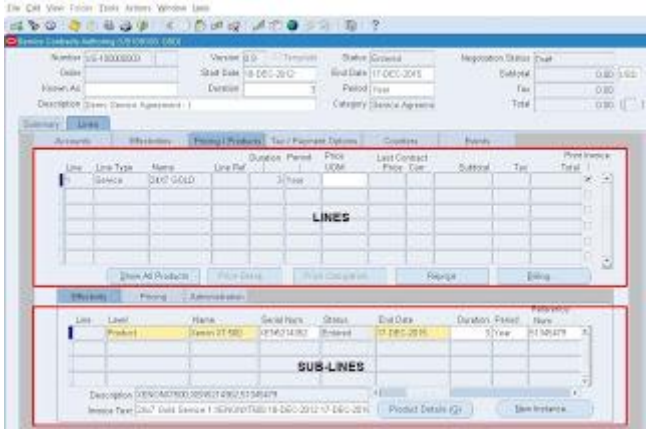
Service Items	Serviceable Items
• Service Item is an inventory item which will be sold from Service contract. • It has an associated coverage template. • It's available at line level only. • Service Item is defined in price list. • Non IB Trackable and contract enabled	• Serviceable item is an inventory item which will get serviced. • No association with coverage template. • It is available at sub-line level. • IB Trackable and Contract coverage enable

Navigate to Lines ->Effectivities



In effectivities tab we can change the effective date of a contract line. At any given point of time the start date and end date of a line must fall within contract header date.
Additionally we can choose different price list and modify the coverage details for specific line.

Navigate to Lines -> Pricing/Products



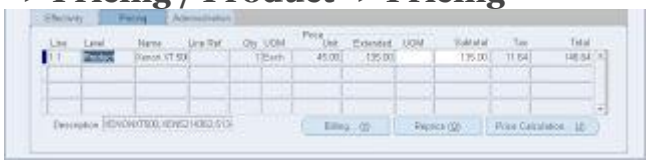
Pricing / Products will open the doors to navigate to sub-lines. It actually hold the consolidated price and taxing details of its child sub-lines. A single service line can have multiple sub-lines.
In this section we navigate to sub-lines and enter the serviceable item details.

Navigate to Lines -> Pricing / Products ->Effectivity

Choose the covered level of the serviceable item. It determines what is actually going to be services. Oracle supports 6 different levels

- **Item** refers as an item in inventory. There is no reference to install base.
- **Party** refers to a party which has install base item instances. If we choose party all the product will be covered that is owned by the party
- **Product** represents a single item instance. We need to enter instance number in Reference Number filed.
- **Site** refers to a particular site of a party. If chosen all the instance installed to that site will be covered.
- **System** refers to a group of item instance that is clubbed together
- **Customer** refers to party account. If chosen all item instances associated with the customer account will be covered and eligible for service.
- We have selected 'Product'. Based on that either enter the serial number in 'Serial Number' field or enter the instance number in 'Reference Number' field.
- We can also change the effectivity dates at sub-line level. While doing so we need to make sure start and end dates fall with Line start and end date.
- The description field shows the concatenated value of "Item Description || Serial Number ||Instance Number"

Navigate to Lines -> Pricing / Product -> Pricing

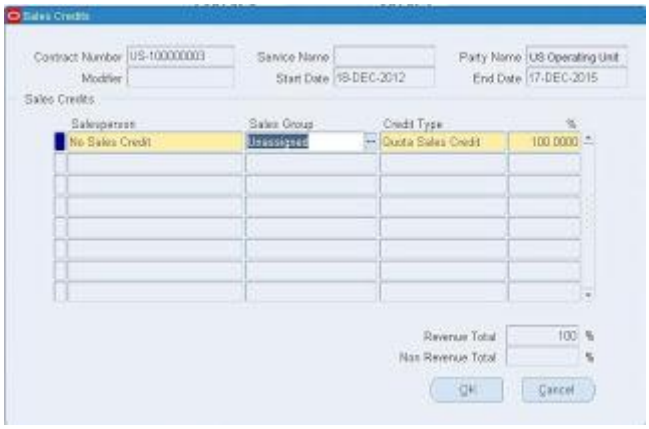


At sub-line pricing tab we get the price for the sub-line. We can define the billing schedule and reprice the sub-line as well.

Navigate to Lines -> Pricing / Product -> Administration

We can change a renewal type for sub-line, if we don't want to copy a particular sub-line while the contract is getting renewed.

Navigate to Actions - > Sales Credit



Enter the sales credit details. This is used to generate sales representative commission.

Once we are done with contract authoring we need to perform Quality Assurance check to verify all the required information / details are in place to activate the contract.

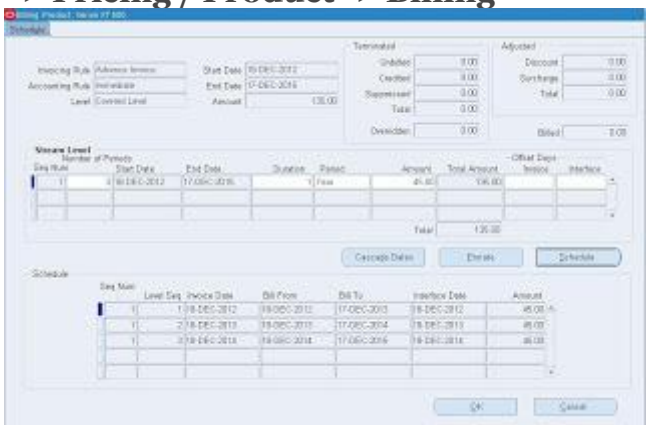
Navigate to Actions -> Check QA

QA check will result in 3 states. Success, Warning or Error.

- If a process result is success it as passed the check

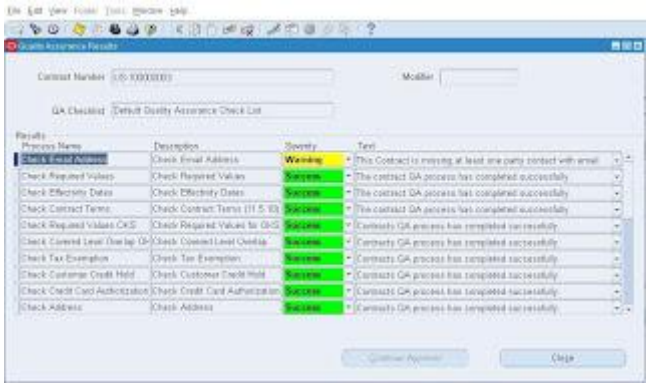
- If a process result comes as warning, we can ignore it and continue considering that warning does not impact have any legal / business impact.
- If a process results in Error then we can't continue. We must resolve the error to continue.
- As we see there are errors coming for billing schedule. We have not yet entered billing schedule, Let's enter the Billing Schedule.

Navigate to Lines -> Pricing / Product -> Billing



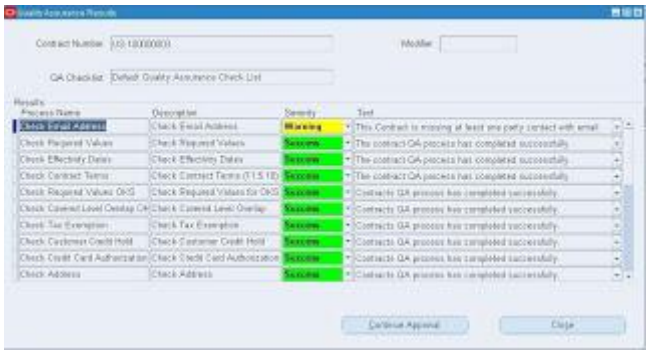
Billing schedule is required to let the system know how we going to bill to customer. Service Contact gives us flexibility to have flexible billing schedule which is suited for business / customer. We can bill the contract all at one go, annually, quarterly or monthly.
We will bill the customer annually; since it's a 3 years contract it generated 3 lines in the schedule.

Once done, we will perform QA check again.



Navigate to Actions -> Check QA
This time we don't have any error, so now this contract is good to go with approval process. But if you noticed the 'Continue Approval' button is grayed out. This is because we have only opted for QA Check. To submit it for approval

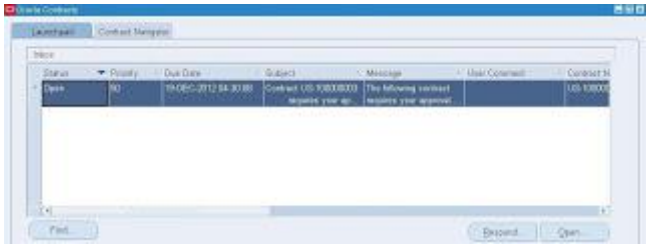
Navigate to Actions - > Submit for approval OR **Summary - > Administration ->Submit**



Now can see Continue Approval is enabled. Click on Continue Approval. Upon click the contract will be submitted for approval. Based on the approval setup done in AME (Approval Management Engine) it will choose the approver and send a notification.

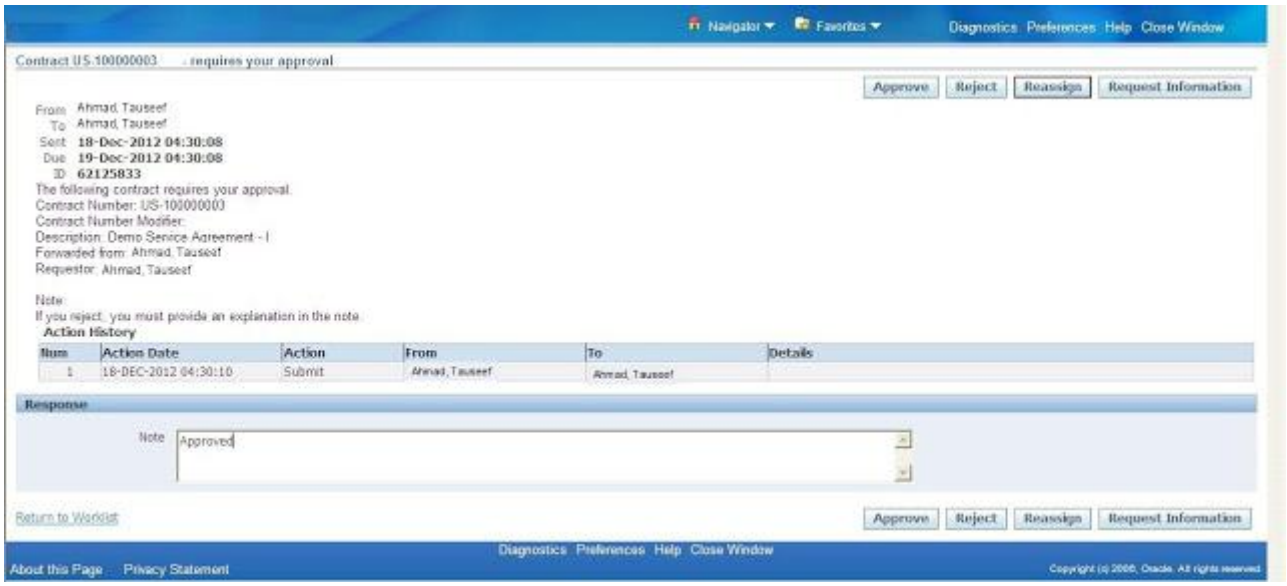
While the contract approval workflow is active the 'Monitor' and 'Stop' button will be activate also the active check box will be checked. That shows the approval workflow is action. We can check the current status of approval workflow by clicking on 'Monitor' button. If we need to stop the approval process then click on 'Stop' button.

Once upon submission we can find the notification in **Launch Pad –> Inbox**



We may also receive the notification in email if configuration is in place and workflow notification mailer services are up and running.

Click on **Respond**. It will open oracle workflow notification that allows to take required action.



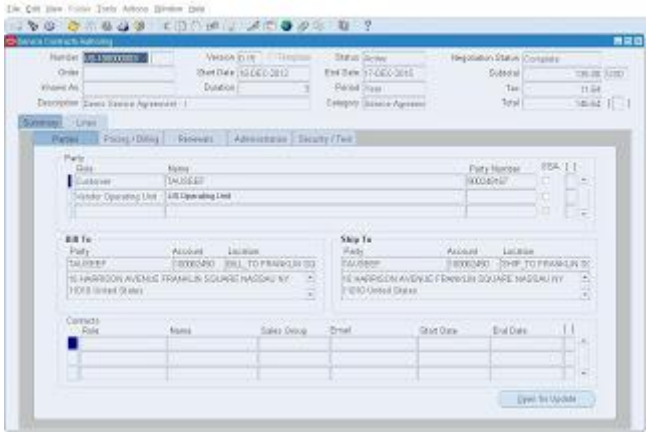
Provide some notification connects if needed and click on **Approve**. We can also **reject** but that mandates the notes. If you are not the right person we can reassign to correct person. If you feel it you need more justification before approving you can use '**Request Information**' button.

Once its approved, we will get another notification in our Inbox for signing the contract. Please note that whoever is assigned as approval in AME need to approve the contract. As far as signing is concerned it needed to be signed by the user who submitted the contract for approval.

Click on **Respond** to sign the contract. Oracle workflow notification page opens.

If you agree to sign the contract, click on '**Yes**'. Once its signed we get the details in our workflow notification.

The contract is approved and signed now. If we open it now the status will be changed to '**Active**'.



Once the contract is Active it does not allow to make any changes. All the fields become read-only. If there is need to make changes to active contract we need to use 'Open for Update' button. The contract will be moved to **QA Hold** and upon successful **Tools -> QA Check** it will again move to **Active**.

Contd… Oracle Service Contracts - Part-V (Billing a Contract)

Oracle Service Contracts - Part-V (Billing a Contract)

Billing Service Contract

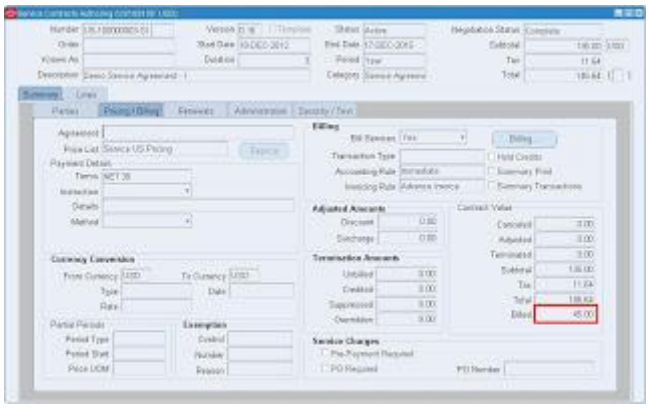
There is a concurrent program 'Service Contract Main Billing Program' that needed to run to bill the contract. Once we run the program it will interface data to Receivable to generate invoice.

Note – After Service Contract Main Billing program completes service contract will show as Billed in service contract. It does not mean an Invoice is generated. To generate the invoice we need to run Auto Invoice from Receivables.

Service Contract Manager -> View -> Request -> Submit a New Request -> Choose '**Service Contract Main Billing Program**' form LOV

Contract Number – Input contract number that you want to bill. You can leave other null.

Upon completion service contract will be showing as billed.



Point to be noted – The billed amount only shows as $45. The billing schedule is created for 3 years annually. Right now only eligible line is of value $45. Others have interface date in future.

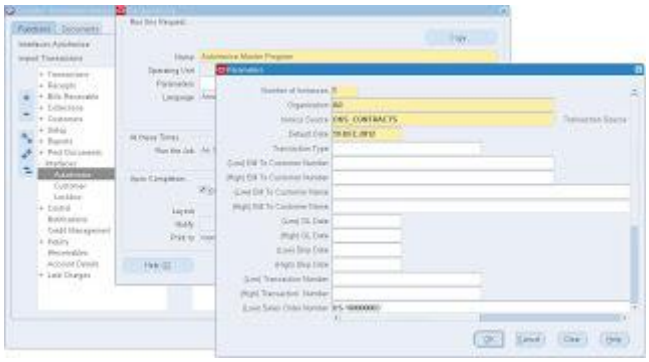
To generate the actual invoice in AR follow the steps.

Choose Responsibility – **Receivable Manager**

- Navigate to **Interfaces - >AutoInvoice** -> Submit a Request
- Select – **Auto Invoice Master Program**

Parameters :-

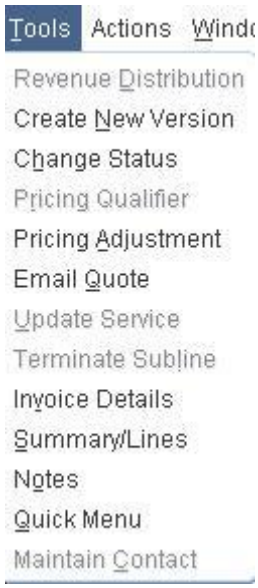
- Invoice Source – **OKS_CONTRACTS**
- Default Date – **SYSDATE**
- Sales Order Number – **Contract Number**



Once completed it will generate invoice.To view the invoice details in Service Contract run the below program.

Service Contract Manager -> View -> Request -> Submit a New Request -> Choose '**Service Contract Fetch Receivables Info for Billing**' form LOV

This will fetch all the invoice details related to service contract and can be view in Service contract form by navigating **Tools -> Invoice Details**



OR Navigate to **Lines -> Pricing / Product -> Billing -> History**

That concludes a cycle of service contract creation to billing.

Unmasking Oracle Service Contract API's

Introduction

With the recent increase in implementation efforts for Oracle CRM products, the IT department now faces new challenges of converting legacy / home grown application data into Oracle. However most of these companies have some prior experience of data conversion into Oracle Application from their earlier ERP implementations. Instead of interface tables and concurrent programs to import legacy data, Oracle uses API's. Unlike 10.7 documentation, these APIs are mostly not documented.

A typical Service Contract has a header and multiple lines. Beside the REGULAR contract information, the header record identifies the CONTRACT PARTIES, RULES to determine renewal, pricing, servicing and approvals. The lines depending on the LINE TYPE is attached to Install Base or a Usage item type (e.g. counters). Also attached to these lines are BILLING SCHEDULES and RULES that determine line servicing, interfaces to Oracle AR and creating Invoices.

Oracle 11i Contracts for services shares its table / data structure with Oracle Core Contracts. In fact the service API's after setting the flags depending on the profile options and other set-up values call the core contract APIs to create the contract. Below ERD establishes high-level relationship between Core Contract tables.

Screenshot 1

Given the complex nature of conversion, data requirements and mapping exercise should not be carried out in isolation. It should include process owners, members of legacy IT, consultants (functional and technical) who are putting Oracle Service Contracts in place. Like any conversions, there is a need to create staging tables to store the contract header and contract line data received from legacy systems. The data stored in these tables is practically a load of flat files with no or minimum data translations or lookups. This table would help in reconciling what has been received from legacy and what was converted into Oracle.

Next in line would be, to define audit tables. These tables would store the final version of the data values, which are being used to create the contract and some additional fields to facilitate conversion and reconciliation. As stated earlier there is minimum documentation on the API's which would lead project team inclining to create direct inserts into Oracle tables. This is not recommended.

Core API

Below mentioned is the core API's that team uses to convert Oracle Service Contracts. It is assumed that billing schedules are not converted and they will be generated within application.

OKS_CONTRACTS_PUB.Create_Contract_Header:

This API is used to create contract header information. The API is called using P_K_Header_Rec, P_Header_Contacts_Tbl, P_Header_Sales_Crd_Tbl and P_Header_Articles_Tbl. Depending on your conversion needs and the available information in the legacy application you would be required to populate each of these in-parameters. However if no information is available for sales credit or contract articles you could leave the P_Header_Sales_Crd_Tbl and P_Header_Articles_Tbl NULL. However it is a must that you populate the P_K_Header_Rec with all the values. At least 1 contact information for the vendor should be created in P_Header_Contacts_Tbl. The value for the contact_object_code would be 'OKX_SALESPERS'. If you need to create additional parties on the contract header depending on your billing requirements and other entity relationships you may have to call Okc_Contract_Party_Pub.Create_K_Party_Role. Also you would need to determine in advance how do you wish to bring these contracts into the Application. These contracts can be brought in as

'ENTERED', which would mean the process owners would have to submit the contract for approval using Contracts Approval Workflow.

OKS_CONTRACTS_PUB.Create_Service_Line

This API is used to create Contract Service Line for each contract header. The API is called using P_K_Line_Rec, P_Contact_Tbl and P_Line_Sales_Crd_Tbl. Again depending on your needs and available information you would populate each of these parameters. However if no information were available for the P_Contact_Tbl and P_Line_Sales_Crd_Tbl, it would be okay to let them be NULL from conversion perspective. The key aspect of service line API is to understand the relationships between the shipping entity and the billing entity. Especially if the item is serviceable with a usage component and has been leased out using a third party company. Also important is to attach this item with the appropriate Install Base record if Install Base application is in use.

OKS_CONTRACTS_PUB.Create_Covered_Line

This API is used to create Covered Lines for each service line you create. Before you call this API you are required to set values for P_K_Covered_Rec and P_Price_Attribs_In. Depending on your pricing needs values for P_Price_Attribs_In will have to be set. If there is no complex pricing requirements it is okay to leave the value for this record type null. Values of certain columns of P_K_Covered_Rec should be same as P_K_Line_Rec e.g. value for PERIOD in the record should be same as the value for the P_K_Line_Rec.Usage_Type or value for Line_Renewal_Type should be either FUL/DNR/KEP.

Conversion Logic

The process flow diagram (screenshot 2) could be used as a guide to assist in conversion of contracts.

About the Author

OKS_CONTRACTS_PUB.Create_Contract_Header:

This API is used to create contract header information. The API is called using P_K_Header_Rec, P_Header_Contacts_Tbl, P_Header_Sales_Crd_Tbl and P_Header_Articles_Tbl. Depending on your conversion needs and the available information in the legacy application you would be required to populate each of these in-parameters. However if no information is available for sales credit or contract articles you could leave the P_Header_Sales_Crd_Tbl and P_Header_Articles_Tbl NULL. However it is a must that you populate the P_K_Header_Rec with all the values. At least 1 contact information for the vendor should be created in P_Header_Contacts_Tbl. The value for the contact_object_code would be 'OKX_SALESPERS'. If you need to create additional parties on the contract header depending on your billing requirements and other entity relationships you may have to call Okc_Contract_Party_Pub.Create_K_Party_Role. Als o you would need to determine in advance how do you wish to bring these contracts into the Application. These contracts can be brought in as 'ENTERED', which would mean the process owners would have to submit the contract for approval using Contracts Approval Workflow.

OKS_CONTRACTS_PUB.Create_Service_Line

This API is used to create Contract Service Line for each contract header. The API is called using P_K_Line_Rec, P_Contact_Tbl and P_Line_Sales_Crd_Tbl. Again depending on your needs and available information you would populate each of these parameters. However if no information were available for the P_Contact_Tbl and P_Line_Sales_Crd_Tbl, it would be okay to let them be NULL from conversion perspective. The key aspect of service line API is to understand the relationships between the shipping entity and the billing entity. Especially if the item is serviceable with a usage component and has been leased out using a third party company. Also important is to attach this item with the appropriate Install Base record if Install Base application is in use.

OKS_CONTRACTS_PUB.Create_Covered_Line

This API is used to create Covered Lines for each service line you create. Before you call this API you are required to set values for P_K_Covered_Rec and P_Price_Attribs_In. Depending on your pricing

needs values for
P_Price_Attribs_In will have to be set. If there is no complex pricing requirements it is okay to leave the value
for this record type null. Values of certain columns of P_K_Covered_Rec should be same as P_K_Line_Rec e.g.
value for PERIOD in the record should be same as the value for the P_K_Line_Rec.Usage_Type or value for
Line_Renewal_Type should be either FUL/DNR/KEP.

Contract Header Creation : For creating contract header API is:*OKS_CONTRACTS_PUB.CREATE_CONTRACT_HEADER*

Line Creation : You creating lines you can this api *OKS_CONTRACTS_PUB.CREATE_SERVICE_LINE*

Sub Line Creation : This is important when underline item is IB track able then we can create covered product sub line with IB link established. API you can use is: *OKS_CONTRACTS_PUB.CREATE_COVERED_LINE*

UPDATING CONTRACT HEADER - You can use okc_contract_pub.update_contract_header API to Update contract Header

```
okc_contract_pub.update_contract_header
( p_api_version => 1.0,
p_init_msg_list => okc_api.g_true,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
p_restricted_update => okc_api.g_false,
p_chrv_tbl => l_chrv_tbl_in,
x_chrv_tbl => l_chrv_tbl_out);
```

UPDATING CONTRACT LINE - You can use okc_contract_pub.update_contract_line API to Update contract Lines

```
okc_contract_pub.update_contract_line (
p_api_version => 1,
p_init_msg_list => OKC_API.G_TRUE,
p_restricted_update => OKC_API.G_FALSE,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
p_clev_tbl => l_clev_tbl,
x_clev_tbl => l_x_clev_tbl
);
```

CASCADE DATE -You can use oks_bill_sch.cascade_dates_sll to cascase date in billing agreement(schedule). You just need to pass only the contract line ID

```
oks_bill_sch.cascade_dates_sll
(
p_top_line_id => <OKS Contract Line ID>,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data
);
```

DEFAULTING ATTRIBUTES FROM LINES TO SUBLINES : You can use below API for daulting attributes

```
oks_attr_defaults_pvt.default_lines_to_sublines
(
lines_sublines_tbl => l_line_table,
x_return_status => l_return_status,
x_msg_tbl => l_msg_tbl
);
```

CREATING CONTACT AT CONTRACT HEADER LEVEL :You can use *okc_contract_party_pub.create_contact to create contact at header.*

```
okc_contract_party_pub.create_contact (
p_api_version => 1,
p_init_msg_list => fnd_api.g_true,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
p_ctcv_rec => l_ctcv_rec,
x_ctcv_rec => l_x_ctcv_rec
);
```

UPDATING CONTACT AT CONTRACT HEADER LEVEL :You can use *okc_contract_party_pub.update_contact to create contact at header.*

```
okc_contract_party_pub.update_contact (
p_api_version => 1,
p_init_msg_list => OKC_API.G_FALSE,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
p_ctcv_rec => l_ctcv_rec,
x_ctcv_rec => l_x_ctcv_rec
);
```

CREATING CONTACT E-MAIL ADDRESS AT CONTRACT HEADER LEVEL :You can use *OKS_EXTWAR_UTIL_PUB.Contact_Point to create contact at header.*

```
OKS_EXTWAR_UTIL_PUB.Contact_Point (
p_api_version => 1,
p_init_msg_list => 'T',
P_commit => 'F',
P_contact_point_rec => l_cpoint_rec,
x_return_status => l_return_status,
x_msg_count => l_msg_count,
x_msg_data => l_msg_data,
x_contact_point_id => l_x_contact_point_id
);
```

Sales Credit : For creating sales credit separately, we will have to use API:*OKS_SALES_CREDIT_PUB.INSERT_SALES_CREDIT*.