**Instructivo de Despliegue**

AMDOCS API COMMON WEB SERVICES/ AMDOCS API COMMON EJB

Tema: ­­­ Instructivo de Despliegue

*Versión 1.0*

Fecha: 08-08-2017

Contenido

[1. Introducción - 3 -](#_Toc493494769)

[1.1. Objetivo - 3 -](#_Toc493494770)

[1.2. Alcance - 3 -](#_Toc493494771)

[2. Historial de Revisiones - 3 -](#_Toc493494772)

[3. Documentos Adjuntos (N/A) - 3 -](#_Toc493494773)

[4. Descripción General - 4 -](#_Toc493494774)

[5. Instalación Web Services - 4 -](#_Toc493494775)

[5.1. Componentes - 4 -](#_Toc493494776)

[5.2. Archivos de configuración - 4 -](#_Toc493494777)

[6. Uso de los archivos de configuración del web services - 6 -](#_Toc493494778)

[6.1. Despliegue del web services - 7 -](#_Toc493494779)

[7. Actualización de la aplicación - 14 -](#_Toc493494780)

[7.1. Para la actualización de la aplicación debemos cambiar la configuración del dominio, entonces clickeamos “Lock & Edit” en el “Change Center”: - 14 -](#_Toc493494781)

[7.2. Sobre “Deployments” en la consola de administración, seleccionamos la aplicación, y clickeamos “Update” - 15 -](#_Toc493494782)

[7.3. En la siguiente página, seleccionamos el path de la nueva versión o mismo el redespliegue del mismo artefacto: - 15 -](#_Toc493494783)

[7.4. En “Change Center”, click “Active Changes” : - 16 -](#_Toc493494784)

[8. Despliegue de AMDOCS API COMMON EJB - 16 -](#_Toc493494785)

[9. Despliegue de una nueva versión AMDOCS API COMMON EJB - 16 -](#_Toc493494786)

[10. Configuración del archivo de log - 18 -](#_Toc493494787)

[11. Configuración de los JDNI - 20 -](#_Toc493494788)

## Introducción

* 1. Objetivo

El Objetivo del presente documento es describir los pasos necesarios para realizar el despliegue de la aplicaciones AMDOCS API COMMON WEB SERVICES y AMDOCS API COMMON EJB.

* 1. Alcance

El presente documento tiene como alcance describir los pasos necesarios para la instalación de: amdocs-api-common-web-services-ear y amdocs-api-common-ejb-ear.

El presente documento solo especifica como instalar los componentes mencionados en el ambiente productivo de Telefónica Argentina.

## Historial de Revisiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fecha | Versión | Autor | Descripción | Revisado Por |
| *08-08-17* | 1.0 | Carlos Centurion | Creación del Documento |  |
| *18-09-17* | 2.0 | Carlos Centurion | Se agrega detalle al proceso de despliegue. | Carlos Centurion |
| *19-09-17* | 2.1 | Carlos Centurion | Se modifico la configuración de las opciones para JAVA\_OPTIONS | Nestor Spedalieri |

## Documentos Adjuntos (N/A)

Se acompañan los siguientes documentos para la comprensión del/de los requerimiento/s incluido/s en el presente.

|  |  |
| --- | --- |
| Título | Archivo |
|  |  |
|  |  |
|  |  |

## Descripción General

El proyecto está compuesto por la siguiente estructura de componentes a ser instalados:

* amdocs-api-common-web-services-ear
* amdocs-api-common-ejb-ear

El primer componente es el web services en si y es el que se entrega al cliente, mientas que el segudo es un aplicación que emula un backend implementando los contratos de los “client kits” para cada módulo del core de AMDOCS.

## Instalación Web Services

amdocs-api-common-web-services-ear es una aplicación compuesta por dos modulos que representan las unidades de negocio de AMDOCS: “abp” y “oms”.

* 1. Componentes

Los módulos están empaquetados en un archivo en formato ear (Enterprise Application aRchive) que contiene las dos aplicaciones del tipo war (Web application Archive) además de los Client kit’s de amdocs y dependencias varias.

* 1. Archivos de configuración
     1. abp\_ws.properties

Este archivo contiene las propiedades que setean el funcionamiento del modulo abp

| **ParamName** | **ParamValue** | **Description** |
| --- | --- | --- |
| environment.type | credentials/  ticket | Este parámetro configura el modo login a los servicios de AMDOCS según los valores :   * ticket : la llamada remota utiliza el ticket generado por el servicio de usuario ASM * credentials : la llamada remota se realiza mediante user/password |
| direct.login.wrapper.userId | XXXXXXXXXXXXX | Parámetro para la geracion del ticket de ASM contra el login de AMDOCS. |
| direct.login.wrapper.password | XXXXXXXXXXXXX | Parámetro para la geracion del ticket de ASM contra el login de AMDOCS. |
| direct.login.wrapper.appId | XXXXXXXXXXXXXXXX | Parámetro para la geracion del ticket de ASM contra el login de AMDOCS. |
| direct.login.wrapper.env | XXXXXXXXXXXXXXXX | Parámetro para la geracion del ticket de ASM contra el login de AMDOCS. |
| java.naming.factory.initial | weblogic.jndi.WLInitialContextFactory | Este valor setea el contexto sobre en el servidor weblogic para las invocaciones de métodos remotos. |
| java.naming.provider.url | t3://{ip}:{port} | url de conexión al backend de AMDOCS para la invocación de los métodos remotos |
| java.naming.security.credentials | na | Este parámetro seteado con valor “na” es utilizado para invocar métodos remotos mediante el ticket de ASM.  Utilizado solo cuando environment.type=ticket. |
| java.naming.security.principal | xxxxxxx | Solo utilizado cuando environment.type=credentials. Contendra el usuario que se necesita para logearse al backend de AMDOCS |
| flag.local.context | true/false | Determina si la invocación de los métodos remotos es local o remota |
| use.encryption | true/false | Determina si los pares user/password de este archivo deben ser desencriptados al momento de la invocación de los métodos remotos. |
| CREDENTIALS.java.naming.factory.initial | weblogic.jndi.WLInitialContextFactory | Utilizado cuando el environment.type=credentials. |
| CREDENTIALS.java.naming.provider.url | t3://{ip}:{port} | Utilizado cuando el environment.type=credentials. Url del provider para la invocación de los métodos remotos. |
| CREDENTIALS.java.naming.security.principal | xxxxxxxxxxxxxxxx | Utilizado cuando el environment.type=credentials. Este parametro representa el user con el cual genera el contexto para las invocaciones remotas. |
| CREDENTIALS.java.naming.security.credentials | xxxxxxxxxxxxxxxx | Utilizado cuando el environment.type=credentials. Este parametro representa el password con el cual genera el contexto para las invocaciones remotas. |
| ASM3.java.naming.factory.initial | weblogic.jndi.WLInitialContextFactory | Este valor setea el contexto sobre en el servidor weblogic para las invocaciones de métodos remotos de ASM. |
| ASM3.java.naming.provider.url | t3://{ip}:{port} | Url del provider para la invocación de los métodos remotos de ASM. |
| ASM3.java.naming.security.principal | xxxxxxxxxxxxxxxxxx | User con el cual se genera el contexto para la llamada remota a ASM |
| ASM3.java.naming.security.credentials | xxxxxxxxxxxxxxxx | Password con el cual se genera el contexto para la llamada remota a ASM |
| ASM3.flag.local.context | true/false | Determina si la invocación de los métodos remotos es local o remota |
| asm.jndi.path | amdocsBeans.ASM3LoginServicesHome | JDNI utilizado para el lookup hacia ASM |

* + 1. oms\_ws.properties

El archivo de configuacion para el modulo de “oms” es exactmente igual al descripto en el cuadro anterior 5.2.1.

## Uso de los archivos de configuración del web services

Los modulos “abp” y “oms” dependen del set de dos variales de configuracion que deben cargarse al classpath en tiempo de runtime :

* -D\_ABP\_WEB\_SERVICES\_CONFIG
* -D\_OMS\_WEB\_SERVICES\_CONFIG

Una configuracion tipica dentro de un servidor weblogic (/u01/domains/practia\_variables.sh) seria por ejemplo:

**set JAVA\_OPTIONS = “${JAVA\_OPTIONS} -D\_ABP\_WEB\_SERVICES\_CONFIG=** **/u01/app/oracle/config /abp\_ws.properties”**

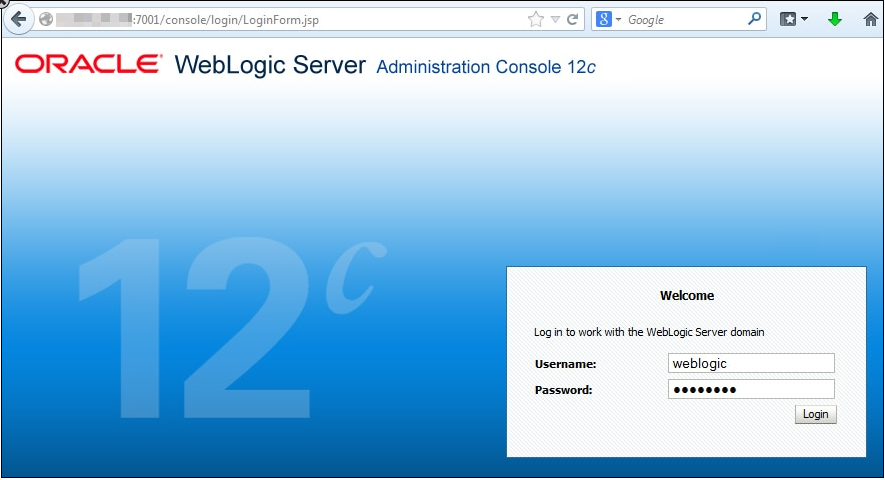
**set JAVA\_OPTIONS = “${JAVA\_OPTIONS} -D\_OMS\_WEB\_SERVICES\_CONFIG=** **/u01/app/oracle/config /oms\_ws.properties”**

Estas dos variables son obligatorias para el correcto funcionamiento de la aplicativo.

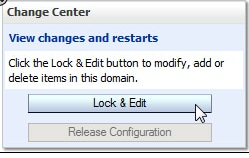
* 1. Despliegue del web services

A continuación, se describe el proceso de despliegue del web services

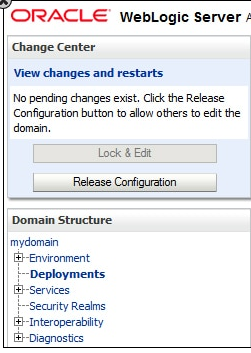
* + 1. En la pantalla de bienvenida, logearse con user/pass a la consola de administración



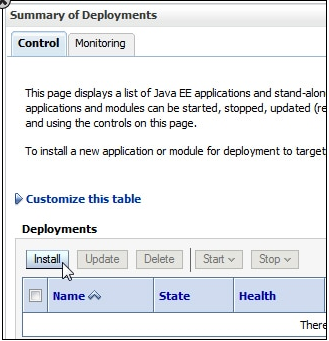
* + 1. Para el despliegue de una aplicación debemos cambiar la configuracion del dominio. En el “Change Center” y clickear **“Lock & Edit”:**

****

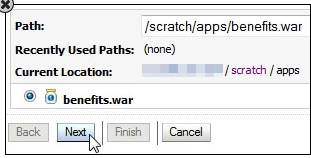
* + 1. Bajo la esctrucura del dominio , click “**Deployments”:**



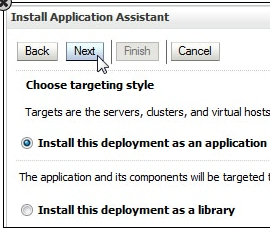
* + 1. Sobre la derecha, debajo de la tabla “Deployments”, click **Install :**



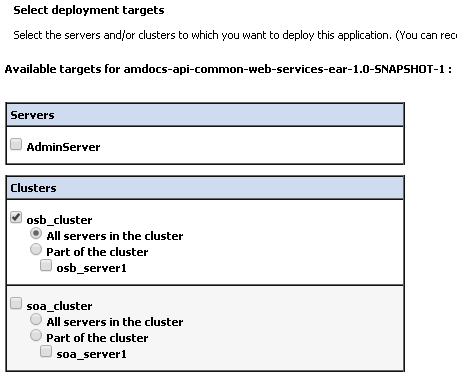
* + 1. Localizar el archivo a deplegar :



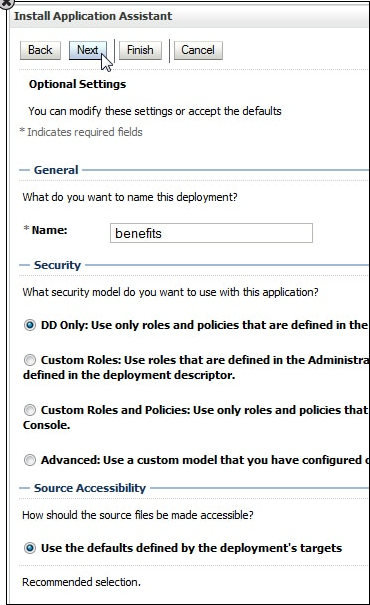
* + 1. En la sig pantalla, asegurarse que la instalación de este despliegue es una “**aplicación”**, el cual debe seleccionarse:



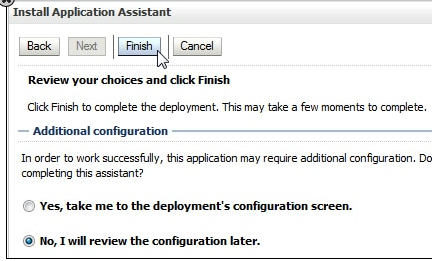
* + 1. En la sig. pantalla deberá seleccionar el server o el manejado donde correrá la aplicación:



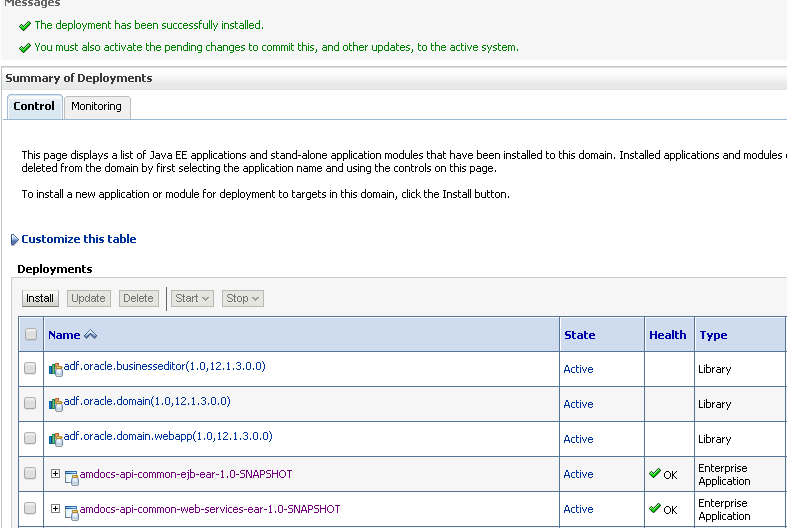
* + 1. En la siguiente pantalla, mantener todos los valores por default y click “**Next”**



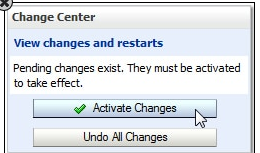
* + 1. En la siguiente pantalla, seleccionar “**No, I will review the configuration later”,** y clickear **“Finish”**



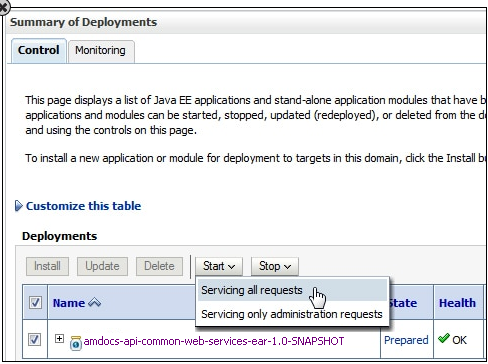
* + 1. El mensaje indica que el despliegue ha instalado un aplicativo, pero debemos activar estos cambios:



* + 1. Se activan los cambios en el “**Change Center”**



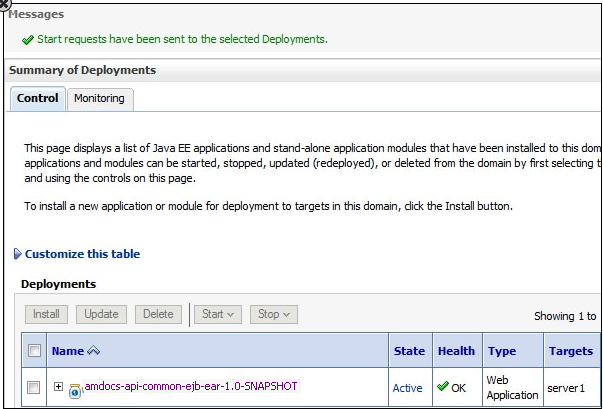
* + 1. En la tabla “Deployments” seleccionar el check box a la izquierda de “amdocs-api-common-web-services-ear” y seleccionar “**Servicing all requests**” en la lista Start :



* + 1. Clickear **“Yes”** para continuar.



* + 1. Un mensaje indica que todos los request fueron enviados. El stado de la aplicación se transforma en “active”, esto significa que la aplicación es disponible para ser accedida.



* + 1. Una vez finalizada la instalación deberíamos poder ver los sig mensajes



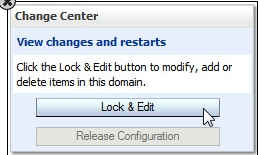
Y por ultimo



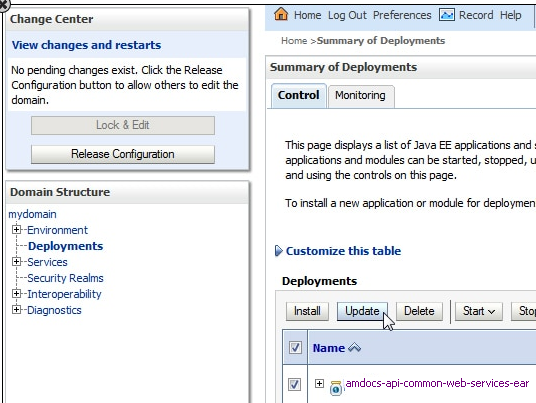
## Actualización de la aplicación

Al actualizar la aplicación, ya sea por una nueva versión para despliegue o por “redeploy” de la misma debemos ejecutar los sig. pasos:

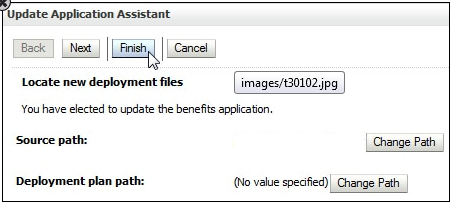
* 1. Para la actualización de la aplicación debemos cambiar la configuración del dominio, entonces clickeamos “**Lock & Edit**” en el “**Change Center**”:

****

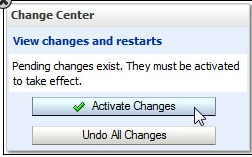
* 1. Sobre “**Deployments**” en la consola de administración, seleccionamos la aplicación, y clickeamos “**Update**”



* 1. En la siguiente página, seleccionamos el path de la nueva versión o mismo el redespliegue del mismo artefacto:



* 1. En “Change Center”, click “Active Changes” :



**NOTA:** Después de los cambios son activados, el estado de la aplicación es “Active”. No se necesita un “Start” manual.

## Despliegue de AMDOCS API COMMON EJB

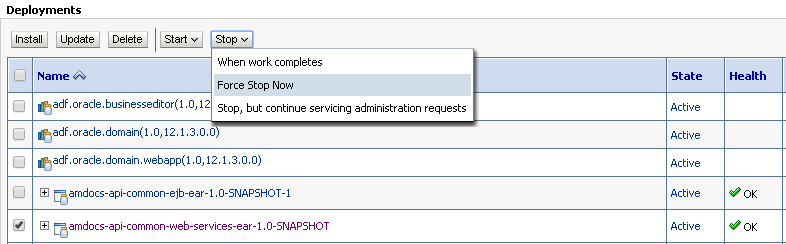
La aplicación AMDOCS API COMMON EJB tiene la funcionalidad de hacer de backend en los ambientes controlados de test. Esta aplicación solo es utilizada en los ambientes de practia y no debe considerarse como entregable al cliente.

Esta aplicación expone los JDNI a consumir por el servicio web, implementando los ck’s enviados por AMDOCS.

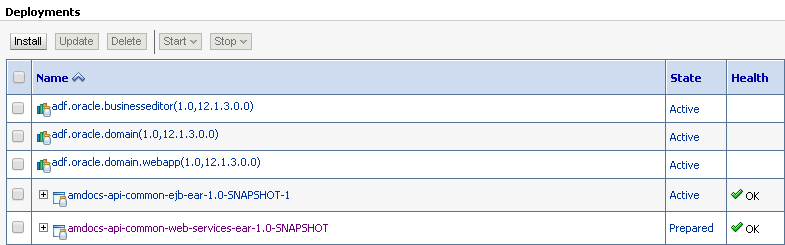
## Despliegue de una nueva versión AMDOCS API COMMON EJB

Ante un despliegue de una versión del artefacto debemos respetar lo sig.:

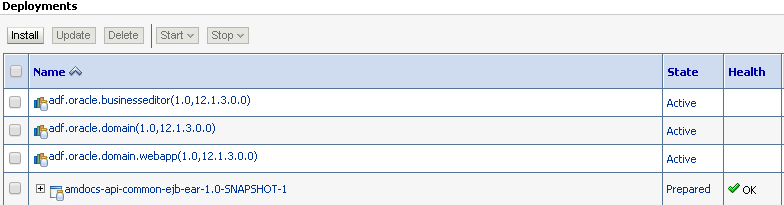
9.1 Los web’s services utilizan el artefacto “AMDOCS API COMMON EJB” como el backend a realizar las llamadas remotas, por ende, si los dos artefactos ya estuvieran desplegados y activos debemos “stopear” los web services:



Que quedaran con el estado “Prepared”

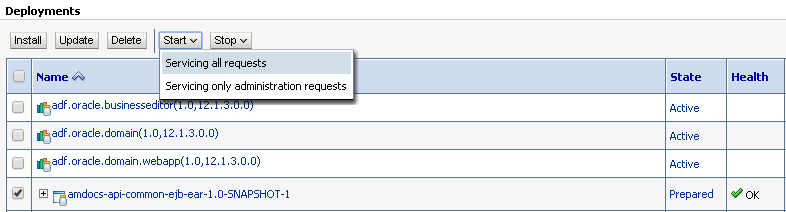


9.2 Una vez que dejamos a los servicios en estado “**Prepared**” podemos “Stopear” los ejb’s de la misma manera:

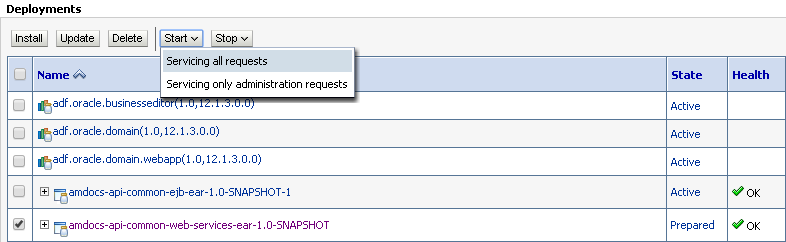


Y Procedemos a la actualización detallada en la sección 7.

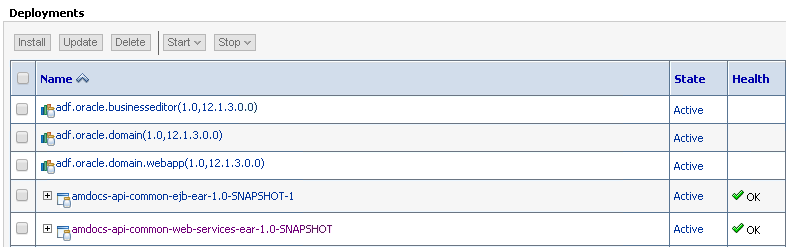
9.3 Una vez realizada el redespliegue del artefacto, deben ser los ejb’s los primeros en ser “Starteado”:



Dejándolo “**Active**” para dar paso al “Starteado” del web services:



Y finalmente los dos activos:



## Configuración del archivo de log

La aplicación utiliza Log4j2 para la traza del funcionamiento de la aplicación. Esta api se configura con un archivo log4j2.xml y agregándola al path en tiempo de runtime sobre el archivo /u01/domains/practia\_variables.sh:

**JAVA\_OPTIONS "${JAVA\_OPTIONS} -Dlog4j.configurationFile=/u01/app/oracle/config/log4j2.xml"**

**export JAVA\_OPTIONS**

Se adjunta un ejemplo de la configuración del archivo log4j2.xml al momento del reléase de este documento:

Una vez configurado este archivo, en tiempo de runtime, en el dominio en donde se está ejecutando la aplicación generara una carpeta llamada “amdocs-api-common-web-services-log” en cual contendrán los sigs. archivos:

* amdocs-api-common-abp-ws.log (traza todas las entradas en nivel “info” del módulo ABP)
* amdocs-api-common-abp-ws-err.log (traza todas las entradas en nivel “error” del módulo ABP)
* amdocs-api-common-oms-ws.log (traza todas las entradas en nivel “info” del módulo OMS)
* amdocs-api-common-oms-ws-err.log (traza todas las entradas en nivel “error” del módulo OMS)
* asm3.log (traza todo el funcionamiento del servicio ASM)

## Configuración de los JDNI

El web services toma de cada uno de sus archivos properties (abp\_ws y oms\_ws) el path “JDNI” para la invocación de cada llamada remota.

Al implementar un nuevo servicio se deberá agregar en el final del archivo el nuevo JDNI path, si por ejemplo, damos de alta un servicio denominado ARIBUserServices se debe agregarlo al final del archivo de la sig. manera jdni.path.new\_services.

ej: en el archivo abp\_ws.properties agregar

**jdni.path. ARIBUserServices=JDNI\_VALUE**

Donde JDNI\_VALUE es el JDNI provisto por AMDOCS.

Fin del Documento



