



JUNTA DE ANDALUCIA

Soporte Platina para los servicios de Testing. PlatinaV3

Manual de Instalación API Manager

Versión: 0100

Fecha:19/01/2015

[3.3.0.0]

Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este documento sin el previo consentimiento expreso y por escrito de la Junta de Andalucía.

| | |
|--|--|
|  Soporte Platina para los servicios de Testing. PlatinaV3 Manual de Instalación API Manager | Consejería de Hacienda y Administración Pública |
|--|--|

HOJA DE CONTROL

| | | | |
|---------------------|--|----------------------------|----|
| Organismo | Consejería de Hacienda y Administración Pública | | |
| Proyecto | Soporte Platina para los servicios de Testing. PlatinaV3 | | |
| Entregable | Manual de Instalación API Manager | | |
| Autor | Unidad de Proyectos de Sistemas Horizontales | | |
| Aprobado por | | Fecha Aprobación | |
| | | Nº Total de Páginas | 46 |

REGISTRO DE CAMBIOS

| Versión | Causa del Cambio | Responsable del Cambio | Fecha del Cambio |
|----------------|-------------------------|--|-------------------------|
| 0100 | Versión inicial | Unidad de Proyectos de Sistemas Horizontales | 19/01/2015 |

CONTROL DE DISTRIBUCIÓN

| Nombre y Apellidos |
|--|
| Antonio Blanco Morales (CHAP) |
| Francisco Rodríguez Corredor (SANDETEL) |
| Juan Antonio Campano Berlanga (SANDETEL) |



| | |
|---|-----------|
| INTRODUCCIÓN..... | 5 |
| 1.1 Objeto..... | 5 |
| 1.2 Alcance..... | 6 |
| DESCRIPCIÓN DEL SISTEMA..... | 7 |
| 2.1 Componentes fundamentales..... | 7 |
| 2.2 Relación con otros sistemas..... | 8 |
| RECURSOS HARDWARE..... | 9 |
| 3.1 Servidores..... | 9 |
| 3.1.1 Interfaces de red..... | 10 |
| 3.2 Conectividad..... | 11 |
| 3.3 Restricciones..... | 12 |
| RECURSOS SOFTWARE..... | 13 |
| 4.1 Matriz de certificación..... | 13 |
| 4.1.1 URLs de acceso al clúster AM..... | 13 |
| INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE..... | 15 |
| INSTALACIÓN DEL SISTEMA..... | 18 |
| 6.1 Requisitos previos..... | 18 |
| 6.2 Procedimiento de instalación..... | 18 |
| 6.2.1 Desempaquetado del componente..... | 19 |
| 6.2.1.1 Descompresión del paquete:..... | 19 |
| 6.2.1.2 Creación del enlace simbólico:..... | 19 |
| 6.2.2 Configuración AM..... | 19 |
| 6.2.1.3 Configuración clúster..... | 20 |
| | 20 |
| Configuración Multicasting:..... | 20 |
| Configuración WKA:..... | 21 |
| 6.2.1.4 Configuración registro..... | 22 |
| 6.2.1.5 Configuración Carbon:..... | 24 |
| 6.2.1.6 Configuración autenticación y almacén de usuarios:..... | 1 |
| 6.2.1.7 Configuración de la caché:..... | 2 |
| 6.2.1.8 Copiado del driver OJDBC al directorio lib..... | 3 |
| 6.2.1.9 Ajustes de valores de la JVM:..... | 3 |



| | |
|--|----|
| 6.2.1.10 Habilitar sincronización nodo..... | 4 |
| En el nodo Master:..... | 4 |
| En los nodos Slaves:..... | 4 |
| VERIFICACIÓN DEL PROCESO DE INSTALACIÓN..... | 6 |
| 7.1 Verificación 1. Componente AM MASTER..... | 6 |
| 7.2 Verificación 2. Componente AM en SLAVE1..... | 1 |
| 7.3 Verificación 3. Verificación de puertos a la escucha en los nodos..... | 2 |
| Parada y arranque del servicio AM..... | 3 |
| 8.1 Monitorización de ficheros de logs..... | 3 |
| Supervisión del servicio..... | 4 |
| 9.1 Pruebas Generales..... | 4 |
| 9.2 Pruebas específicas..... | 5 |
| ANEXOS..... | 8 |
| Anexo I: Tunning del sistema operativo..... | 8 |
| /etc/sysctl.conf file..... | 8 |
| /etc/security/limits.conf..... | 8 |
| Anexo II: Tunning del AM..... | 9 |
| Anexo III. Configuración de URLs frontales e interfaz de servicio..... | 10 |
| Anexo IV: Ajustes de memoria de la Maquina Virtual Java..... | 12 |
| Anexo V: Configuración de proxy para acceso a internet..... | 13 |
| Anexo VI: Demonización del servicio..... | 14 |

1 INTRODUCCIÓN

1.1 Objeto

El contenido del presente documento describe los pasos necesarios para la instalación del componente API Manager de la plataforma PLATINA V3.

A modo informativo y de forma resumida, describimos los productos implicados en la plataforma PLATINA V3:

| Módulo | Descripción | Abreviatura |
|--|--|-------------|
| Bus de servicio | Se trata de Bus de Servicio de Empresa (WSO2 ESB) de código abierto, rápido, ligero y fácil de usar. Está basado en el motor de mediación Apache Synapse. Permite principalmente el enrutamiento de mensajes, la mediación, transformación, trazabilidad, programación de tareas, enrutamiento por error, balanceo de carga y otras funcionalidades más avanzadas. | ESB |
| Repositorio | Es una implementación de código abierto (WSO2 Government Registry) de un registro-repositorio SOA. Proporciona un completo repositorio de meta-datos y permite la gestión completa de versiones de servicios, la gestión de su ciclo de vida, un modelo de administración basado en usuarios, roles y permisos, y otras características, tales como el etiquetado, la clasificación y comentarios. Todo gestionable a través de la interfaz de usuario basada en web. | GREG |
| Servidor de Identidad | Se trata de un componente para la gestión de identidad y de permisos basado en código abierto (WSO2 Identity Server) y que tiene soporte para OpenID, tarjetas de información (Information Card), XACML y SAML 2.0. | IS |
| Monitor de actividad de negocio | BAM es el componente basado en software libre (WSO2 BAM) de la plataforma que se encarga de la monitorización de métricas SOA, de indicadores clave de negocio que hayan sido definidos y cualquier otro parámetro que sean medibles. La información que ofrece este herramienta tiene como objetivo principal cubrir las necesidades de monitorización y comprensión de las actividades que se desarrollan en el organismo, tanto desde un punto de vista de negocio como desde el ámbito de la tecnología de la información (TI). | BAM |

| | | |
|--|---|--|
|  JUNTA DE ANDALUCÍA | Soporte Platina para los servicios de Testing. PlatinaV3 Manual de Instalación API Manager | Consejería de Hacienda y Administración Pública |
|--|---|--|

| Módulo | Descripción | Abreviatura |
|--------------------------------------|--|-------------|
| Servidor de Gadgets | <p>El servidor de Gadget (GS) es un componente que actúa como capa de presentación de la plataforma y nos permite extender la plataforma Platina con nuevas funcionalidades a medida.</p> <p>En esta versión de platina, se ofrece la Consola de Administración de Platina a través de este componente.</p> | GS |
| Servicio de Colas de Mensajes | <p>Este componente es un servicio de colas de mensajes basado en software libre (Apache ActiveMQ) que implementa JMS 1.1 de la especificación J2EE 1.4</p> <p>El objetivo principal del componente es, por un lado, dar soporte JMS a Platina a nivel de transporte, y por otro, posibilitar el almacenamiento temporal de mensajes, con diversos propósitos, que pasan a través de Platina.</p> | AMQ |
| Message Broker | <p>Componente que ofrece soporte para arquitecturas dirigidas por eventos. Este componente está basado en Apache Qpid que es un referente de hoy día en la ingeniería de mensajes. MB permite el acceso unificado y simultáneo a la mensajería mediante diversos protocolos, tales como JMS, WS-Eventing, Amazon SQS y AMQP.</p> | MB |
| API Manager | <p>Portal para facilitar la publicación y el consumo de servicios, así como la gestión de los mismos.</p> | AM |

1.2 Alcance

Este documento contiene una descripción del sistema, los recursos hardware y software necesarios así como una descripción del proceso de instalación, configuración y verificación de cada uno de los componentes que lo forman. Este documento va dirigido al personal técnico designado por la Consejería para la realización de la instalación de este sistema en cualquiera de los entornos disponibles en su organización.

2 DESCRIPCIÓN DEL SISTEMA

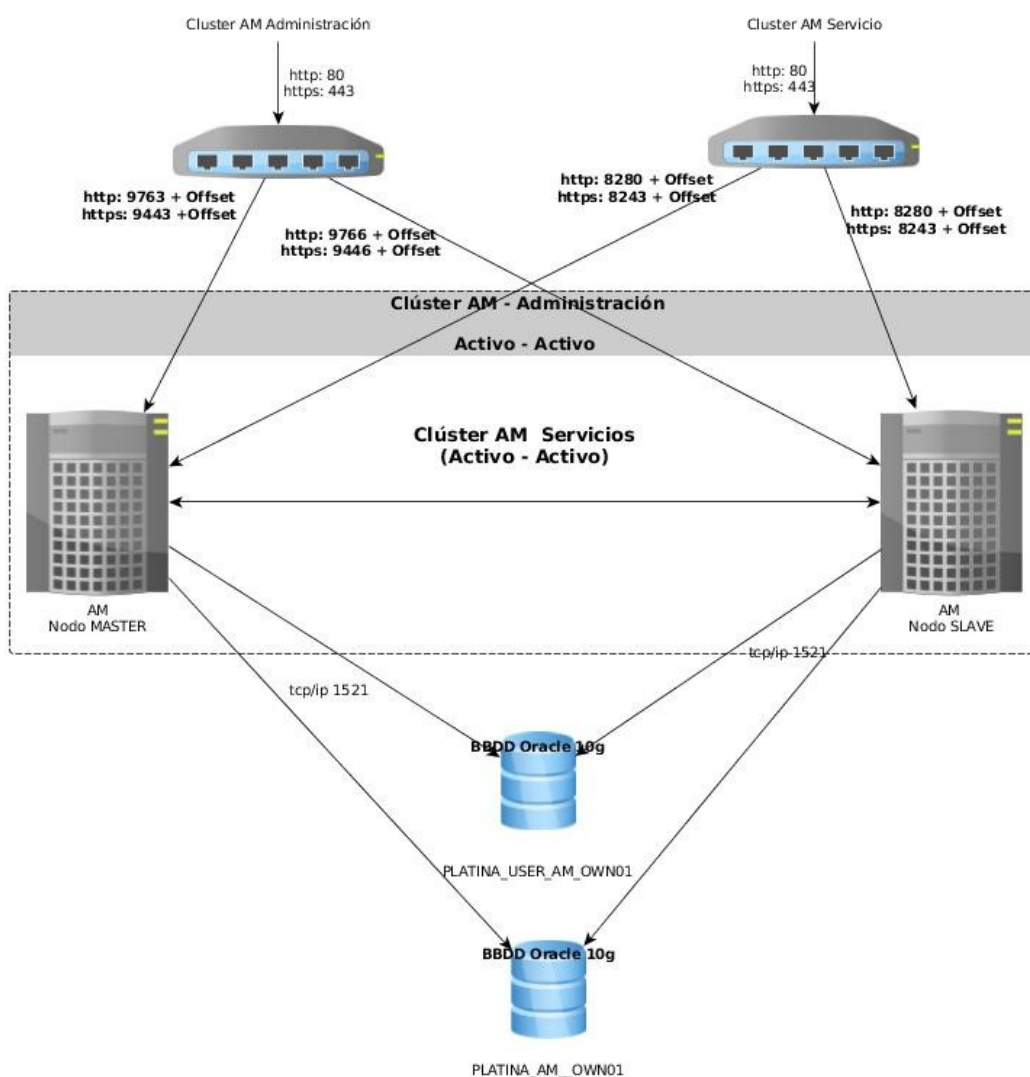
2.1 Componentes fundamentales

El componente AM de platina V3 va a estar configurado en modo clúster por dos nodos: MASTER Y SLAVE1. Cada uno de los nodos debe estar desplegado en un servidor distinto.

Se trata de un clúster en en dos niveles:

- Clúster de servicios de administración: ACTIVO-ACTIVO.
- Clúster a nivel de servicios desplegados en el AM: ACTIVO-ACTIVO.

En el siguiente diagrama se puede ver la arquitectura del clúster de Platina V3.





2.2 Relación con otros sistemas

El componente se puede integrar con BAM para hacer recolección de métricas de uso.

Es posible integrar con IS para aplicar políticas de autenticación y autorización de acceso a recursos.

El componente AM puede hacer uso del registro centralizado.

Es posible realizar las integraciones anteriormente descritas, pero en este documento, se tratará la instalación del componente API Manager de forma aislada del resto de componentes. Se proporcionarían instrucciones adicionales específicas para cada integración, en caso de ser necesarias.

3 RECURSOS HARDWARE

3.1 Servidores

En la siguiente ilustración se muestran los componentes hardware requeridos para el componente AM de Platina V3:

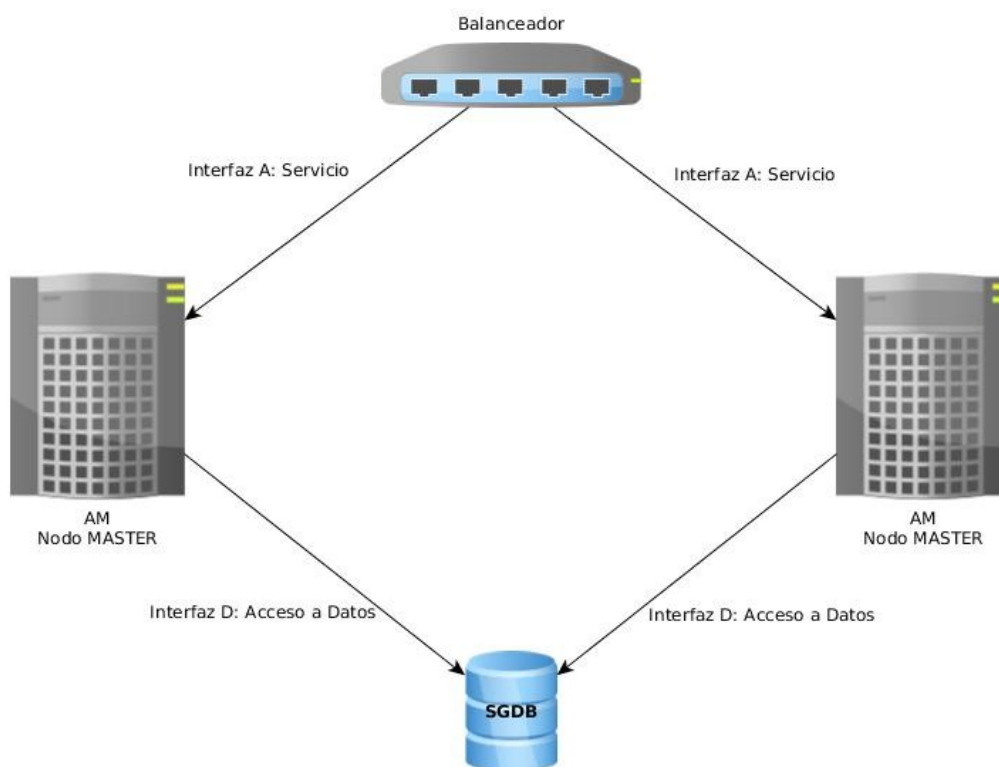


Ilustración 1: Arquitectura Hardware del Componente AM de Platina V3

Los requerimientos técnicos de los nodos de la plataforma se especifican en la siguiente tabla:

Nodo MASTER

| Dato | Valor mínimo | Valor recomendado |
|-----------------------|------------------------|--------------------------|
| Procesador | Intel Pentium D 2.8GHz | >> |
| Memoria RAM | 4.00 GB | >> |
| Tamaño Almacenamiento | 50 GB | >> |

Nodo SLAVE1

| Dato | Valor mínimo | Valor recomendado |
|-----------------------|------------------------|--------------------------|
| Procesador | Intel Pentium D 2.8GHz | >> |
| Memoria RAM | 4.00 GB | >> |
| Tamaño Almacenamiento | 50 GB | >> |

3.1.1

Interfaces de red

Se han identificado cuatro interfaces de red para configurar de forma óptima el clúster de Platina V3.

Nos obstante, debemos aclarar que no es obligatorio disponer todas las interfaces operativas, aunque sí muy recomendable.

En la siguiente tabla se describen cada una de las interfaces que podemos configurar:

Interfaces de Acceso a Red

| Interfaz | Tipo de comunicación | Comentario |
|--------------------|---|-------------------|
| A: Servicio | <p>Esta interfaz principal del nodo:</p> <ul style="list-style-type: none"> Se reciben todas las peticiones de servicio en el AM. Permite que el AM se comunique con otros servicios que puedan ser consumidos. | Es obligatoria |

| | | |
|--------------------------|---|---|
| | <ul style="list-style-type: none"> Ofrece el acceso a la consola de administración de AM | |
| B: multicast | Esta interfaz es utilizada por los nodos del clúster para descubrir de forma automática nuevos nodos en el clúster. | No es obligatoria |
| C: clúster | A través de esta interfaz, los nodos que forman el clúster comparten la información. | No es obligatoria. |
| D: Acceso a Datos | Interfaz utilizada por los nodos del clúster para el acceso a la base de datos | No es obligatoria, pero muy recomendable. |

3.2 Conectividad

En la siguiente tabla se indica cual es la interfaz recomendada para cada tipo de comunicación y qué puerto concreto se utiliza. También si es un puerto de entrada o salida.

| Nodo | Entrada | Salida | Interfaz | Servicio |
|---------------------------|---------------|-----------|--------------|-------------------------|
| MASTER/ SLAVE1 | 9443 + Offset | - | A (Servicio) | Administración AM (SSL) |
| | 9763 + Offset | - | A (Servicio) | Administración AM |
| | 8280 + Offset | - | A (Servicio) | Servicios AM (HTTP) |
| | 8243 + Offset | - | A (Servicio) | Servicios AM (HTTPS) |
| | - | NTP (123) | A (Servicio) | Acceso a servidor NTP |
| | - | 1521 | D (Datos) | Acceso a Oracle |



3.3 Restricciones

N/A

4 RECURSOS SOFTWARE

4.1 Matriz de certificación

Seguidamente se recoge la compatibilidad de los componentes de PLATINA V3, con las distintas versiones del software base y componentes externos utilizados.

| Elemento | Versión (es) Soportada (s) |
|-------------------|--|
| AM | WSO2 AM 1.8.0 |
| Platina | Platina v3.3.0 |
| Sistema Operativo | Redhat 5.4. |
| JDK | JDK 1.6.x |
| JCE | Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 6 |
| Oracle | 10g v 10.2.0.1 |
| JDBC Driver | ojdbc14.jar, mysql-connector-java-5.1.34-bin.jar |
| Navegadores | Firefox 3.0.x o superior, Internet Explorer 7 o Google Chrome. |

4.1.1 URLs de acceso al clúster AM

Es este punto estudiamos las URLs de este componente que deben ser procesadas y transformadas a una URL unificada a través de un balanceador o proxy inverso:

1. Administración/configuración del AM: Consola y servicios de administración.

- Urls de acceso son:
 - Consola de administración:
 - **https:<nodo>:9443/carbon**
 - Servicios de administración:
 - **https:<nodo>:9443/services**
 - **http:<nodo>:9763/services**
- Es un clúster ACTIVO/ACTIVO – MASTER/SLAVE.
- El objetivo del clúster:
 - Alta disponibilidad.

Nota: Sólo el MASTER puede hacer operaciones de escritura sobre servicios de administración (cambiar la configuración del clúster).

2. Servicio del AM: Este clúster es el más crítico de los dos debido al número de peticiones que va a recibir. Está configurado aprovechando la configuración clúster que ofrece Axis2. Por defecto está activada la réplica de estados entre los nodos:

- Urls de acceso son:
 - **https:<nodo>:8243/services**
 - **http:<nodo>:8280/services**
- Es un clúster ACTIVO-ACTIVO.
- El objetivo del clúster:
 - Alta disponibilidad.
 - Balanceo de carga.
 - Escalabilidad.

5 INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE

A continuación, se resumen los pasos necesarios para la instalación y configuración del software base. Entendemos por software base, aquel del que depende directamente el sistema para su correcta instalación y funcionamiento.

Dichos componentes se deben instalar en todas las máquinas que componen el sistema, en nuestro caso, nodo MASTER y nodo SLAVE1.

| Sistema Operativo (Linux) | |
|----------------------------------|--|
| Descripción | Sistema Operativo instalado en cada uno de los nodos del clúster. |
| Localización | Soporte corporativo JA |
| Procedimiento de instalación | |
| Paso 1 | Soporte corporativo JA |
| Procedimiento de configuración | |
| Paso 1 | Crear usuario y grupo platina |
| Paso 2 | Creamos el directorio de instalación de la plataforma Platina V3, que denominaremos en adelante [RaízPlatina] . (Por ejemplo podemos usar la ruta /opt/platina) |
| Paso 3 | Hacemos propietario al usuario platina del directorio [Raíz Platina] #chown platina:platina [RaízPlatina] |
| Paso 4 (Opcional) | Tuning del Sistema operativo. Este paso es opcional, aunque sí recomendable para mejorar el rendimiento del sistema. Para más detalles ver apartado (Anexo I. Tuning del Sistema Operativo) |

| JDK 1.6.x | |
|------------------------------|--|
| Descripción | Java Development Kit |
| Localización | Página web del fabricante para su descarga: http://www.oracle.com/technetwork/java/javase/index-137561.html |
| Procedimiento de instalación | |
| Paso 1 | Los pasos necesarios para la instalación de este componente se detallan en la página web del fabricante: |

| | | |
|--------------------------------|--|--|
| | http://www.oracle.com/technetwork/java/javase/index-137561.html | |
| Procedimiento de configuración | | |
| Paso 1 | Los pasos necesarios para la configuración del JDK se detallan en la pagina web del fabricante: http://www.oracle.com/technetwork/java/javase/system-configurations-135212.html | |
| Parámetros a configurar | | |
| Parámetro 1 | JAVA_HOME | Exportar la variable con la ruta donde está instalado el JDK, para que sea visible para el usuario platina . Por ejemplo añadiendo esta linea en el fichero /home/platina/.bash_profile : export JAVA_HOME=[directorio instalación JDK] |
| Parámetro 2 | PATH | Actualizar el valor de la variable con la ruta donde están instalados los binarios del JDK, para que sea visible para el usuario platina . Por ejemplo añadiendo esta linea en el fichero /home/platina/.bash_profile : export PATH=\$PATH:\$JAVA_HOME/bin |

Java Cryptography Extension (JCE)

| | |
|------------------------------|---|
| Descripción | Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 6 |
| Localización | <p>Pagina web del fabricante para su descarga:</p> <p>http://www.oracle.com/technetwork/java/javase/downloads/jce-6-download-429243.html</p> |
| Procedimiento de instalación | |
| Paso 1 | Descargar el fichero: jce_policy-6.zip |
| Paso 2 | Descomprimir el fichero. |
| Paso 3 | Abrir fichero README.txt donde indica las instrucciones de instalación |

Procedimiento de configuración

N/A

Parámetros a configurar

N/A

Cliente NTP

Descripción Servicio cliente NTP

Localización Repositorio de paquetes del Sistema operativo.

Procedimiento de instalación

Paso 1 # yum install ntp

Procedimiento de configuración

Paso 1 Editamos el fichero /etc/ntp.conf:

- El el parámetro "server", ponemos un servidor NTP, por ejemplo ntp.junta-andalucia.es
- Guardamos los cambios en el fichero

Paso 2 Ejecutamos el comando para activar el servicio:

chkconfig ntpd on

Paso 3 Ejecutamos el comando para forzar la sincronización:

#ntpdate ntp.junta-andalucia.es

Paso 4 Arrancamos el servicio:

#service ntpd start

6 INSTALACIÓN DEL SISTEMA

6.1 Requisitos previos

Para iniciar el proceso de instalación del sistema se debe comprobar que se cumplen los siguientes requerimientos previos:

1. Se encuentra correctamente instalados los esquemas de base de datos requeridos por el componente.
2. Correcta instalación del software base descrito en el apartado **5 INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE.**
3. Disponer del paquete entregable: **wso2am-1.8.0.zip** que contiene todo lo necesario para el despliegue.

6.2 Procedimiento de instalación

Como requisito previo, a la instalación, es necesario que se encuentren correctamente instalados los esquemas de base de datos requeridos por el componente.

Para más detalle, se puede consultar el Manual de Instalación de Platina v3 (**MIN_PLATINAV3_Manual_Instalacion_Platina.odt**).

Los esquemas de base de datos requeridos son:

| Esquema | Tablesp ace | Producción | | Pruebas | |
|------------------------|----------------|------------|--------|---------|--------|
| | | Inicial | Máximo | Inicial | Máximo |
| PLATINA_AM_OWN01 | Datos | 500 MB | 10 GB | 100 MB | 1 GB |
| PLATINA_AM_OWN01 | Índices | 100 MB | 5 GB | 100 MB | 1 GB |
| PLATINA_USERS_AM_OWN01 | Datos | 500 MB | 10 GB | 100 MB | 1 GB |
| PLATINA_USERS_AM_OWN01 | Índices | 100 MB | 5 GB | 100 MB | 1 GB |

El almacén de usuarios utilizado, es distintos al de resto de componentes, ya que inicialmente se instalará este componente de manera aislada.

6.2.1 Desempaquetado del componente.

Todas las acciones descritas se realizarán con el usuario platina del SO.

6.2.1.1 Descompresión del paquete:

Descomprimos el fichero wso2am-1.8.0.zip en directorio [RaízPlatina].

Por ejemplo, en la ruta /opt/platina

6.2.1.2 Creación del enlace simbólico:

Creamos un enlace simbólico a la carpeta que se genera tras la descompresión para poder referirnos a ella cómodamente. Ejecutamos el comando:

```
$ln -s [RaízPlatina]/wso2am-1.8.0 [RaízPlatina]/am
```

En adelante este directorio será referenciado como **\$AM_HOME**

6.2.2 Configuración AM

A continuación se detalla las modificaciones a realizar para la correcta configuración del componente.

6.2.1.3 Configuración clúster

Distinguiremos dos opciones a la hora de realizar la configuración del clúster:

- *Utilizando Multicasting*
- *Utilizando WKA*

Multicasting es la opción recomendada y configurada por defecto en Platina V3. Este mecanismo permite que se vayan añadiendo dinámicamente nuevos miembros esclavos al clúster sin necesidad de configuración en el resto de nodos y sin requerir el reinicio de los mismos. Este mecanismo requiere que la red de datos utilizada permita comunicación multicast.

La otra opción es utilizar WKA (Well Known Address), donde habría que identificar a cada miembro del clúster de forma manual en la configuración. Cuando se utiliza la configuración por WKA, cada vez que se añada un nuevo nodo se requerirá un cambio en la configuración de todos los nodos así como su reinicio.

6.2.1.3.2 Configuración Multicasting:

Editar fichero **\$AM_HOME/repository/conf/axis2/axis2.xml** y realizar las siguientes modificaciones:

Habilitar clúster:

```
<clustering class="org.wso2.carbon.core.clustering.hazelcast.HazelcastClusteringAgent"
enable="true">
```

Identificación de miembros del clúster:

```
<parameter name="membershipScheme">multicast</parameter>
```

Establecer nombre del dominio:

```
<parameter name="domain">platina.am.juntadeandalucia.grupo1</parameter>
```

Dirección y puerto utilizados en multicasting:

```
<parameter name="mcastAddress">228.0.0.4</parameter>
```

```
<parameter name="mcastPort">45564</parameter>
```

IP de la interfaz de red utilizada para el multicasting:

```
<parameter name="mcastBindAddress">IP. Ejemplo. 10.240.203.112</parameter>
```

IP del nodo que se esta configurando: Esta es la IP que es informada por el nodo cuando se une al clúster, por lo que no debe ser 127.0.0.1, localhost, o similares. Bastará con la IP de la interfaz de red donde esta escuchando el servicio:

```
<parameter name="localMemberHost">IP. Ejemplo. 10.240.203.112</parameter>
```

Puerto utilizado por el nodo:

```
<parameter name="localMemberPort">4003</parameter>
```

Descomentar los parámetros para que el nodo actúe como administrador de grupo (Solo nodo Master):

```
<groupManagement enable="true">
```

```
  <aplicación clúster name="platina.am.juntadeandalucia.grupo1"
  description="AM - Grupo 1"
  agent="org.wso2.carbon.core.clustering.hazelcast.HazelcastClusteringAgent"/>
```

```
</groupManagement>
```

6.2.1.3.3 Configuración WKA:

Editar fichero **\$AM_HOME/repository/conf/axis2/axis2.xml** y realizar las siguientes modificaciones:

Habilitar clúster:

```
<clustering class="org.wso2.carbon.core.clustering.hazelcast.HazelcastClusteringAgent"
enable="true">
```

Identificación de miembros del clúster:

```
<parameter name="membershipScheme">wka</parameter>
```

Establecer nombre del dominio:

```
<parameter name="domain">platina.am.juntadeandalucia.grupo1</parameter>
```

IP del nodo que se esta configurando: Esta es la IP que es informada por el nodo cuando se une al clúster, por lo que no debe ser 127.0.0.1, localhost, o similares. Bastará con la IP de la interfaz de red donde esta escuchando el servicio:

```
<parameter name="localMemberHost">IP. Ejemplo. 10.240.203.112</parameter>
```

Puerto utilizado por el nodo:

```
<parameter name="localMemberPort">4003</parameter>
```

Listado de miembros estáticos (solo WKA):

```
<members>

  <member>

    <hostName>[IP/host nodo1]</hostName>

    <port>[puerto]</port>

  </member>

  <member>

    <hostName>[IP/Host nodo2]</hostName>

    <port>[puerto]</port>

  </member>

</members>
```

Descomentar los parámetros para que el nodo actúe como administrador de grupo (Solo nodo Master):

```
<groupManagement enable="true">

  <applicationDomain name="platina.am.juntadeandalucia.grupo1" description="AM - Grupo
1" agent="org.wso2.carbon.core.clustering.hazelcast.HazelcastGroupManagementAgent"/>

</groupManagement>
```

6.2.1.4 Configuración registro

Editar fichero: **\$AM_HOME/repository/conf/registry.xml**

Configuración del Datasource:

```
<currentDBConfig>wso2registry</currentDBConfig>
```

```
<readOnly>false</readOnly>
```

```
<enableCache>true</enableCache>
```

```
<registryRoot></registryRoot>
```

currentDBConfig: Nombre de la configuración de BBDD utilizada.

ReadOnly: Toma un valor booleano (false/true). Estableciendo la propiedad a false, se crea una instancia Lectura-Escritura. Estableciéndola a true, una instancia inmutable. El escenario de una instancia lectura-escritura y uno o varios nodos de solo lectura resuelve posibles problemas de concurrencia.

Estableceremos false para el nodo Master, mientras que crearemos instancias inmutables para los nodos esclavos.

EnableCache: Habilita/Deshabilita la caché

registryRoot: Define el “punto de montaje” raíz del registro. En un escenario de varias instancias atacando a un único registro de gobierno, se puede optar por tres configuraciones distintas:

- A y B al mismo nivel: En este caso, las instancias A y B tienen el mismo punto de montaje, ya sea “/” , “/raíz” o el identificar que se quiera dar.
- A es el padre de B: A tiene el punto de montaje en “/” o “/raíz” mientras que B cuelga de “/hijo” o “/raíz/hijo”
- A y B están aislados: A tiene el punto de montaje en “/a” y B tiene como punto de montaje “/b”

Completar las propiedades del Datasource de AM

```
<dbConfig name="wso2registry">
```

```
<url>jdbc:oracle:thin:@10.240.203.30:1521/PRUSICOP</url>
```

```
<userName>PLATINA_AM_OWN01</userName>
```

```
<password>PLATINA_AM_OWN01</password>
```

```
<driverName>oracle.jdbc.driver.OracleDriver</driverName>  
  
<maxActive>50</maxActive>  
  
<maxWait>60000</maxWait>  
  
<minIdle>5</minIdle>  
  
</dbConfig>
```

Es posible completar la información del datasource a utilizar en el fichero
\$AM_HOME/repository/conf/datasources/master-datasources.xml y
referenciar el nombre jdbc:

```
<dbConfig name="wso2registry">  
    <dataSource>jdbc/AM_DATASOURCE</dataSource>  
</dbConfig>
```

6.2.1.5 Configuración Carbon:

Editar fichero **\$AM_HOME/repository/conf/carbon.xml**

Establecer el Offset (desplazamientos de puertos)

```
<Offset>N</Offset>
```

Donde N es un numero que indica el desplazamiento de los puertos. Es decir, a todos los puertos por defecto utilizados por el producto, se le sumara el valor de N.

Descomentar la sección de configuración de persistencia del Mediador:

```
<MediationConfig>  
  
    <LoadFromRegistry>false</LoadFromRegistry>  
  
    <SaveToFile>false</SaveToFile>
```




**Soporte Platina para los servicios de
Testing. PlatinaV3**
Manual de Instalación API Manager

**Consejería de Hacienda y
Administración Pública**

```
<Persistence>enabled</Persistence>
```

```
<RegistryPersistence>enabled</RegistryPersistence>
```

```
</MediationConfig>
```

LoadFromRegistry: Controla cuando cargar la configuración desde el registro. **Establecer a true para los nodos esclavos. False para el Maestro.**

Configurar URL Frontal de administración y servicios ofrecidos por AM:

```
<ServerURL>https://{carbon.local.ip}:{carbon.management.port}$  
{carbon.context}/services/</ServerURL>
```

Configurar WebContextRoot

```
<WebContextRoot></WebContextRoot>
```

WebContextRoot : Contexto raíz. Si se deseará publicar en un contexto distinto, por ejemplo, /am, modificar el valor.

Configurar nombre host:

```
<HostName>am.platina.i-administracion.junta-andalucia.es</HostName>
```

HostName: Establecer el nombre de host del frontal utilizado.

6.2.1.6 Configuración autenticación y almacén de usuarios:

Editar fichero: **\$AM_HOME/repository/conf/user-mgt.xml**

Este almacén será distinto al utilizado por el resto de componentes.

Establecer nombre del role administración, así como el nombre de usuario y password del administrador:

```
<AdminRole>admin</AdminRole>  
  
<AdminUser>  
  
    <UserName>admin-am</UserName>  
  
    <Password>admin</Password>  
  
</AdminUser>
```

Configurar el Datasource:

```
<Property name="url">jdbc:oracle:thin:@10.240.203.30:1521/PRUSICOP</Property>  
  
    <Property name="userName">PLATINA_USERS_AM_OWN01</Property>  
  
    <Property name="password">PLATINA_USERS_AM_OWN01</Property>  
  
    <Property name="driverName">oracle.jdbc.driver.OracleDriver</Property>  
  
    <Property name="maxActive">50</Property>  
  
    <Property name="maxWait">60000</Property>  
  
    <Property name="minIdle">5</Property>
```

Es posible completar la información del datasource a utilizar en el fichero
\$AM_HOME/repository/conf/datasources/master-datasources.xml y
referenciar el nombre jdbc:

```
<Property name="dataSource">jdbc/AM_USERS_DATASOURCE</Property>
```

6.2.1.7 Configuración de la caché:

Editar fichero: **\$AM_HOME/repository/conf/etc/cache.xml**

```
<configuration>

  <clustering>

    <enabled>true</enabled>

    <clusterName>platina-am-cache</clusterName>

  </clustering>

  <cacheMode>replicated</cacheMode>

  <sync>false</sync>

  <maxIdleSeconds>1800</maxIdleSeconds>

</configuration>
```

6.2.1.8 Copiado del driver OJDBC al directorio lib

Copiar el driver OJDBC proporcionado al directorio:

`$AM_HOME/repository/components/lib`

6.2.1.9 Ajustes de valores de la JVM:

Los valores de memoria utilizados por defecto no son los recomendables para un entorno productivo. Se recomienda cambiar estos valores por los siguientes:

- `-Xms2048m`
- `-Xmx2048m` (mismo valor Xms)
- `-XX:MaxPermSize=256m`

Esta modificación se realizará en el fichero:

\$AM_HOME/bin/yajsw/conf/wrapper.conf

Ver **Anexo IV: Ajustes de memoria de la Máquina Virtual Java**, para más detalle.

6.2.1.10 Habilitar sincronización nodo

6.2.1.10.1 En el nodo Master:

Editar fichero **\$AM_HOME/repository/conf/carbon.xml**

Se utilizará un repositorio de Subversión para la sincronización de artefactos de los nodos. Se debe crear un nuevo repositorio y disponer de las credenciales necesarias para poder escribir en la ruta establecida del repositorio.

Para configurar, descomentar y establecer las siguientes propiedades:

```
<DeploymentSynchronizer>
  <Enabled>true</Enabled>
  <AutoCommit>true</AutoCommit>
  <AutoCheckout>true</AutoCheckout>
  <RepositoryType>svn</RepositoryType>
  <SvnUrl>http://svnrepo.example.com/repos/</SvnUrl>
  <SvnUser>username</SvnUser>
  <SvnPassword>password</SvnPassword>
  <SvnUrlAppendTenantId>true</SvnUrlAppendTenantId>
</DeploymentSynchronizer>
```

6.2.1.10.2 En los nodos Slaves:

Editar fichero **\$AM_HOME/repository/conf/carbon.xml**

Descomentar y establecer las siguientes propiedades:

```
<DeploymentSynchronizer>
```

```
  <Enabled>true</Enabled>
```

```
  <AutoCommit>false</AutoCommit>
```

```
  <AutoCheckout>true</AutoCheckout>
```

```
  <RepositoryType>svn</RepositoryType>
```

```
  <SvnUrl>http://svnrepo.example.com/repos/</SvnUrl>
```

```
  <SvnUser>username</SvnUser>
```

```
  <SvnPassword>password</SvnPassword>
```

```
  <SvnUrlAppendTenantId>true</SvnUrlAppendTenantId>
```

```
</DeploymentSynchronizer>
```

7 VERIFICACIÓN DEL PROCESO DE INSTALACIÓN

En este apartado se recogen las comprobaciones mínimas que deberán realizarse una vez finalizado el despliegue para asegurar la correcta configuración e instalación del sistema.

Verificación 1. Componente AM MASTER

| Prueba 1 | |
|--|--|
| Paso | Descripción |
| 1º | Arrancamos el componente AM en el nodo MASTER: #/etc/init.d/ platina-am start |
| 2º | Monitorizamos los ficheros de logs: <ul style="list-style-type: none"> • [RaizPlatina]/AM/repository/logs/wrapper.log • [RaizPlatina]/AM/repository/logs/wso2carbon.log • [RaizPlatina]/AM/repository/logs/wso2-apigw-errors.log • [RaizPlatina]/AM/repository/logs/wso2-apigw-service.log • [RaizPlatina]/AM/repository/logs/wso2-apigw-trace.log |
| 3º | Ejecutar la siguiente URL desde el navegador: https://<nodo-master>:9443/carbon/admin/login.jsp https://<nodo-master>:9443/publisher https://<nodo-master>:9443/store |
| Resultados esperados | |
| <ol style="list-style-type: none"> 1. El resultado debe ser la pantalla de administración de AM del nodo. 2. No se muestra ningún error en el fichero log que estamos monitorizando. | |

7.2 Verificación 2. Componente AM en SLAVE1

| Prueba 1 | |
|--|---|
| Paso | Requisitos previos |
| Paso | Descripción |
| 1º | <p>Arrancamos el componente AM en el nodo SLAVE1:</p> <pre>#/etc/init.d/ platina-am start</pre> |
| 2º | <p>Monitorizamos los ficheros de logs:</p> <ul style="list-style-type: none"> • [RaizPlatina]/AM/repository/logs/wrapper.log • [RaizPlatina]/AM/repository/logs/wso2carbon.log • [RaizPlatina]/AM/repository/logs/wso2-apigw-errors.log • [RaizPlatina]/AM/repository/logs/wso2-apigw-service.log • [RaizPlatina]/AM/repository/logs/wso2-apigw-trace.log |
| 3º | <p>Ejecutar la siguiente URL desde el navegador:</p> <pre>https://<nodo-slave1>:9443/carbon/admin/login.jsp</pre> <pre>https://<nodo-master>:9443/publisher</pre> <pre>https://<nodo-master>:9443/store</pre> |
| Resultados esperados | |
| <ol style="list-style-type: none"> 1. El resultado debe ser la pantalla de administración de AM del nodo. 2. No se muestra ningún error en el fichero log que estamos monitorizando. | |

7.3 Verificación 3. Verificación de puertos a la escucha en los nodos.

| Prueba 1 | |
|---|--|
| Paso | Prerequisitos |
| 1º | Está operativo el componente AM en el nodo que vamos a verificar. |
| Paso | Descripción |
| 1º | <p>Comprobamos qué puertos están a la escucha en el nodo. Se debe ejecutar la prueba tanto en el nodo MASTER, como en el nodo SLAVE1</p> <p>#netstat -ntlp grep java</p> |
| Resultados esperados | |
| <p>Debemos obtener, al menos los siguientes puertos a la escucha, aunque puede haber alguno más en ese momento.</p> <p>Nota: Las direcciones IP pueden cambiar en función de las interfaces de conexión que tengamos definidas. También el pid de los procesos java:</p> <pre> tcp6 0 0 :::9999 :::* LISTEN 4142/java tcp6 0 0 :::8243 :::* LISTEN 4142/java tcp6 0 0 :::8280 :::* LISTEN 4142/java tcp6 0 0 :::59642 :::* LISTEN 4142/java tcp6 0 0 10.238.105.52:10397 :::* LISTEN 4142/java tcp6 0 0 :::9443 :::* LISTEN 4142/java tcp6 0 0 :::9763 :::* LISTEN 4142/java tcp6 0 0 :::11111 :::* LISTEN 4142/java </pre> | |

8 Parada y arranque del servicio AM

La parada y arranque del servicio AM se realizará a través de la consola de comandos en cada uno de los nodos (MASTER y SLAVE1).

Los comandos para controlar el servicio del componente AM son:

- Arrancar el servicio:
`#!/etc/init.d/platina-am start`
- Parada del servicio:
`#!/etc/init.d/platina-am stop`
- Consultar el estado del servicio:
`#!/etc/init.d/platina-am status`

8.1 Monitorización de ficheros de logs

Debemos monitorizar los resultados de los siguientes ficheros para detectar cualquier error en arranque o la parada del servicio AM:

- [RaizPlatina]/am/logs/wrapper.log
- [RaizPlatina]/am/logs/wso2carbon.log
- [RaizPlatina]/am/logs/wso2-apigw-errors.log

9 Supervisión del servicio

En este apartado indicaremos las pruebas de supervisión recomendadas para monitorizar el buen funcionamiento del servicio AM.

9.1 Pruebas Generales

| IDENTIFICADOR | | |
|---------------|---------------------------------|---|
| PG1 | Nombre | Latencia de Red |
| | Descripción | <p>Medir tiempo de respuesta de servidores a mensajes ping (tiempo entre mensajes ICMP echo e ICMP echo reply) para los nodos y para el balanceador.</p> <p>Se deben comprobar todas las interfaces de red de los servidores a monitorizar.</p> |
| | Resultado | Tiempo (segundos) |
| | Servidor ejecuta | Cliente en la misma red que la interfaz de los servidores a monitorizar. |
| | Servidores a Monitorizar | <ul style="list-style-type: none"> • Nodo MASTER • Nodo SLAVE1 • Balanceador |
| | Frecuencia | 5 minutos |
| | Umbral Aviso | |
| | Umbral Problema | |

9.2 Pruebas específicas

| IDENTIFICADOR | | |
|---------------|---------------------------------|--|
| PE1 | Nombre | Procesos Java |
| | Descripción | Determinar el número de procesos (padre + hijos) existentes |
| | Resultado | Número de procesos |
| | Servidor ejecuta | N/A |
| | Servidores a Monitorizar | <ul style="list-style-type: none"> Nodo MASTER (Resultado es de 1 proceso java) Nodo SLAVE1 (Resultado es de 1 proceso java) |
| | Frecuencia | |
| | Umbral Aviso | |
| | Umbral Problema | Resultado distinto al indicado |

| IDENTIFICADOR | | |
|---------------|---------------|-------------------------------|
| PE2 | Nombre | Monitorización de puertos TCP |

| | | |
|--|---------------------------------|---|
| | Descripción | <p>Medir tiempo de establecimiento de conexión con los puertos TCP indicados (desde que se lanza solicitud de establecimiento de conexión TCP, hasta que el servidor la acepte):</p> <ul style="list-style-type: none"> • 9443 • 9763 • 8280 • 8243 |
| | Resultado | |
| | Servidor ejecuta | Cliente con visibilidad sobre el servidor y sobre la interfaz en la que se encuentra a la escucha el puerto monitorizado. |
| | Servidores a Monitorizar | Nodos MASTER y SLAVE |
| | Frecuencia | 5 minutos |
| | Umbral Aviso | Alguno de los puertos no responde. |
| | Umbral Problema | Alguno de los puertos no responde. |

| IDENTIFICADOR | | |
|---------------|-------------------------|--|
| PE3 | Nombre | Monitorización de consola de administración de AM |
| | Descripción | Verificación de que la consola de administración está activa, comprobando que responde la url: https://<nodo>:9443/carbon/admin |
| | Resultado | Página web de la consola de administración del AM |
| | Servidor ejecuta | Cliente con visibilidad de los nodos MASTER y SLAVE1. |

| | | |
|--|---------------------------------|--|
| | Servidores a Monitorizar | <ul style="list-style-type: none"> NODO MASTER <ul style="list-style-type: none"> <code>https://<nodo-master>:9443/carbon/admin</code> NODO SLAVE1 <ul style="list-style-type: none"> <code>https://<nodo-slave1>:9443/carbon/admin</code> |
| | Frecuencia | 5 minutos |
| | Umbral Aviso | Servicio detenido |
| | Umbral Problema | Servicio detenido |

| IDENTIFICADOR | | |
|---------------|--------------------------|---|
| PE4 | Nombre | Monitorización de servicios de AM |
| | Descripción | Verificación de que los servicios desplegados en el AM están activos, comprobando que responden las urls: <ul style="list-style-type: none">• http://<nodo>:8280/services• https://<nodo>:8243/services |
| | Resultado | Pagina web con los servicios desplegados y activos en el AM. |
| | Servidor ejecuta | Cliente con visibilidad de los nodos MASTER y SLAVE1. |
| | Servidores a Monitorizar | NODO MASTER <ul style="list-style-type: none">• http://<nodo-master>:8280/services• https://<nodo-master>:8243/services NODO SLAVE1 <ul style="list-style-type: none">• http://<nodo-slave1>:8280/services• https://<nodo-slave1>:8243/services |
| | Frecuencia | 5 minutos |
| | Umbral Aviso | Servicio detenido |
| | Umbral Problema | Servicio detenido |

ANEXOS

Anexo I: Tunning del sistema operativo.

○ **/etc/sysctl.conf file**

Mejora del rendimiento de la interfaz de red y del S.O. Para el AM en despliegues en producción.

```
net.ipv4.tcp_fin_timeout = 30
fs.file-max = 2097152
net.ipv4.tcp_tw_recycle = 1
net.ipv4.tcp_tw_reuse = 1
net.core.rmem_default = 524288
net.core.wmem_default = 524288
net.core.rmem_max = 67108864
net.core.wmem_max = 67108864
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 65536 16777216
```

○ **/etc/security/limits.conf**

Se modifica el nº de ficheros abiertos por usuarios del sistema.

```
* soft nfile 4096
* hard nfile 65535
```

Anexo II: Tuning del AM.

◦ \$AM_HOME/repository/conf/nhttp.properties

Se puede mejorar el rendimiento del transporte HTTP-NIO creando este fichero y especificando valores para los parámetros indicados.

Si el fichero no es creado, el AM tomará unos valores por defecto, que se corresponden con los valores indicados, en el siguiente ejemplo:

```
http.socket.timeout=60000
http.socket.buffer-size=8192
http.tcp.nodelay=1
http.connection.stalecheck=0

# HTTP Sender thread pool parameters
snd_t_core=20
snd_t_max=100
snd_alive_sec=5
snd_qlen=-1
snd_io_threads=2

# HTTP Listener thread pool parameters
lst_t_core=20
lst_t_max=100
lst_alive_sec=5
lst_qlen=-1
lst_io_threads=2
```

Anexo III. Configuración de URLs frontales e interfaz

Por un lado, si son conocidas, debemos indicar cuales son las urls frontales para acceso a los servicios de AM desde el exterior, tanto para el protocolo http, como para el https.

Por otro lado, hay que fijar la **ip** o **host** que se va a utilizar como **interfaz A** (de servicio) del nodo. Para ello sustituiremos **[direccion o host de interfaz A]** por el valor que corresponda al nodo.

Los cambios en este fichero hay que realizarlos, tanto en el nodo MASTER, como en el SLAVE1.

Debemos modificar el fichero **[RaízPlatina]/am/Repository/conf/axis2.xml**

```
<!-- ===== -->
<!--      Transport Ins (Listeners)      -->
<!-- ===== -->

<!-- the non blocking http transport based on HttpCore + NIO extensions -->
<transportReceiver name="http"
class="org.apache.synapse.transport.nhttp.HttpCoreNIOListener">
  <parameter name="port" locked="false">8280</parameter>
  <parameter name="non-blocking" locked="false">true</parameter>
  <parameter name="bind-address" locked="false">[direccion o host de interfaz A]</parameter>
  <parameter name="WSDLEPRPrefix" locked="false">http://dominio-balanceador/am/services</parameter>
  <parameter name="httpGetProcessor"
locked="false">org.wso2.carbon.transport.nhttp.api.NHttpGetProcessor</parameter>
  <!--<parameter name="priorityConfigFile" locked="false">location of priority
configuration file</parameter>-->
</transportReceiver>

  <!-- the non blocking https transport based on HttpCore + SSL-NIO extensions -->
  <transportReceiver name="https"
class="org.apache.synapse.transport.nhttp.HttpCoreNIOSSLListener">
    <parameter name="port" locked="false">8243</parameter>
    <parameter name="non-blocking" locked="false">true</parameter>
    <parameter name="bind-address" locked="false">[direccion o host de interfaz A]</parameter>
    <parameter name="WSDLEPRPrefix" locked="false">https://dominio-balanceador/am/services</parameter>
    <!--<parameter name="priorityConfigFile" locked="false">location of priority
configuration file</parameter>-->
    <parameter name="httpGetProcessor"
locked="false">org.wso2.carbon.transport.nhttp.api.NHttpGetProcessor</parameter>
    <parameter name="keystore" locked="false">
```

Si conocemos la url frontal de los servicios de administración del AM, también será necesario modificar el fichero **[RaízPlatina]/am/repository/conf/carbon.xml:**

```
<!--
  The URL of the back end server. This is where the admin services
  are hosted and
  will be used by the clients in the front end server.
  This is required only for the Front-end server. This is used when
  separating BE server from FE server
-->
```




**Soporte Platina para los servicios de
Testing. PlatinaV3**
Manual de Instalación API Manager

**Consejería de Hacienda y
Administración Pública**

```
<ServerURL>http://<dominio-platina>/am/admin/services/</ServerURL>
```

Nota: Cualquier modificación en los ficheros anteriores, requiere un reinicio del nodo para que los cambios se vean reflejados.

Anexo IV: Ajustes de memoria de la Maquina Virtual Java

Para realizar los ajustes debemos editar el fichero **\$AM_HOME/bin/yajsw/conf/wrapper.conf**

Los parámetros principales de configuración a modificar son los relativos a la configuración de la memoria que se va a utilizar.:

- Xms: Memoria inicial.
- Xmx: Memoria máxima
- XX:MaxPermSize: Tamaño de memoria de generación permanente

FICHERO: bin/yajsw/conf/wrapper.conf

```
...
# Java Additional Parameters
wrapper.java.additional.1=-Xms2048m
wrapper.java.additional.2=-Xmx2048m
wrapper.java.additional.3=-XX:MaxPermSize=256m
wrapper.java.additional.4=-Dcarbon.home=.
wrapper.java.additional.5=-Djava.endorsed.dirs=lib/endorsed
wrapper.java.additional.6=-Dcom.sun.management.jmxremote
wrapper.java.additional.7=-Dwso2.server.standalone=true
wrapper.java.additional.8=-Djava.io.tmpdir=tmp
wrapper.java.additional.9=-Dwso2.transports.xml=repository/conf/mgt-transports.xml
wrapper.java.additional.10=-Dcarbon.registry.root=/
wrapper.java.additional.11=-Dcom.sun.jndi.ldap.connect.pool.authentication=simple
wrapper.java.additional.12=-Dcom.sun.jndi.ldap.connect.pool.timeout=3000
```

Anexo V: Configuración de proxy para acceso a internet

En caso que fuese necesario configurar un proxy para proporcionar acceso a internet a un nodo para configurar el acceso a internet, debemos editar el fichero `$AM_HOME/bin/yajsw/conf/wrapper.conf` del nodo AM y añadir las siguientes líneas en la sección correspondiente:

```
wrapper.java.additional.14=-Dhttp.proxyHost=192.168.205.2  
wrapper.java.additional.15=-Dhttp.proxyPort=8000
```

Nota: Los valores de `additional.14` y `15` deben ser consecutivos con el último usado en el fichero `wrapper.conf`

Anexo VI: Demonización del servicio

Para realizar la demonización del servicio, se ha optado por la herramienta yajsw (Yet Another Java Service Wrapper. Se ha utilizado la versión 11.06 de dicha herramienta, que se puede descargar desde el siguiente enlace:

<http://sourceforge.net/projects/yajsw/files/latest/download?source=files>

También se facilita con la entrega.

Los pasos a seguir para demonizar el servicio son los siguientes:

- **Instalación de la herramienta:**

La instalación básicamente consiste en descomprimir el fichero yajsw-stable-11.06.zip a una ubicación, por ejemplo, `$AM_HOME/bin/yajsw`. Una vez descomprimido, será necesario asignar permisos de ejecución a todos los scripts existentes en el directorio bin de la herramienta:

```
chmod +x *.sh
```

- **Generación del fichero wrapper.conf**

Para la generación del fichero wrapper.conf, será necesario tener corriendo la aplicación. Si no está en funcionamiento, la levantaremos con el siguiente comando:

```
/opt/platina/am/bin/wso2server.sh -start
```

Una vez levantada, identificaremos el PID del proceso JAVA de nuestra instancia:

```
ps -ef | grep java
```

[SALIDA]

```
platina 23039 23020 0 Apr24 ? 00:08:29  
/usr/lib/jvm/jdk1.6.0_20/bin/java -Xbootclasspath/a ....
```

Y ejecutaremos el script `$AM_HOME/bin/yajsw/bin/genConfig.sh <PID>`, donde `<PID>` es el PID del proceso Java obtenido en el paso anterior. Para nuestro ejemplo se ejecutaría:

```
$AM_HOME/bin/yajsw/bin/genConfig.sh 23039
```

Esto generará el fichero wrapper.conf en el directorio por defecto que será

\$AM_HOME/bin/yajsw/conf/wrapper.conf

Habrá que realizar algunas modificaciones a dicho fichero. Cambiaremos los siguientes parámetros:

```
wrapper.app.account=platina  
wrapper.logfile=$AM_HOME/repository/logs/wrapper.conf  
wrapper.console.title=WSO2 AM 1.8.0  
wrapper.ntservice.name=platina-am  
wrapper.ntservice.displayname=WSO2 Platina AM 1.8.0  
wrapper.ntservice.description=WSO2 Platina AM 1.8.0
```

Donde:

wrapper.app.account: Es la cuenta de usuario existente en el sistema con el que se levantará la aplicación.
wrapper.logfile: Ruta donde se guardaran los logs.
wrapper.console.title: Título mostrado
wrapper.ntservice.name: Nombre del servicio que se creará en /etc/init.d
wrapper.ntservice.displayname: Nombre mostrado del servicio
wrapper.ntservice.description: Descripción

Los parámetros que se le pasan a la JVM, se especifica mediante el parámetro:

wrapper.java.additional.XY = <PARAMETRO>

Donde XY es una secuencia de números que irá incrementando conforme se vayan añadiendo nuevos parámetros. Por ejemplo, si queremos utilizar la codificación UTF, se añadirá el siguiente parámetro:

```
wrapper.java.additional.26 = -Dfile.encoding=UTF8
```

La secuencia 26, indica que existen 25 parámetros previos ya configurados en el fichero.

- **Instalación del demonio:**

Una vez realizada las modificaciones sobre el fichero wrapper.conf, ejecutaremos el siguiente script:

\$AM_HOME/bin/yajsw/bin/installDaemon.sh

Que creará en el fichero /etc/init.d/platina-am , así como las correspondientes entradas en los diferentes niveles de ejecución.

Una vez instalado el servicio, se podrá arrancar, parar , reiniciar y ver el estado de la aplicación con los siguientes comandos:



**Soporte Platina para los servicios de
Testing. PlatinaV3**
Manual de Instalación API Manager

**Consejería de Hacienda y
Administración Pública**

/etc/init.d/platina-am start
/etc/init.d/platina-am stop
/etc/init.d/platina-am restart
/etc/init.d/platina-am status

Para obtener más información sobre el proyecto y más opciones de configuración, visitar la página del proyecto:

<http://yajsw.sourceforge.net/>