

# Manual de puesta en Cluster del Servidor de Firma de la plataforma @Firma 4.0.

**CONTROL DE COMPROBACIÓN Y APROBACIÓN**

Documento nº: TI-20-1074-CLU-001  
Revisión: 1  
Fecha: 23/09/04

**REALIZADO**

23/09/04

Javier  
Cerceda  
Garcia  
Analista

---

**COMPROBADO**

---

**APROBADO**

---

## CONTROL DE MODIFICACIONES

Documento nº: TI-20-1074-CLU-001  
Revisión: 1  
Fecha: 23/09/04

Rev. 1  
Fecha 23/09/04  
Autor/es JCG  
Descripción Borrador inicial

## CONTROL DE DISTRIBUCIÓN

Documento nº: TI-20-1074-CLU-001  
Revisión: 1  
Fecha: 23/09/04

### Copias Electrónicas:

La distribución de este documento ha sido controlada a través del sistema de información.

### Copias en Papel:

La vigencia de las copias impresas en papel está condicionada a la coincidencia de su estado de revisión con el que aparece en el sistema electrónico de distribución de documentos.

El control de distribución de copias en papel para su uso en proyectos u otras aplicaciones es responsabilidad de los usuarios del sistema electrónico de información.

Fecha de impresión 14/10/04 16:43

### Distribución en Papel:

Nombre o Cargo y (Organización)	Nº de Ejemplares	Referencia de la carta de transmisión y fecha

## Índice

1	Objeto	6
2	Alcance	7
3	Documentos de referencia	8
4	Introducción	9
5	Servicio de Alta Disponibilidad para Servidor de Firma.	10
6	Consideraciones sobre la puesta en Cluster	13
7	Configuración de las aplicaciones Clientes	15
7.1	Acceso a las interfaces RMI-IIOP mediante HA-JNDI	15
7.2	Acceso a las interfaces WEBSERVICES	16
ANEXOS		18
A.1	Cluster en JBoss	19

## 1 Objeto

Es objeto de este documento describir el proceso de puesta en cluster del Servidor de Firma de la plataforma @firma Versión 4.0, la plataforma de Autenticación y Firma Digital de la Junta de Andalucía.

Los objetivos globales de este proceso son:

- ✍ Describir en que consiste poner en cluster el Servidor de Firma, la solución adoptada, los elementos necesarios y la función de cada uno de ellos.
- ✍ Describir los pasos necesarios a realizar por el administrador del sistema para poner en cluster el Servidor de Firma.
- ✍ Describir los componentes de la plataforma @firma que atenderán las peticiones de los distintos tipos de Autenticación y Firma Digital.
- ✍ Describir en que se ven afectados las aplicaciones clientes que acceden a la plataforma: WebServices y RMI-IIOP.

## 2 Alcance

El siguiente documento recoge la puesta en cluster del componente Servidor de Firma de la plataforma @firma Versión 4.0, la plataforma de Autenticación y Firma Digital de la Junta de Andalucía.

### 3 Documentos de referencia

- Documento TI-20-1074-INS-001, Manual de Instalación de la plataforma @Firma Versión 4.0.
- Documento TI-20-1074-ARQ-001, Manual de Arquitectura de la plataforma @Firma Versión 4.0.



## 4 Introducción

Se denomina @Firma a la plataforma Centralizada de Autenticación y Firma Digital presente en la Consejería de Justicia y Administración Pública.

La plataforma @Firma 4.0 esta formada básicamente por tres elementos: Servidor de Firma, Sistema de Custodia y Fachada de Comunicaciones.

Es objeto de este documento presentar los elementos necesarios para habilitar al Servidor de Firma como un sistema de Alta Disponibilidad, escalable y tolerante a fallos, y describir el nuevo comportamiento que deben tener los clientes para trabajar con el Servidor.

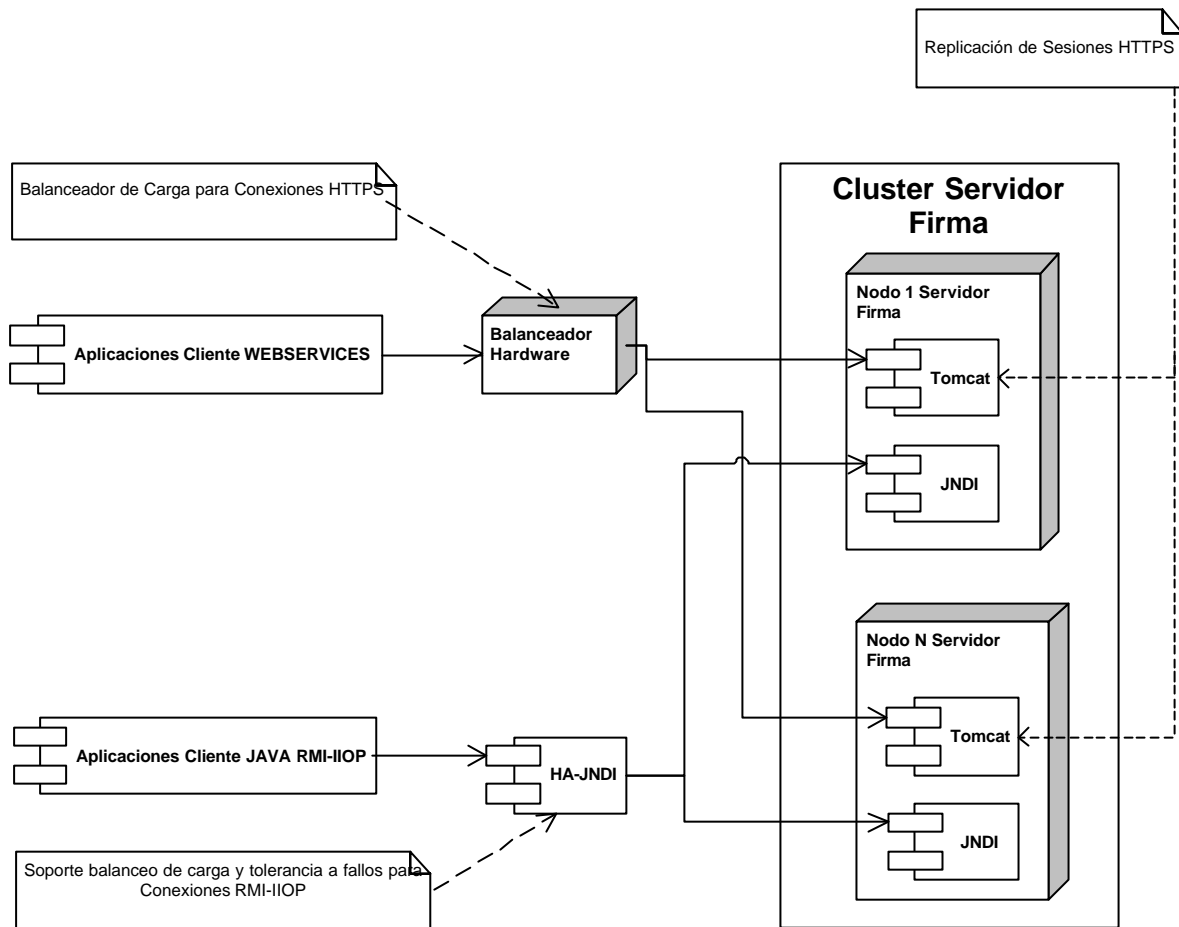
## 5 Servicio de Alta Disponibilidad para Servidor de Firma.

Para convertir el Servidor de Firma en un sistema de Alta Disponibilidad, tolerante a fallos, escalable y con capacidad de balanceo de carga se debe poner en cluster, utilizando los nodos que se consideren oportunos, normalmente dos.

La solución que se ha adoptado se basa en la combinación de tres elementos:

- ? El servicio de cluster de JBoss.
- ? El servicio de replicación de sesiones de Tomcat.
- ? Algún/os elemento/s externo/s para el balanceo de carga en los servicios Http/https y ftp.

La siguiente figura resume de forma gráfica la arquitectura básica de la solución adoptada, incluyéndose cada uno de los elementos participantes.



En la puesta en cluster del Servidor de Firma se han adoptado dos soluciones distintas en función del tipo de servicio:

- ? Los servicios publicados por RMI-IIOP/SSL necesitan para su puesta en cluster la configuración del "servicio de cluster" disponible con JBoss. Para ello se ha configurado el servicio, se ha activado para la partición "FirmaPartition" (nombre de cluster) y se han configurado todos los componentes de firma para su registro dentro de dicha partición. Las aplicaciones clientes accederán a los componentes publicados dentro de la Partición a través de HA-JNDI.
- ? Los servicios publicados por HTTPS, en nuestro caso todos los WEBSERVICES, necesitan la colaboración de dos elementos. Por un lado, se ha configurado el servicio de replicación de sesiones dentro del Tomcat de JBoss. Este servicio únicamente contempla la duplicación de sesiones, no contempla capacidades de balanceo de carga. Esta funcionalidad deberá ser suministrada por un elemento externo a la plataforma, un balanceador hardware. Un balanceador hardware es

posible que no sea consciente de cuando se ha caído alguno de los nodos. Por ello, es posible también desarrollar un balanceador software que tenga alguna funcionalidad añadida y sea más inteligente. En este caso no se ha desarrollado y se ha optado por el hardware.

Todo sistema puesto en cluster debe de contemplar mecanismos para la sincronización de variables de aplicación entre cada una de las instancias que componen el cluster. En términos técnicos estaríamos hablando de mecanismos de "Sincronización de Singletons" bajo aplicaciones en cluster. La sincronización de estos elementos software se realiza por las aplicaciones utilizando servicios básicos, a bajo nivel, proporcionados por el clúster (DRM) o servicios a nivel de aplicación como JMS.

En el caso concreto de @Firma, se ha adaptado el Sistema de Configuración para permitir la sincronización de la configuración en todos y cada uno de los nodos del cluster. Se ha utilizado el Servicio DRM (Distributed Replicant Manager) presente en el API básico del cluster de JBoss.

## 6 Consideraciones sobre la puesta en Cluster

Se utilizará una configuración de cluster Activo-Activo por motivos de eficiencia y aprovechamiento de recursos.

Partimos de que hemos realizado la instalación básica de la plataforma @firma 4.0. En esta configuración disponemos de las siguientes máquinas:

- Sistema de Custodia
- Servidor de Firma
- Fachada de Comunicaciones

El primer paso consiste en disponer de una máquina nueva para instalar un nuevo Servidor de Firma siguiendo los pasos descritos en el apartado 6.2 del manual de instalación de la plataforma. Esta máquina dispondrá de un cliente oracle y se conectará al Sistema de Custodia anterior.

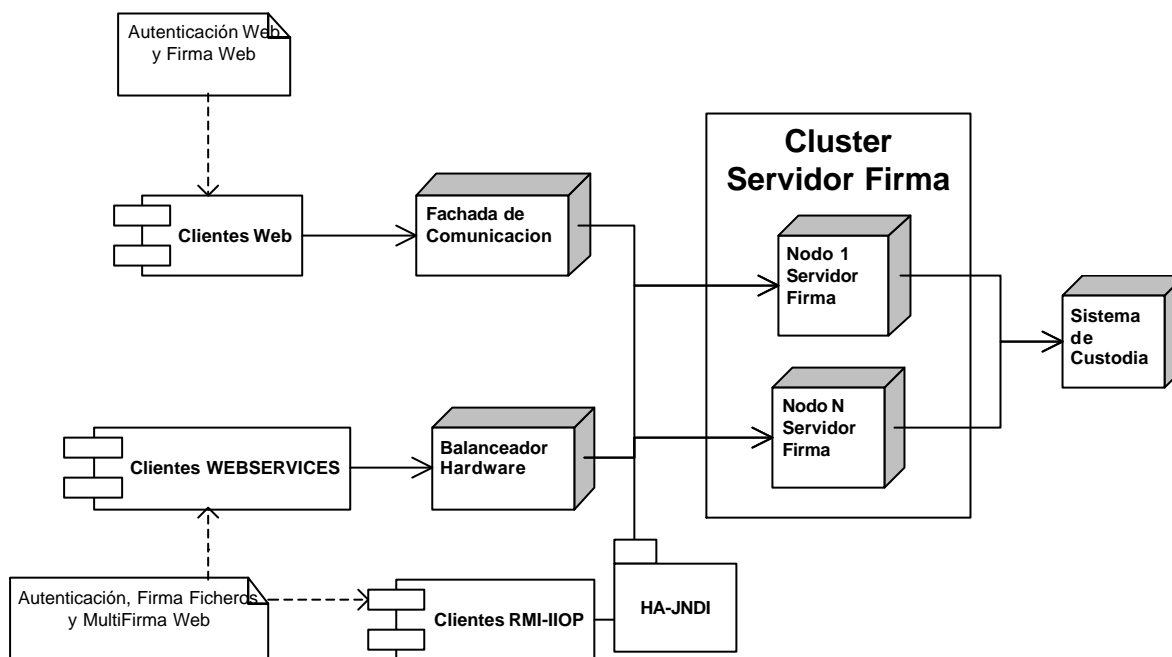
Cuando se inicialicen ambos servidores de firma (Jboss), se pondrán en cluster automáticamente gracias a la configuración del Jboss.

De esta forma ya tenemos dos Servidores de Firma en cluster. Este proceso se repetirá con todos los nodos que deseemos añadir.

El servicio de cluster de Jboss detecta automáticamente los nuevos nodos que se vayan incorporando y gestiona aquellos que se vayan dando de baja. De esta forma, este proceso es transparente para el administrador del sistema.

Por otro lado la configuración del sistema se replica automáticamente en todos los nodos presentes en el cluster cuando se envía la configuración desde la herramienta de administración de cualquier nodo.

La siguiente figura muestra los elementos descritos anteriormente, la función de cada uno, los procesos y peticiones atendidos por cada uno.



A continuación se describen las peticiones atendidas por cada elemento:

- Fachada de comunicaciones: Autenticación Web y Firma Web.
- Balanceador Hardware: WebServices de Autenticación, Firma de Ficheros, Multifirma Web y API Consulta Firma Web.
- Nodos de Servidor Firma: Interfaces RMI-IIOP de Autenticación, Firma de Ficheros, Multifirma Web y API Consulta Firma Web.

## 7 Configuración de las aplicaciones Clientes

### 7.1 Acceso a las interfaces RMI-IIOP mediante HA-JNDI

Los clientes que acceden a las interfaces RMI-IIOP de la plataforma @firma deberán utilizar el servicio HA-JNDI, para ello simplemente deben indicar una lista de servidores de conexión en vez de un solo servidor en la configuración de la conexión a la plataforma e indicar que se conecten por el puerto 1100.

- En el caso de Firma de Ficheros se utiliza el paquete "ApiFirmaCliente.jar", concretamente la clase com.telvent.utilidades.ConexionFirma. El método setParámetrosConexion recibirá en el parámetro "servidor" la lista de nodos Servidores Firma separados por comas e indicando el puerto 1100:

```
ConexionFirma.setParametrosConexion(" 196.123.53.242:1100,196.123.53.243:1100",  
"user", "pass");
```

- En el caso de Multifirma Web se utiliza el paquete "multifirmacliente.jar", concretamente la clase com.telvent.conexion.ConexionMultiFirma. El constructor de dicha clase recibirá la lista de nodos Servidores Firma separados por comas en el parámetro "servidor".
- En el caso de Multifirma Web Masiva se utiliza el paquete "multifirmamasivacliente.jar", concretamente la clase com.telvent.conexion.ConexionMultiFirmaMasiva. El constructor de dicha clase recibirá la lista de nodos Servidores Firma separados por comas en el parámetro "servidor".
- En el caso de Api Consulta de Firma Web se utiliza el paquete "TelventRemoteApi.jar", concretamente la clase com.telvent.firma.fachada.ConexionFirmaApiFacade. El constructor de dicha

clase recibirá la lista de nodos Servidores Firma separados por comas en el parámetro "servidor" .

- En el caso de Autenticación se utiliza el paquete "ModuloAutenticacionTelvent.jar" , concretamente la clase com.telventi.conexion.ConexiónAutenticación. El constructor de dicha clase recibirá la lista de nodos Servidores Firma separados por comas en el parámetro "servidor" .

## 7.2 Acceso a las interfaces WEBSERVICES

Los clientes que acceden a las interfaces WebServices de la plataforma @firma no se ven afectados, excepto en que deberán utilizar la IP virtual del Balanceador Hardware en vez de utilizar la IP del Servidor de Firma.





# ANEXOS

## A.1 Cluster en JBoss

Un cluster es un conjunto de nodos, en nuestro caso instancias de JBoss. Para formar un cluster, las instancias de JBoss se agrupan formando una o varias particiones.

Dentro de una misma red, pueden coexistir una o varias particiones diferenciándose cada una por un nombre identificativo.

La implementación actual del cluster de JBoss incluye muchas características entre las que podemos citar:

- ? Descubrimiento automático de los nodos del Cluster. Los nodos dentro del cluster se localizan entre ellos de forma automática sin necesidad de configuración adicional.
- ? Balanceo de carga y tolerancia a fallos para los servicios:
  - o JNDI
  - o RMI
  - o Entity Beans
  - o Stateful Session Beans
  - o Stateless Session Beans
- ? Replicación de sesiones Http con Tomcat y Jetty
- ? Dinamyc JNDI Discovery. Los clientes localizan automáticamente el servicio JNDI (HA-JNDI).
- ? Replicación dentro del Cluster del servicio JNDI.
- ? Farming. Distribución de aplicación dentro del cluster en “caliente” (hot-deploy).

Para lograr sus objetivos JBoss duplica sus servicios introduciendo nuevos servicios de alta disponibilidad, los cuales, abstraen al cliente de las problemáticas del cluster y le proporciona, de forma transparente, las características de balanceo de carga y tolerancia a fallos.