

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

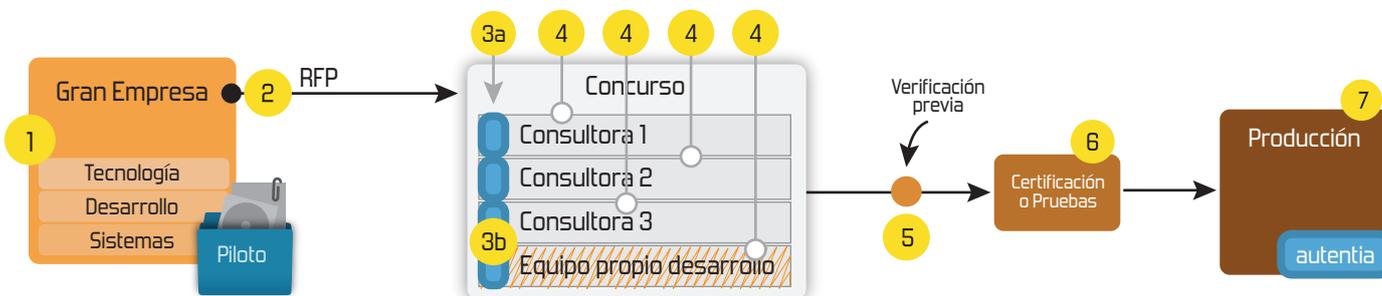
1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

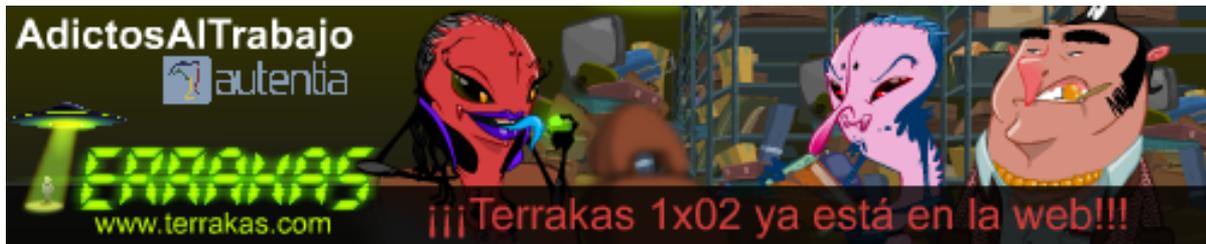
Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

Host
en

[Inicio](#) [Quiénes somos](#) [Tutoriales](#) [Formación](#) [Comparador de salarios](#) [Nuestro libro](#)

» Estás en: [Inicio](#) [Tutoriales](#) Gestores de repositorios Maven



DESARROLLADO POR:

Ángel García Jerez



Consultor tecnológico de desarrollo de proyectos informáticos. Co-autor del libro "Actualización y mantenimiento del PC (Edición de 2010) publicado por Anaya Multimedia

Ingeniero Técnico en Informática de Sistemas e Ingeniero en Informática (premio al mejor expediente de su promoción)

Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

[Ver tutoriales de Ángel García Jerez](#)

¿Te gustaría ser **Controlador Aéreo?** Preparamos el 100% de las p
Inglés + FEAST+ Psicotécnicos+ Ent

[¡Infórmate!](#) **maste**

Fecha de publicación del tutorial: 2012-01-24

Share |

0

[Reg](#)

GESTORES DE REPOSITORIOS MAVEN

1. Introducción
2. Entorno
3. Espacio en disco
4. Rendimiento
5. Memoria
6. Búsquedas
7. Repositorios
8. Auditoría y notificaciones

9. Conclusión

1. Introducción

Desde hace ya unos años el mundo Java utiliza una herramienta llamada Maven que facilita la construcción en los proyectos. Entre los aspectos más importantes a destacar se encuentran las dependencias que nos permite saber qué librerías y qué versiones son necesarias en el proyecto.

Maven utiliza una matriz de repositorios remotos que le permite localizar y descargar todo lo que necesita nuestro proyecto de forma transparente al desarrollador. Además de los repositorios remotos, Maven utiliza un repositorio local que lo utiliza como caché evitando la descarga en las siguientes generaciones de código, reduciendo así el tiempo que supondría volver a descargarse todas las librerías.

En determinados entornos, Maven puede ser más que suficiente pero en grandes organizaciones con restricciones de acceso a Internet, el control de acceso a los repositorios, la exclusión de ciertos repositorios, el consumo del ancho de banda o la administración de los repositorios de la propia organización quedan fuera del alcance de Maven. Debido a estas necesidades aparecieron en el mundo los Repositorios Maven.

En la actualidad podemos destacar tres: Archiva, Artifactory y Nexus. El objetivo de este tutorial es seleccionar la mejor opción en función de los resultados obtenidos. Para el análisis se ha utilizado un entorno standalone con su configuración por defecto que se distribuyen con cada una de ellas. Los entornos de este tutorial son:

- Archiva 1.3.5
- Artifactory 2.4.2
- Nexus 1.9.2.4

2. Entorno

Entorno utilizado para escribir este tutorial:

- **Hardware:** Mac Book Pro (Core 2 Duo 2,8 Ghz, 8 GB RAM, 128 GB SSD)
- **Sistema Operativo:** Lion 10.7.0
- **Active Directory:** Windows 2003 Server
- **JDK:** 1.6
- **Acceso a internet:** ADSL 6 Mbits

3. Espacio en disco

El espacio de utilización en disco es uno de los aspectos importantes a analizar de los gestores de repositorios cuando vamos a gestionar un número importante de ellos (remotos o propietarios). Un gestor que utilice un 2% más de espacio para almacenar el mismo número de artefactos puede ser un problema. Por ejemplo si nuestros repositorios ocupasen 1T de espacio un 2% sería 20G, espacio muy a tener en cuenta.

En la actualidad los gestores de repositorio utilizan dos formas de gestionar el almacenar

Existen unos que sólo utilizan el disco y otros que además del almacenamiento en disco. En el primero se encuentra Nexus y en el segundo están Archiva y Artifactory. Estos dos últimos utilizan una base de datos Derby aunque pueden ser configurados para ser utilizados con los gestores más importantes del mercado.

Una característica importante a destacar en Artifactory es que a pesar de que por defecto gestiona los repositorios también ofrece la posibilidad de usarse sin una base de datos.

Para este punto hemos utilizado un proyecto web multimódulo generado a partir del archetype (2.1.0) en el que sus librerías ocupan un total de 28M. Una vez ejecutado el proyecto el espacio utilizado por cada uno de ellos es el que se muestra en la tabla siguiente.

Gestor de repositorios Maven	Tamaño de ocupación (Mbytes)	Porcentaje respecto a Nexus
Archiva	40	117.6%
Artifactory	37	108.8%
Nexus	34	100.0%

Como se puede ver la aplicación que menor consumo de disco es Nexus seguido de Archiva. Tomando como referencia Nexus, Artifactory consume un 11% más de espacio, mientras que Archiva consume un 17% más.

4. Rendimiento

Para la prueba de rendimiento hemos decidido medir el tiempo que se tarda en generar el multimódulo utilizado anteriormente. Hemos realizado dos medidas con las siguientes situaciones:

- Con repositorios vírgenes. Ni en el repositorio local ni en el repositorio proxy del gestor existen artefactos almacenados.
- Con repositorio local vacío. Sólo el repositorio local está vacío, el repositorio proxy del gestor se encuentra con todos los artefactos del proyecto.

Los resultados obtenidos son:

Gestor de repositorios Maven	Tiempo con repositorios vírgenes (m:s.ms)	Tiempo con repositorio local vacío
Archiva	4:18.162	4:18.162
Artifactory	5:03.855	4:18.162
Nexus	4:08.447	4:08.447

Como se puede apreciar la aplicación más rápida con repositorios vírgenes es Nexus seguida de Nexus y finalmente Artifactory. Respecto a los tiempos una vez cacheados los artefactos por cada gestor los tiempos son muy similares habiendo una diferencia de 3 o 4 segundos entre una y otra.

Las diferencias tan significativas en la primera medida se deben principalmente al número de artefactos que el gestor realiza a la hora de almacenar un artefacto y al tipo de tecnologías utilizadas como la indexación, etc...

Con el objetivo de reducir el tiempo de Artifactory activamos en los repositorios la opción de cachear los artefactos. Con esta opción conseguimos que Artifactory se baje todas las dependencias de un artefacto que estemos solicitando al gestor. Pero mi sorpresa fue que activando dicha opción la construcción seguía prácticamente siendo el mismo.

5 Memoria.

El consumo de memoria es otro aspecto a tener en cuenta. En este punto Nexus sigue siendo el más eficiente, necesita una máquina virtual con una memoria inicial de 28M a diferencia de los otros dos que necesitan 34M.

También se ha observado la memoria que consume una vez arrancado y en este punto también Nexus es el más eficiente, consume 34M a diferencia de los otros dos que consumen 40M.

149M, respecto a 177M de Archiva y 317M de Artifactory.

6 Búsquedas

Las características de búsqueda que ofrecen todos los gestores son muy similares, por artefactos mediante un árbol jerárquico, realizar búsquedas por palabras clave, nombre de artefactos y por grupo, artefacto y versión. Adicionalmente, Nexus y Archiva ofrecen lo que permite conocer la versión de un artefacto del que no sabemos nada utilizando los ficheros. Artifactory permite realizar búsquedas en los POMs y metadatos XML utilizando XPath.

Todos ellos utilizan los índices de repositorios remotos para las búsquedas locales y ante estos son capaces de realizar una descarga incremental.

7 Repositorios

En este apartado ninguno de ellos destaca sobre los demás, todos trabajan con los mismos estándares y proporciona las mismas características básicas necesarias para su gestión, como el control de manejar repositorios maven1/maven2 o la gestión de repositorios snapshot o release.

La gestión de repositorios snapshot o release es algo peculiar en Nexus porque no por un repositorio se almacene artefactos del tipo snapshot o releases cosa que los otros sí. El problema ya que Nexus permite agrupar repositorios para que se comporten como uno solo.

Otra característica interesante que ofrecen todas las herramientas es la posibilidad de definición o exclusión que permiten restringir el acceso a ciertos artefactos o jerarquía del repositorio.

A pesar de llamarlos de forma diferente todos ellos trabajan con los mismos tipos de repositorios:

- Repositorios Locales: son los repositorios internos de la organización y deben estar en la misma máquina donde se encuentre desplegada la aplicación. Pueden llamarse Hosted Repositories (Nexus), Local Repository (Artifactory) o Managed Repository (Archiva).
- Repositorios Remotos: son aquellos que se encuentran en Internet y que queremos cachear en nuestra organización. Lo podemos encontrar como Proxy Repository (Nexus) o Remote Repository (Artifactory, Archiva).
- Grupo Repositorios: son repositorios que agrupan otros repositorios a los que se accede mediante una única URL. Nexus y Archiva lo llaman Group Repository y Virtual Repository en Artifactory.

Nexus define además otro tipo al que llama Virtual Repository que nos permite transformarse en maven2 o viceversa. Esta característica también la ofrecen los otros pero en forma de repositorios vistos anteriormente.

Una desventaja de Artifactory es la imposibilidad de indicar la ruta a un repositorio local o de repositorios de cualquier tipo en la misma ruta. El resto nos da la opción de almacenar repositorios de la aplicación o en otra ruta diferente del servidor.

8 Auditoría y notificaciones

De los tres gestores los únicos que nos proporcionan capacidades de auditoría dentro de Nexus, Artifactory y Archiva. Con Artifactory podemos acceder a varios logs: peticiones, importación de la aplicación. En cambio Archiva a pesar de disponer un buscador de eventos por repositorio, y fechas la información que nos ofrece no es muy relevante: creación/eliminación/modificación de directorios, copiado de un directorio o subida de ficheros mediante la interfaz web.

Respecto a las notificaciones sólo Nexus y Archiva tienen estas características. Nexus ofrece notificaciones de determinados eventos en cualquier repositorio como artefactos desplegados o actualizados, cambios de configuración, etc. En cambio Archiva nos ofrece RSS de cada repositorio de cualquier evento que se produzca en él.

9 Conclusión

Desde el punto de vista de las funcionalidades cualquier gestor que seleccionemos nos aporta lo mismo, por tanto la selección de uno u otro se debe centrar en los resultados de rendimiento en disco. Analizando estos resultados la elección parece clara. Nexus es el mejor seguido de

Por favor, vota +1 o compártelo si te pareció interesante

Share |

0

También puedes seguirnos a través de:



Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial, podemos ofrecerte un mejor servicio.

Enviar comentario

(Sólo para usuarios registrados)

» **Regístrate** y accede a esta y otras ventajas «

COMENTARIOS



Esta obra está licenciada bajo licencia Creative Commons de Reconocimiento-No comercial 2.5



Impulsores

Comunidad

[¿Ayuda?](#)

sin clicks

0 personas han traído clicks a esta página



powered by [karmacracry](#)

Copyright 2003-2012 © All Rights Reserved | [Texto Completo](#) | [Condiciones de uso](#) | [Banners](#)

