

¿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

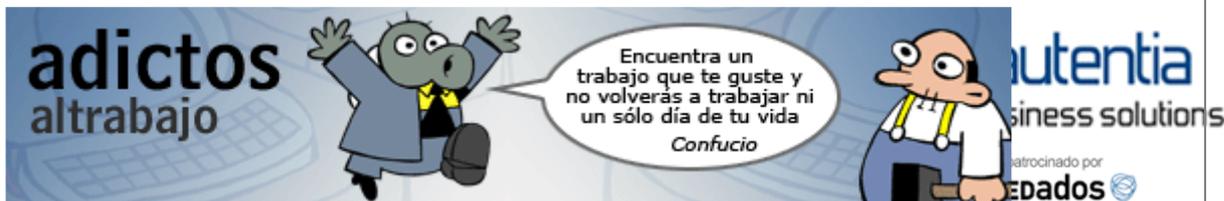
Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

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)



E-mail:

Contraseña:

[Deseo registrarme](#)

[He olvidado mis datos de acceso](#)

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

Estás en: [Inicio](#) [Tutoriales](#) [Cómo evitar tener más de dos cabezas en Mercurial](#)



DESARROLLADO POR:
 Alejandro Pérez García

Alejandro es socio fundador de Autentia y nuestro experto en J2EE, Linux y optimización de aplicaciones empresariales.

Ingeniero en Informática y Certified ScrumMaster

Si te gusta lo que ves, puedes contratarle para darte ayuda con soporte experto, impartir **cursos presenciales** en tu empresa o para que **realicemos tus proyectos como factoría** (Madrid). Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

[Catálogo de servicios Autentia](#)



Últimas Noticias

- [Comic Flash sobre las factorías de software retos y oportunidades](#)
- [Mi primer coderetreat, Chispas!!!](#)
- [Entregamos nuestro primer diploma ...](#)
- [Comic Flash de Head Hunting](#)
- [XI Charla Autentia - Mule - Recordatorio](#)

[Anuncios Google](#) [Tutoriales](#) [Java Software](#) [Flash Intro Tutorial](#)

Fecha de publicación del tutorial: 2009-02-26



Share | [Regístrate para votar](#)

Cómo evitar tener más de dos cabezas en Mercurial

Creación: 16-10-2010

Índice de contenidos

1. Introducción
2. Entorno
3. Configurando un Hook en Mercurial para evitar tener más de una cabeza
4. Conclusiones
5. Sobre el autor

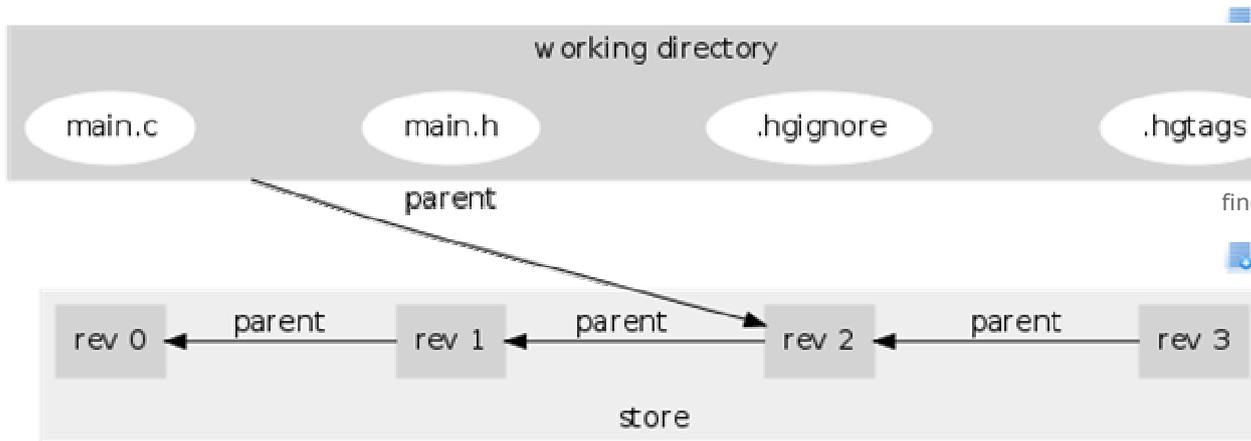
1. Introducción

Mercurial es un repositorio de código distribuido. Cada vez que hacemos un commit en el repositorio se crea una nueva revisión.

[Histórico de NOTICIAS](#)

Últimos Tutoriales

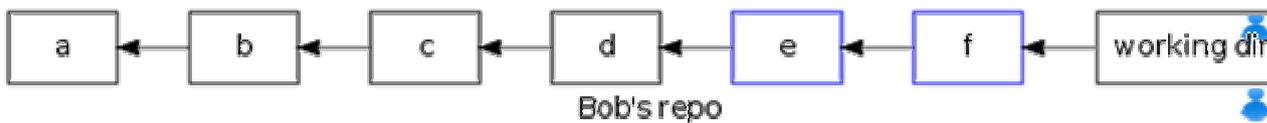
[Reunión Madrid Ágil 14-10-2010: Equipos autogestionados, y motivación del individuo y del equipo](#)



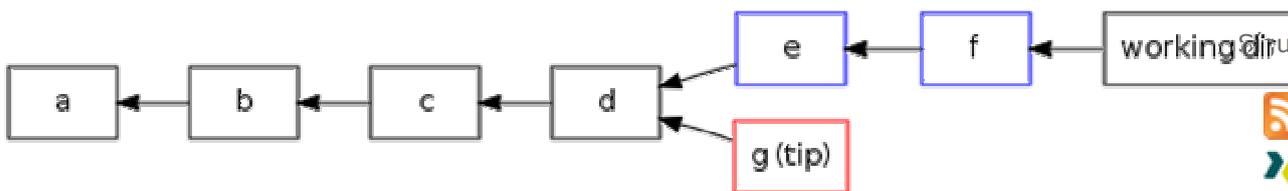
TAQ (Interesting Questions), estado una SPI con jQuery
 .Extension: misos de grano fino en Mercurial
 Google Chrome Developer Toolbar.
 Introducción a HTML5.

Ahora imaginemos que estamos trabajando varios a la vez (lo que viene siendo un equipo ;). En estos casos es muy habitual que, aunque Mercurial es distribuido y cada uno puede ir "por su cuenta", tengamos un repositorio "central" donde al final todos los desarrolladores acabarán subiendo (push) todos sus cambios. Este repositorio central suele estar enlazado con un sistema de integración continua (por ejemplo el Hudson), y el código que aquí se encuentra será el que usemos para sacar a nuestras releases (digamos que el repositorio donde consolidamos los cambios de todos los desarrolladores).

Podría pasar, y de hecho es bastante fácil, que en este repositorio central aparecieran más de una cabeza. Imaginemos que tenemos dos desarrolladores: Alice y Bob. Los dos clonan el repositorio central cuando éste está en la revisión "d". Y se ponen a hacer cambios, de forma que Bob añade la revisión "e" y la "f", mientras que Alice añade la revisión "g".



Si ahora los dos suben los cambios al repositorio central (push) como los dos partieron de la revisión "d", se crearán dos cabezas.



Últimos Tutoriales del Autor

- Reunión Madrid Ágil 14-10-2010: Equipos autogestionados, y motivación del individuo y del equipo
- Spring + REST + JSON = SOAUI

Redescubriendo el Agilismo

Reunión Madrid Ágil 21-09-2010: Estrategias de Branches, y división de una historia en tareas

Cacoo, herramienta colaborativa para hacer diagramas, incluso prototipado de pantallas o UML

Grupos a través de:



Esto no es para nada recomendable en el repositorio central, ya que aquí lo que queremos precisamente es tener una única "línea" de trabajo donde se mezcla (merge) adecuadamente el código de todos los desarrolladores.

Podríamos entrar en el repositorio central y hacer un merge a mano, pero esto es más bien incómodo tratándose de un repositorio central. Además lo que queremos es que cada desarrollador sea responsable de hacer el merge con el código de sus compañeros y garantizar que el resultado funciona (por lo menos debería pasar todos los test ;)

En este tutorial vamos a ver cómo podemos configurar Mercurial para no permitir que exista más de una cabeza en el repositorio central.

Nota: Idea sacada de <http://www.softwareprojects.com/resources/programming/t-mercurial-hook-forbid-2-heads-1910.html>

Últimas ofertas de empleo

2010-08-30
 Otras - Electricidad - BARCELONA.

2010-08-24
 Otras Sin catalogar - LUGO.

2010-06-25
 T. Información - Analista / Programador - BARCELONA.

2. Entorno

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 17' (2.93 GHz Intel Core 2 Duo, 4GB DDR3 SDRAM, 128GB Solid State Drive).
- NVIDIA GeForce 9400M + 9600M GT with 512MB
- Sistema Operativo: Mac OS X Snow Leopard 10.6.4
- Mercurial 1.6.2

3. Configurando un Hook en Mercurial para evitar tener más de una cabeza

Mercurial, al igual que otros tantos repositorios de código, dispone de un sistema de eventos, de forma que cuando se hacen operaciones sobre el repositorio podemos enganchar un **Hook** (garfio) a este evento, para realizar ciertas operaciones. Es muy típico, por ejemplo configurar un hook para que, en cuento se hace commit se llame al sistema de integración continua.

Vamos a utilizar esta funcionalidad para que cuando se haga push en el repositorio central, se ejecute un script que compruebe el número de cabezas, y si es superior a 1, abortaremos el push y daremos un mensaje de error.

Lo primero que vamos a hacer es copiar el siguiente script en el directorio de extensiones de Mercurial del repositorio central (puede ser cualquier directorio, ya que luego indicaremos la ruta en el fichero de configuración). Por ejemplo, lo podemos poner en `/var/lib/hg/hgext/forbid_2head/forbid_2head.py`

```
01 # win32text.py - LF <-> CRLF translation utilities for Windows users
02 #
03 # This software may be used and distributed according to the terms
04 # of the GNU General Public License, incorporated herein by reference.
05 #
06 # To forbid pushes which creates two or more headss
07 #
08 # [hooks]
09 # pretxnchange group.forbid_2heads = python:forbid2_head.forbid_2heads
10
11 from mercurial import ui
12 from mercurial.i18n import gettext as _
13
14 def forbid_2heads(ui, repo, hooktype, node, **kwargs):
15     if len(repo.heads()) > 1:
16         ui.warn(_('Trying to push more than one head, try run "hg merge"
before it.n'))
17     return True
```

Ahora en el fichero de configuración de Mercurial (el `/etc/mercurial/hgrc`) del repositorio central añadimos:

```
1 [hooks]
2 pretxnchange group.forbid_2heads = python:forbid_2head.forbid_2heads
3
4 [extensions]
5 forbid_2head=/var/lib/hg/hgext/forbid_2head/forbid_2head.py
```

A partir de ahora si al hacer un push va a provocar que exista más de una cabeza, el repositorio central abortará la operación y nos mostrará el mensaje "Trying to push more than one head, try run "hg merge" before it.", evitando así posibles despistes de los desarrolladores.

4. Conclusiones

Como digo siempre, es fundamental conocer las herramientas que manejamos y automatizar todos los procesos posibles para evitar despistes que luego nos pueden hacer perder mucho tiempo.



Alejandro Pérez
alejandropparci

ACOJONANTE: yo quiero un simulador de estos para jugar en casa!!!
<http://ow.ly/2UMIX>
15 hours ago

All artists are good craftsmen, but not all craftsmen are good artists. Pure craft addresses an external purpose, art does not.
3 days ago

Si cuando haces la comunión te regalan unos evangelios, cuando te matrículas de informática deberían regalarte el Clean Code
12 hours ago



Join the conversation

Con este tutorial hemos visto cómo configurar un pequeño Hook de Mercurial para forzar a los miembros del equipo a mezclar su código con el del resto de compañeros antes de hacer la subida al repositorio central donde tendremos configurado el servidor de integración continua.

5. Sobre el autor

Alejandro Pérez García, Ingeniero en Informática (especialidad de Ingeniería del Software) y Certified ScrumMaster.

Socio fundador de Autentia (Formación, Consultoría, Desarrollo de sistemas transaccionales).

<mailto:alejandropg@autentia.com>

Autentia Real Business Solutions S.L. - "Soporte a Desarrollo".

<http://www.autentia.com>

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

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, 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-Sin obras derivadas 2.5

Copyright 2003-2010 © All Rights Reserved | [Texto Completo](#) | [Condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) |

