

Control de versiones con Mercurial

¿Qué es el control de versiones?

El control de revisiones es el proceso de administrar diferentes versiones de una pieza de información.

¿Que es mercurial?

Mercurial es un sistema de control de versiones multiplataforma , para desarrolladores de software. Está implementado principalmente haciendo uso del lenguaje de programación Python , pero incluye una implementación binaria de diff escrita en C . Mercurial fue escrito originalmente para funcionar sobre GNU/Linux. Ha sido adaptado para Windows , Mac OS X y la mayoría de otros sistemas tipo Unix .

Características de mercurial

Mercurial cuenta con un conjunto único de propiedades que lo hacen una elección particularmente buena como sistema de control de revisiones, puesto que:

- Es fácil de aprender y usar.
- Escala de forma excelente.
- Es fácil de acondicionar.

Si los sistemas de control de revisiones le son familiares, debería estar listo para usar Mercurial en menos de cinco minutos. Si no, sólo va a tomar unos pocos minutos más. Las órdenes de Mercurial y su conjunto de características son uniformes y consistentes generalmente, y basta con que siga unas pocas reglas generales en lugar de un montón de excepciones.

En un proyecto pequeño, usted puede comenzar a trabajar con Mercurial en pocos momentos. Crear nuevos cambios y ramas, transferir cambios (localmente o por la red); y las operaciones relacionadas con el estado y el historial son rápidas. Mercurial buscar ser ligero y no incomodar en su camino combinando poca sobrecarga cognitiva con operaciones asombrosamente rápidas.

La utilidad de Mercurial no se limita a proyectos pequeños: está siendo usado por proyectos con centenas de miles de contribuyentes, cada uno conteniendo decenas de miles de ficheros y centenas de megabytes de código fuente Si la funcionalidad básica de Mercurial no es suficiente para usted, es muy fácil extenderlo. Mercurial se comporta muy bien con tareas de scripting y su limpieza interna junto con su implementación en Python permiten añadir características fácilmente en forma de extensiones. Hay un buen número de extensiones útiles y populares en este momento, desde ayudar a identificar fallos hasta mejorar su desempeño.

¿Porqué es bueno trabajar con mercurial?

Hay muchas razones por las cuales usted o su equipo desearía usar una herramienta como mercurial para un proyecto.

- Llevar registro del historial y la evolución de su proyecto, para evitar hacer la tarea manualmente. Por cada cambio, tendrá una bitácora de quién lo hizo; por qué se hizo; cuándo se hizo; y de qué se trataba el cambio.
- Puede ayudarle a recuperarse de equivocaciones. Si aplica un cambio que posteriormente se evidencia como un error, puede revertirlo a una versión previa a uno o muchos ficheros. De hecho, una herramienta realmente buena, incluso puede ayudar efectivamente a darse cuenta exactamente cuándo se introdujo el error.
- Le ayudará a trabajar simultáneamente, y a manejar las diferencias entre múltiples versiones de su proyecto.
- Trabajar de manera colaborativa

Mercurial VS. Git

Comparado con Git, Mercurial tiene un fuerte enfoque hacia la facilidad. En términos de rendimiento, Git es extremadamente rápido. En muchos casos, es más rápido que Mercurial, por lo menos en Linux, mientras que Mercurial se comporta mejor en otras operaciones. De todas maneras en Windows, el desempeño y el nivel general de soporte que ofrece Git, al momento de la escritura, está bastante atrás de Mercurial. Mientras que el repositorio de Mercurial no requiere mantenimiento, el repositorio de Git requiere frecuentes “reempaquetados” de sus metadatos. Sin estos, el desempeño se degrada y el uso de espacio crece rápidamente. Un servidor que

contenga repositorios de Git que no sean reempacados rigurosa y frecuentemente requerirá trabajo intenso de disco durante las copias de seguridad, y ha habido situaciones en copias de seguridad diaria que toman más de 24 horas como resultado. Un repositorio recién reempacado de Git es un poco más pequeño que un repositorio de Mercurial, pero un repositorio sin reempacar es varios órdenes de magnitud más grande.

El corazón de Git está escrito en C. Muchas órdenes de Git están implementadas como guiones de línea de comandos o de Perl y la calidad de esos guiones varía ampliamente. He encontrado muchas situaciones en las cuales los guiones no tuvieron en cuenta la presencia de errores que podrían haber sido fatales. Mercurial puede importar el historial de revisiones de un repositorio de Git.

Mercurial vs SVN

Principales diferencias:

- Las modificaciones se realizan de forma atómica (o se actualiza el fichero entero o no se actualiza nada).
- SVN es diferente de Mercurial, porque se trata de un único repositorio que todos los usuarios tienen que guardar sus cambios.
- Mercurial tiene un modelo distribuido. Esto significa que existe un repositorio en cada equipo y que hay repositorio de "Oficial" que la gente elige para confirmar sus cambios.

Referencias:

Mercurial: The definitive guide, Bryan O'Sullivan, 2009.

Agustín Aguilar