

# Trunk-Based Development

Daniel Álvarez Blanco

Eloy Martín Reig

¿Qué es?

El trunk-based development es un modelo de ramas usado en el control de versiones que busca agilizar todos los procesos por los que pasa tanto la agregación como la modificación en un proyecto, es decir, agilizar la implementación, commits, revisión, validación y release. Nació en los años noventa y es uno de los modelos más usados a día de hoy en los controles de versiones además de ser unos de los mejores.

¿Cómo se lleva a cabo?

Al contrario que otros modelos basados en la creación de múltiples ramas, el trunk-based busca unir todas las ramas en una llamada “tronco” donde trabajaran todos los desarrolladores. Esta puede ser tanto la rama “master” o una rama derivada de ella. Además su objetivo es realizar commits pequeños pero abundantes y sucesivos facilitando así la comprobación y validación del código puesto que al ser los commits más pequeños resulta más fácil revisarlos. Al aumenar los commits también se ve incrementado el feedback sobre el código y se pueden arreglar los cambios erróneos antes.

Tipos

1. Solo un tronco

Este es el primer estilo que surgió del trunk-based y, como bien indica su nombre, busca que todos los desarrolladores trabajen sobre el mismo tronco. Este estilo es ideal para grupos pequeños entorno 10 personas.

2. Un tronco y varias ramas de corta duración

Este fue el segundo estilo en aparecer cerca de los dos mil. Al contrario que el anterior busca que sus desarrolladores creen ramas de corta duración (1-3 días) para la implementación de los cambios para tener así la rama principal más despejada y arreglar el problema de los conflictos al dos desarrolladores pushear en la misma rama. Este estilo es perfecto para trabajar con 100+ desarrolladores en un mismo proyecto.

### 3. Un tronco y una patch queue

Impulsado por Google ya entrados en los dos mil, este estilo busca que los desarrolladores trabajen sobre la patch queue así como lo harían en las ramas, pero sin necesidad de crearlas. La patch queue permite que los desarrolladores tengan control sobre los cambios que se producen sobre el proyecto ya que para implementarse necesita que varios hayan aprobado su integración.

#### Ventajas

1. Al disminuir el tamaño de los commits se facilita la revisión
2. Al aumentar los commits aumenta el feedback
3. Esta siempre listo para una release al trabajar directamente sobre el tronco
4. Las ramas al vivir menos facilitan los merges ya que son cambios pequeños
5. Funciona bien en ambientes ajustados porque tienes a todos los desarrolladores pendientes de los cambios

#### Desventajas

1. Requieres a todos los stakeholders involucrados en constantemente en el desarrollo
2. Es difícil transicionar al modelo puesto que es común que se utilicen viejos hábitos y no se lleve a cabo correctamente
3. Necesita una revisión constante de los cambios
4. Los desarrolladores necesitan ser polivalentes y no solo especializarse en un ámbito del proyecto

#### Conclusión

En resumen, el trunk-based development es un modelo usado en el control de versiones ágil muy exigente con los desarrolladores, pero con un gran impacto ya que permite estar al día con el proyecto y ejecutar cambios y ajustes cuasi instantáneos además de poder realizar una release muy rápida para estar al día con los competidores.