

Trunk-Based Development

Álvaro García Fernández - UO289295

Zohaib Akhtar Kausar - UO291060

Alberto Lago Conde - UO288245

Contenidos

1. Git
2. Modelos de ramificación
3. *Trunk Based Development* (TBD)
 1. Introducción
 2. Desarrollo
 3. Principios
 4. Uso de ramas
4. Diferencias entre modelos
 1. Ventajas y desventajas del TBD
5. Aplicaciones del TBD

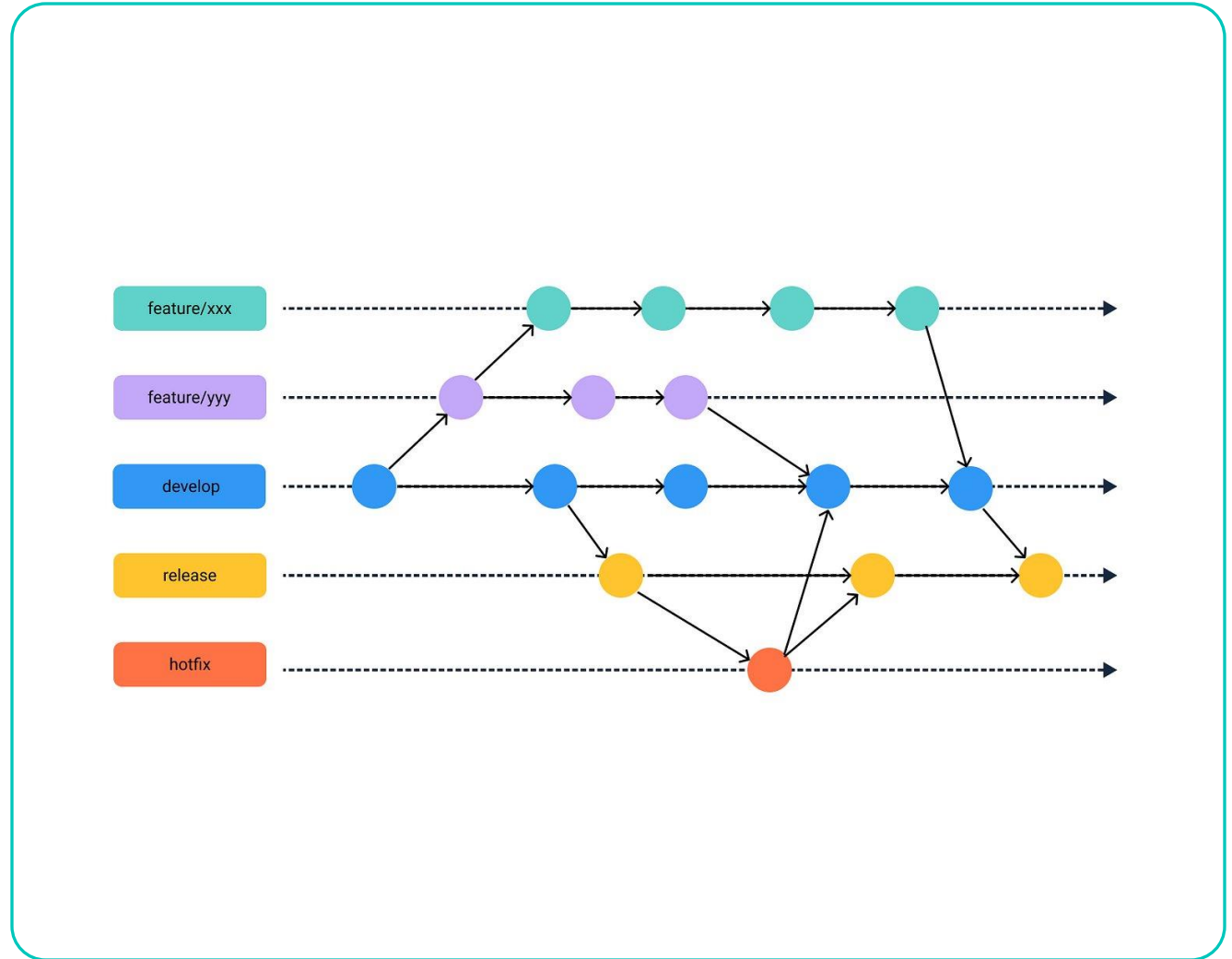
Git

- Control de versiones
- Año **2005**



Modelos de ramificación

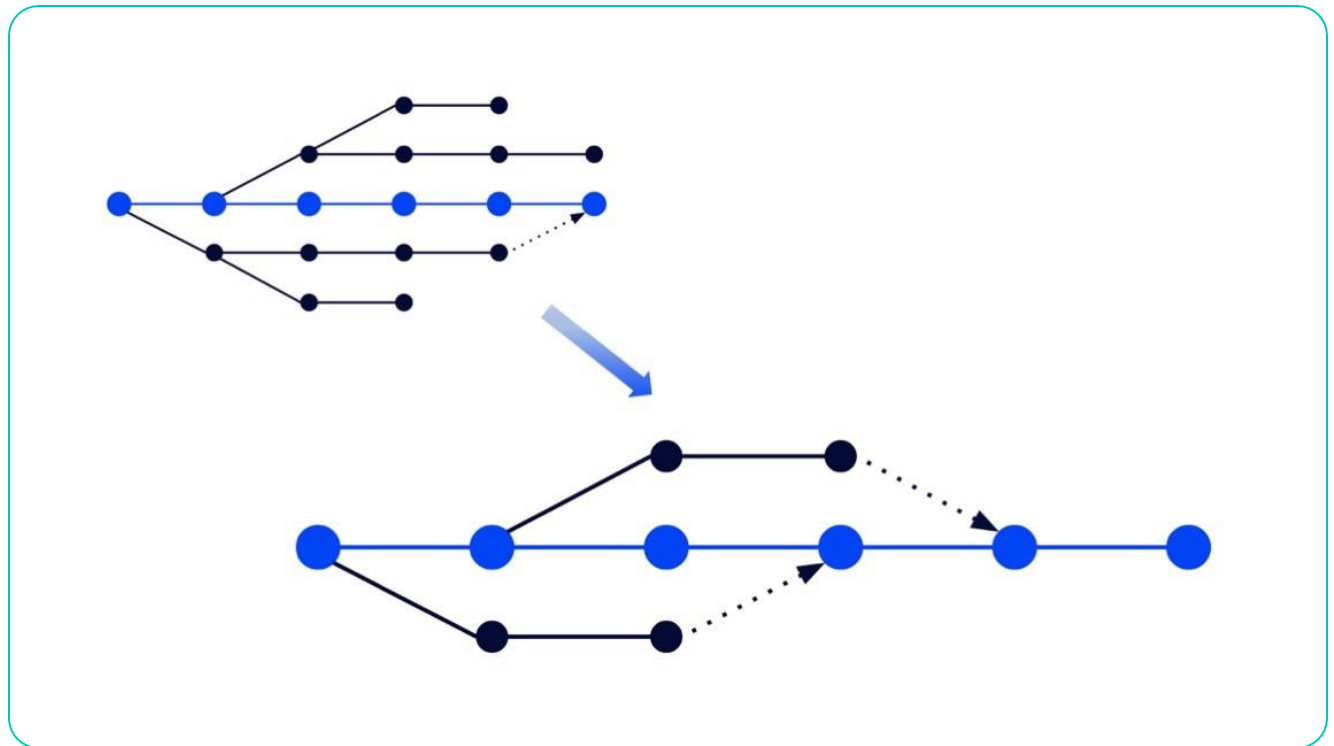
- Ramas
- **Git Flow**
- Trunk Based



Trunk Based Development

Introducción

- Años 90
- 1 rama
- Ahorrar tiempo
- Revisión



Entorno

- **Comunicación y colaboración** fundamentales
- **Adaptabilidad** a cambios constantes y menores
- **Flexibilidad y resiliencia** frente **conflictos de fusión**
- **Personas** con habilidades **polivalentes**



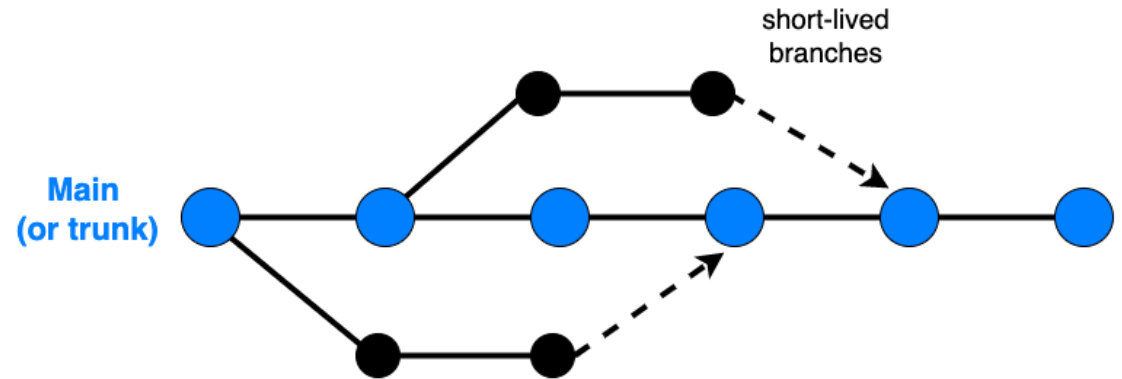
Principios

- Trabajar en Tareas pequeñas y bien definidas
- Code reviews regularmente
- Utilizar pruebas automatizadas
- Utilizar un sistema de integración continua



Uso de ramas

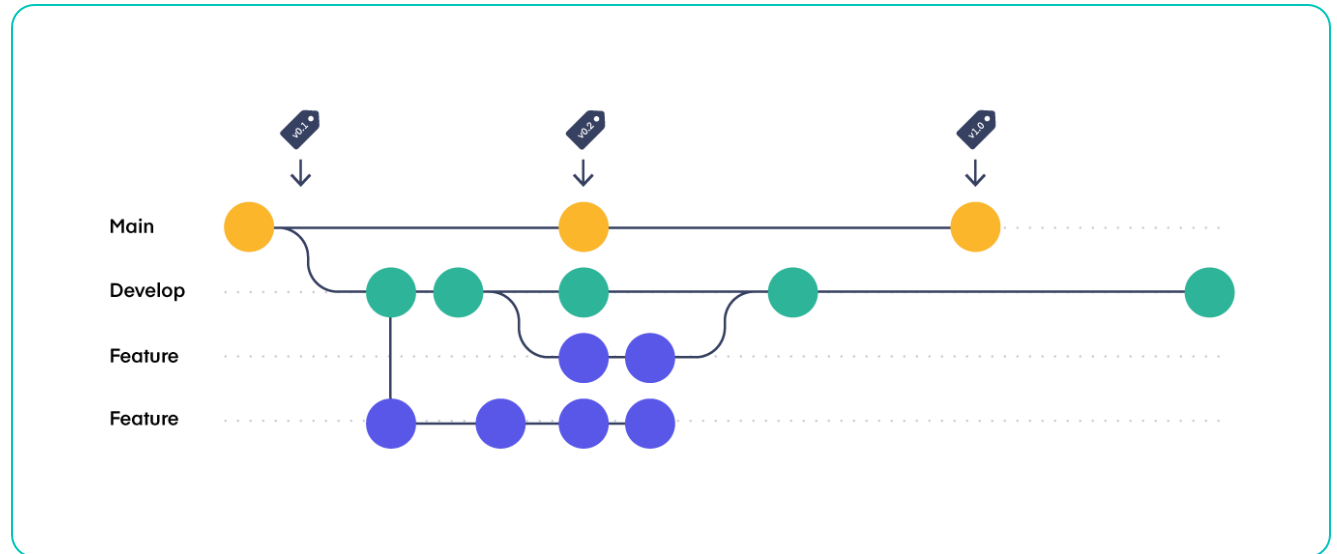
- **Propósito de las ramas:**
Crear branches para agregar nuevas características o solucionar errores.
- **Características de las ramas:**
 - Corta duración
 - Fusionar con la rama principal mediante “pull request”



Diferencias entre modelos

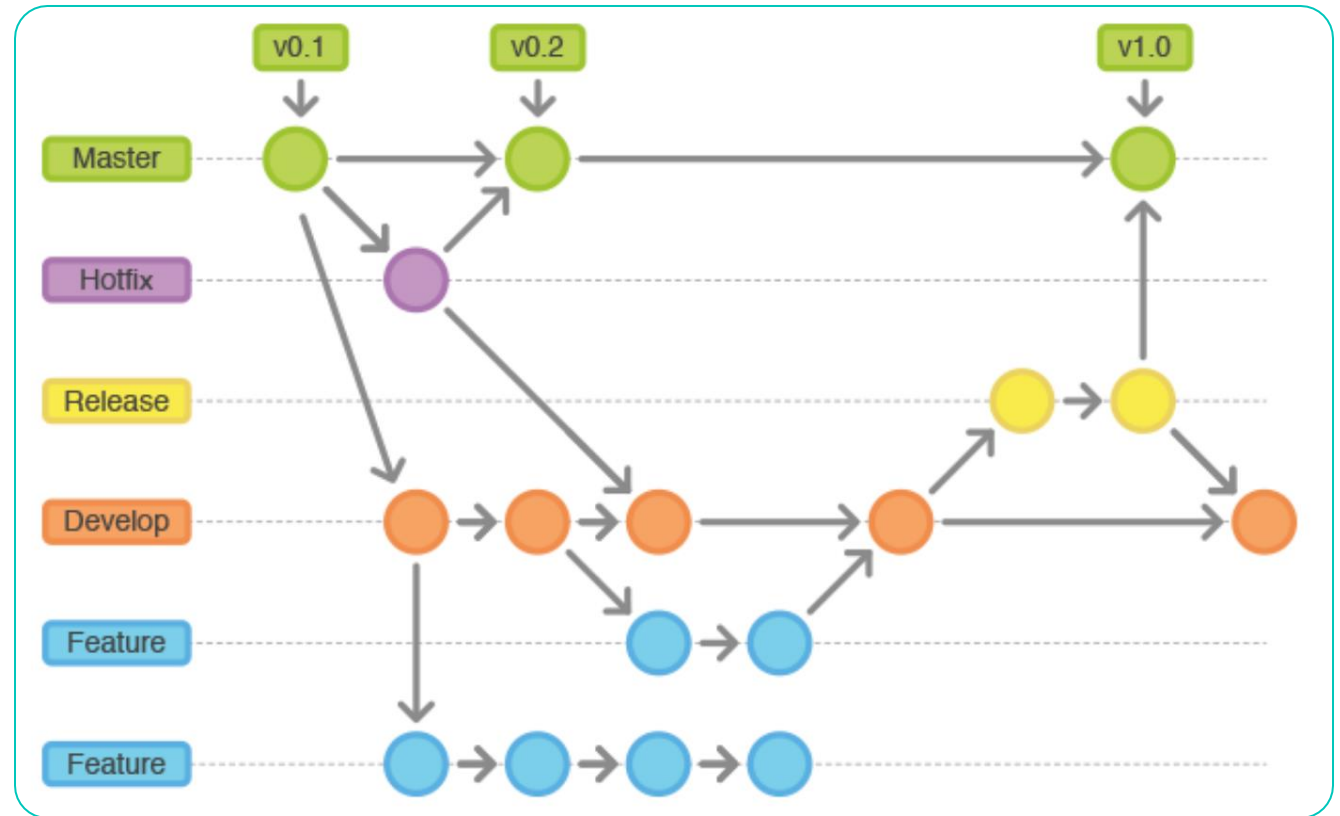
Feature Branching

- Cada característica en una rama
- Permite aislar cambios
- Puede haber un gran número de ramas



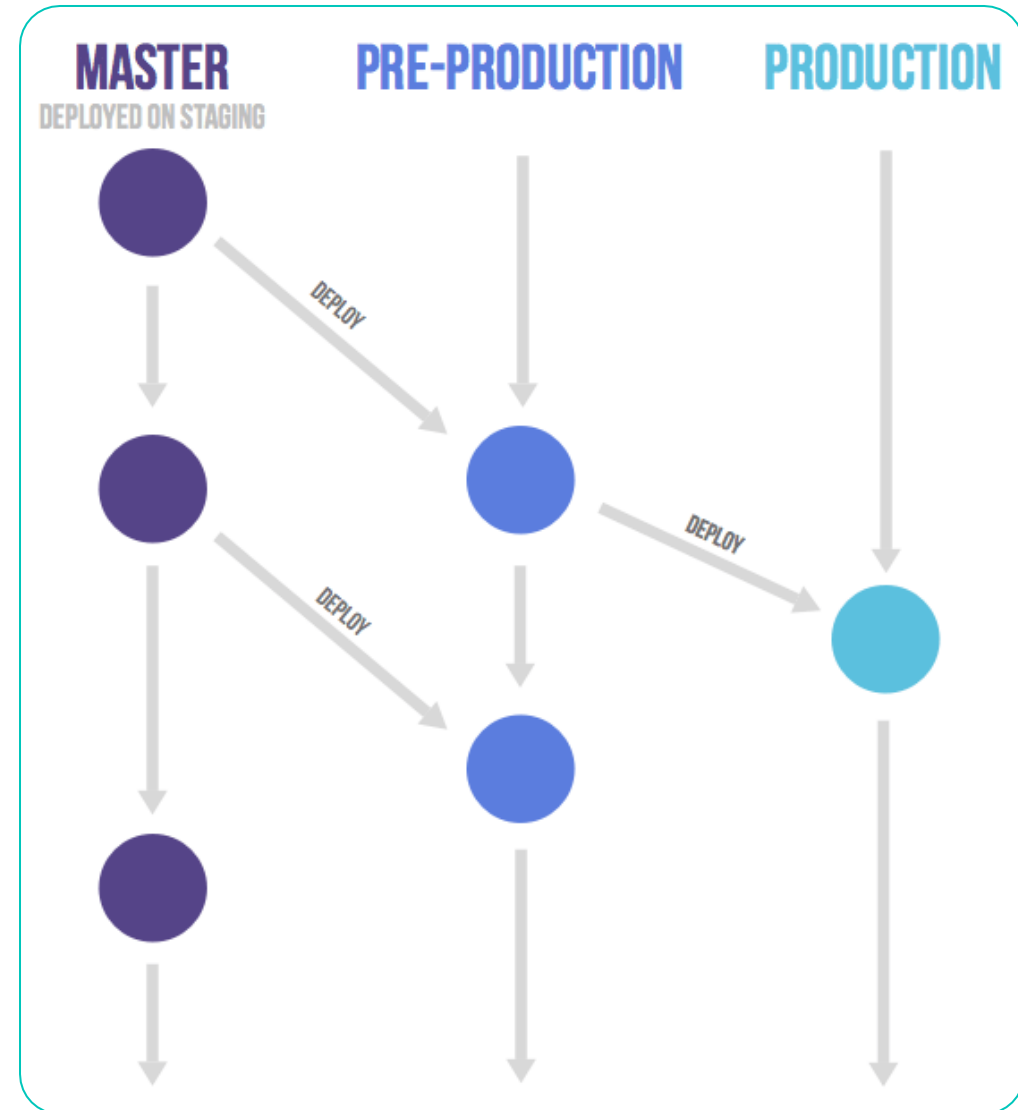
GitFlow

- Ramificación muy específica
- Estructura clara
- Puede volverse complejo



GitLab Flow

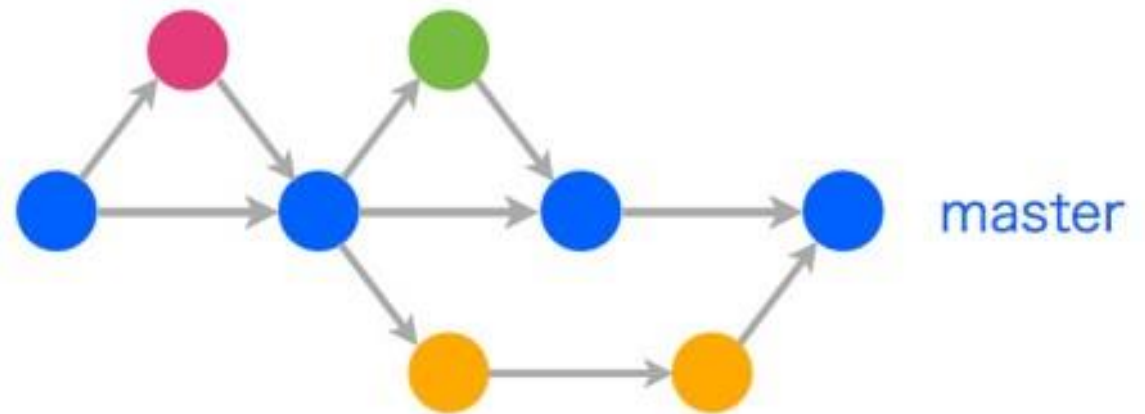
- Versión simplificada de GitFlow
- Facilita la colaboración
- Puede tener problemas de integración



GitHub Flow

- Simplificación de GitFlow
- Para equipos pequeños
- Fácil de entender y usar
- Necesarias pruebas exhaustivas

GitHub flow



Ventajas y desventajas

Ventajas

- Integración Continua
- **Continuous Delivery**
- Menos ramas a mantener
- Menos posibilidad de **Merge Hell**



Desventajas

- Riesgo de cambios no probados
- Presión sobre la revisión de código
- Necesidad de pruebas exhaustivas
- No es apto para todo tipo de proyectos



Aplicaciones del TBD

- Proyectos con *Continuous Delivery*
- Proyectos pequeños / medianos
- Necesidad de nuevas características constantes
- Negocios emergentes / *startups*
- Otros
 - Proyectos I+D
 - Aplicaciones web
 - Servicios *cloud*





FIN

¿Preguntas?