

Ship, Show, ? Ask

Práctica de Git aplicada en el proyecto

M. Fischer, G. Larrosa, G. Martinez

Trabajo colaborativo

Nuestras dudas...

- Si las revisiones de código y trabajo colaborativo se tratan de:
 - Mejorar la calidad de código
 - Compartir conocimientos
- Entonces:
 - ¿Deberían ser necesarias siempre?
 - ¿Todos los cambios tienen que pasar por revisión?
 - ¿Todos tienen margen de mejora o aportan nuevos conocimientos para compartir?



Puede que si o puede que no...

¿Por qué?

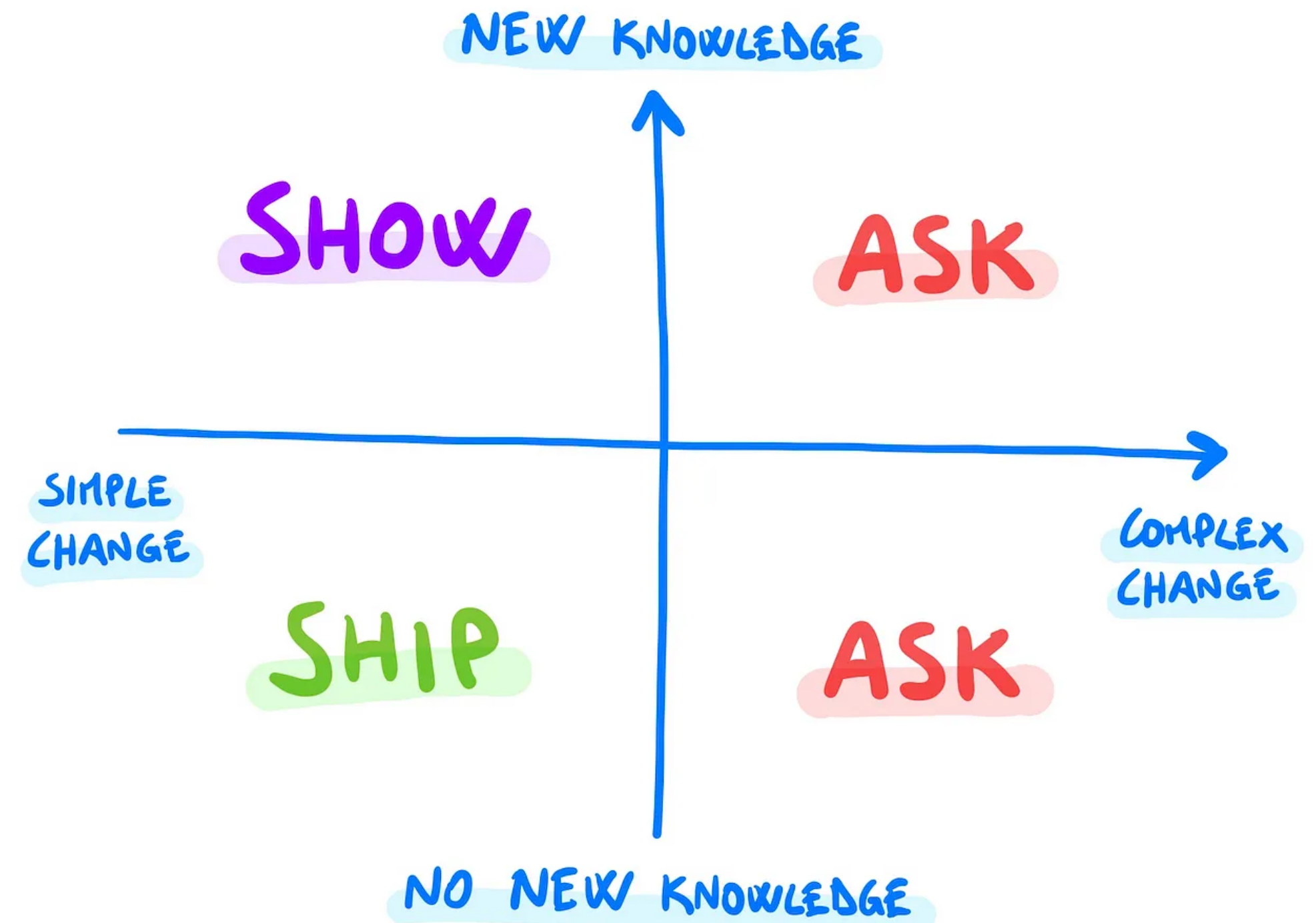
Distintos casos

- 🚢 Ship — Realiza el cambio directamente en su línea principal, sin solicitar una revisión del código. Esto funciona cuando corrige errores sin importancia, agrega funciones utilizando patrones establecidos o realiza cambios colaterales, como mejorar documentos.
- 🔍 Show — Uno mismo crea un PR, ejecuta toda la canalización de CI, la fusiona sin la aprobación de nadie y luego solicita comentarios. De esta manera, se libera rápidamente pero aún crea espacio para la discusión. Funciona en situaciones en las que se desea compartir conocimientos pero no necesariamente necesita comentarios, o los comentarios son valiosos pero no deberían bloquear.
- ? Ask — Se realiza cambios en una sucursal, abre un PR y espera la revisión antes de fusionarse. Este es el proceso estándar que la mayoría de las empresas adoptan hoy en día.



Comparativa

Cuadrado mágico



BREVE DEMO

**En nuestra vida profesional, casi siempre trabajaremos en modo "Ask".
No nos sentiríamos cómodo con el modo "Ship" a menos que estemos hablando de cambios realmente pequeños e insignificantes.
Sin embargo, vemos que el patrón "Show" podría ser una buena manera de obtener lo mejor de ambos mundos en muchos escenarios.**

- Nuestra reflexión

FIN