



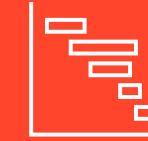
MANUAL DE SOLUCIÓN DE CONFLICTOS EN GITOPS - AUTOMATIZADOS

Autor: Ricardo.Garcia.Atanacio@kyndryl.com

09-2025



Agenda



Identificación de
conflictos en GitOps



Herramientas de
diagnóstico



Pasos para la resolución
de conflictos



Mejores Prácticas

Kyndryl

Identificación de Conflictos

09-2025





¿CUALES SON LOS CONFLICTOS EN GITOPS?



Definición de conflictos en GitOps

Los conflictos ocurren cuando el estado deseado en Git diverge de la infraestructura o aplicaciones desplegadas, causando inconsistencias.



Tipos Comunes de conflictos

Los conflictos comunes incluyen conflictos de ramas, errores de fusión, archivos de configuración corruptos y fallos en la ejecución de la pipeline.



Impacto en las implementaciones

Estos conflictos pueden causar fallos en el despliegue, errores de sincronización y reducir la estabilidad y fiabilidad del sistema.



Importancia de la Detección Temprana

Reconocer los conflictos temprano ayuda a mantener flujos de trabajo de GitOps estables y fiables, y previene el tiempo de inactividad.

Kyndryl

Diagnóstico de Conflictos

09-2025



HERRAMIENTAS DE DIAGNÓSTICO

Dashboard bitbucket

The screenshot shows the Bitbucket dashboard for a pull request. At the top, there's a navigation bar with 'Bitbucket', 'Projects', 'Repositories', and a search bar. Below the bar, a pull request card is displayed for 'Javier Orlando Hurtado Lopez - KYNDRYL' with the ID 'release/mvl00222741' merging into 'master'. The status is 'MERGED'. The title of the PR is 'INICIO DE PASE A PRODUCCIÓN | Cambio Menor | G338 | MVPLEGBCP-222741'. Below the title, there are tabs for 'Overview', 'Diff', and 'Commits', with 'Overview' being the active tab. A 'Details' section follows, showing the user who created the PR and the date ('07 Aug 2025').

Logs de Ejecución de git

The screenshot shows a terminal window titled 'MINGW64:c/Users/S76039/Documents/mvl00222741/Automation_git_conflict'. It displays the contents of a file named 'gitops_activity.log'. The log shows several messages from a user 'S76039' running a script. Key messages include:

```
S76039@pterminalp09 MINGW64 ~/Documents/mvl00222741/Automation_git_conflict
$ cat gitops_activity.log
[Mon Nov 10 16:41:51 SAPST 2025] Message problem BITBUCKET: This pull request ca
n t be merged checkout 'release/mvl00222741'
[Mon Nov 10 16:41:52 SAPST 2025] Solution
[Mon Nov 10 16:41:52 SAPST 2025] Starting GitOps script for user 'S76039'
[Mon Nov 10 16:41:53 SAPST 2025] Directory 'S76039' created.
[Mon Nov 10 16:41:57 SAPST 2025] Set git user.name='S76039' and user.email='rgar
ciao@bcn.com.pe'.
Cloning into 'pepricerspotusdpen'...
warning: ----- SECURITY WARNING -----
warning: | TLS certificate verification has been disabled! |
warning: -----
warning: HTTPS connections may not be secure. See https://aka.ms/gcm/tlsverify f
or more information.
Updating files: 100% (9693/9693), done.
warning: the following paths have collided (e.g. case-sensitive paths
on a case-insensitive filesystem) and only one from the same
colliding group is in the working tree:
```

Kyndryl

Resolución de conflictos

09-2025



```
52
53 # Set git user and email config
54 git config --global user.name "$GIT_USER"
55 git config --global user.email "$GIT_EMAIL"
56 echo "[$(date)] Set git user.name='$GIT_USER' and user.email='$GI"
57
58 # Clone repo
59 git clone "$REPO_URL" 2>&1 | tee -a "../$LOGFILE"
60 REPO_NAME=$(basename "$REPO_URL" .git)
61 cd "$REPO_NAME"
62
63 # Checkout target branch
64 echo "[$(date)] Checkout target branch $CHECKOUT_BRANCH:" | tee -a
65 git checkout "$CHECKOUT_BRANCH" 2>&1 | tee -a "../$LOGFILE"
66
67 # Pull changes from pull branch
68 # Strategy-option --theirs is used to perform a git pull operation
69 echo "[$(date)] Pull changes from pull $CHECKOUT_BRANCH:" | tee -a
70 git pull --strategy-option theirs origin "$PULL_BRANCH" 2>&1 | tee -a
71
72 # Show diff between branches
73 echo "[$(date)] Diff between origin/$CHECKOUT_BRANCH and $CHECKOU"
74 git diff origin/"$CHECKOUT_BRANCH" "$CHECKOUT_BRANCH" --name-stat
75
76 # Push changes to remote
77 git push origin "$CHECKOUT_BRANCH" 2>&1 | tee -a "../$LOGFILE"
78
79 # Log HEAD commit
80 git log HEAD -n 1 2>&1 | tee -a "../$LOGFILE"
81
82 echo "[$(date)] Finished operations: checkout '$CHECKOUT_BRANCH',
```

Pasos Automatizados



Identificación de conflictos

Identifique conflictos revisando los registros de Git o bitbucket para detectar cambios problemáticos.



Análisis de Commit y Archivos

Analiza el último commit y los archivos afectados como ejemplo test.json y Test.sql en busca de errores.



Validación del Estado

Valida el estado del pedido



Corrección y Verificación

Revierte o corrige el commit defectuoso, vuelve a ejecutar gitflow y verifica el estado de finalización del pedido.

Shared: Copiar
\\pfilep09\\sharefilepam\\Ricardo\\Automation_git_conflict

Destino: Pegar directorio local

Editar archivo:
gitops_parameters.yaml

Configuración de parámetros

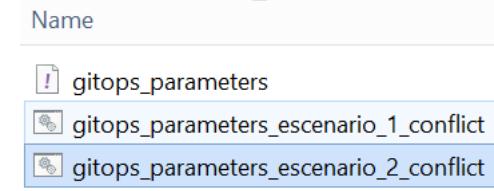
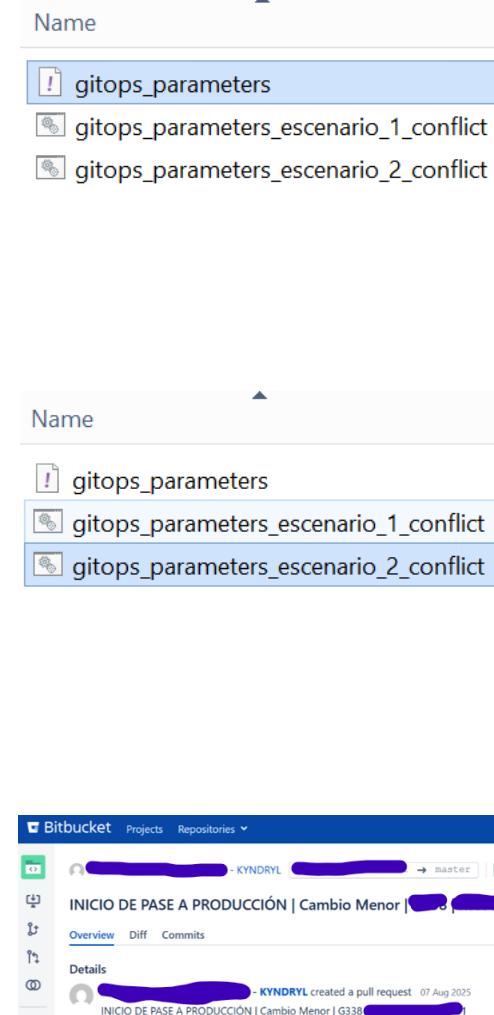
GIT_USER: "matricula"
GIT_EMAIL: "mail@bcp.com.pe"
REPO_URL: "<https://github.com/ricardo685/template.git>"
CHECKOUT_BRANCH: "master"
PULL_BRANCH: "release/xxxxxx"
LOGFILE: "gitops_activity.log"

Example environment file for GitOps script with prompt password input

GIT_USER: "T76039"
GIT_EMAIL: "joseperez@bcp.com.pe"
REPO_URL: "<https://github.com/ricardo685/test.git>"
CHECKOUT_BRANCH: "master"
PULL_BRANCH: "release/MVL0012345"
LOGFILE: "gitops_activity.log"

Name	Date modified	Type
gitops_parameters	10/11/2025 04:37 p.m.	Yaml Source File
gitops_parameters_escenario_1_conflict	28/09/2025 08:24 p.m.	Shell Script
gitops_parameters_escenario_2_conflict	28/09/2025 08:30 p.m.	Shell Script

Gitops Automatizado



Configurar:
gitops_parameters.yaml

1

Ejecutar archivo:
gitops_parameters_resce
nario_1_conflict.sh

2

gitops_parameters_resce
nario_1_conflict.sh

3

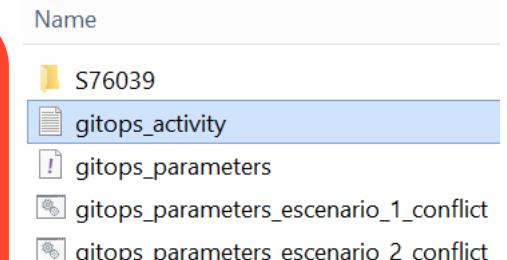
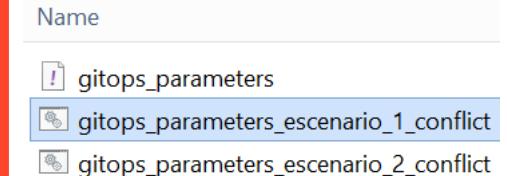
Nota: Solo
ejecutar en caso paso 2
falla.

Validar archivo log:
Gitops_activity

4

Validar bitbucket

5



Kyndryl

Mejores Prácticas

09-2025





Buenas prácticas

1

Validar commits

Antes del Commit. Ayuda a prevenir errores de sintaxis y configuración temprano en el proceso.

2

Automatizar gitops

La automatización de validaciones minimiza errores humanos y acelera el proceso de verificación.

3

Documentar cambios

Documentar cambios y coordinar despliegues mejora la trazabilidad y previene conflictos en los flujos de trabajo del equipo.

Kyndryl

Gracias

09-2025

