

Rollback

Rollback

“ Dernière mise à jour : 2026-05-10

Stratégie

La CI Gitea push 2 tags par build :

- `latest` (mutable, suit toujours le dernier commit `main`)
- `sha-<7chars>` (immuable)

Pour revenir à une version antérieure : pull et utilise un tag `sha-<7chars>`.

Étapes

1. Identifier le commit cible

Sur Gitea : `gitea.thymon.fr/thymon/boardgame-referee/commits/branch/main`. Note le SHA court (7 premiers chars) du commit auquel tu veux revenir.

Ou en CLI :

```
git log --oneline -20
# 2ff9b71 security(ssh): isolate Claude SSH on dedicated 'oracle' user
# 95229af perf(rag): parallelize Qdrant searches + adopt withTimeout
# ...
```

2. Modifier docker-compose.yml

```
services:
  app:
-   image: gitea.thymon.fr/thymon/boardgame-referee:latest
+   image: gitea.thymon.fr/thymon/boardgame-referee:sha-95229af
```

3. Pull + recreate

```
docker compose -f /mnt/user/appdata/boardgame-referee/docker-compose.yml pull app
docker compose -f /mnt/user/appdata/boardgame-referee/docker-compose.yml up -d --force-recreate app
```

4. Vérifier

```
docker logs -f --tail 50 boardgame-referee
curl https://rules.thymon.fr/api/health
```

Rollback BDD ?

⚠ Si la version cible a un schéma SQLite différent, rollback de l'image **ne rollback pas la BDD**. Si tu as joué une migration `0011_xxx.sql` puis tu rollback à une version qui ne la connaît pas :

- Au mieux : la migration est vue comme déjà appliquée (Drizzle skip) et l'app fonctionne mais avec des colonnes en trop dans la table → généralement pas de souci si la requête `select *` ne dépend pas du schéma exact.
- Au pire : du code récent référençait une nouvelle colonne que l'ancienne version ne lit pas → erreur silencieuse ou crash.

Backup SQLite avant migration :

```
docker exec boardgame-referee sqlite3 /app/data/database.db ".backup
/app/data/database.db.bak.$(date +%F)"
# Puis avant rollback, restore :
docker exec boardgame-referee cp /app/data/database.db.bak.<date> /app/data/database.db
docker compose ... restart app
```

Rollback Qdrant ?

Qdrant ne fait pas de migrations — les payloads sont schemaless. Donc un rollback de l'image n'impacte pas Qdrant.

Si tu as ré-ingéré des PDFs avec une nouvelle version du chunker (qui produit des chunks différents), rollback de l'image ne rollback pas les chunks. Si nécessaire :

```
# Snapshot Qdrant AVANT migration (recommandé)
curl -X POST http://192.168.10.100:6333/collections/rules_<slug>/snapshots
# (renvoie un nom de snapshot)

# Restore après rollback
curl -X PUT http://192.168.10.100:6333/collections/rules_<slug>/snapshots/recover \
  -H "Content-Type: application/json" \
  -d '{"location": "<snapshot-name>"}'
```

Rollback NPM config ?

NPM versionne sa propre config dans `database.sqlite` (interne). Si tu as cassé une route ou l'TLS, restaure depuis ton backup `/mnt/user/appdata/nginx-proxy-manager/`.

En pratique

Pour boardgame-referee, le risque "rollback obligatoire" est faible :

- Pas de données utilisateur critiques (ce que tu perds = des questions/réponses RAG, pas des transactions)
- Schéma SQLite stable depuis quelques migrations
- Qdrant peut être ré-ingéré si besoin

Donc rollback simple = changer le tag d'image dans `docker-compose.yml` et redémarrer.

Revision #1

Created 2026-05-10 15:20:01 UTC by thymon

Updated 2026-05-10 15:20:01 UTC by thymon