

# Cronologia delle revisioni

Versione	Data	Autore	Descrizione delle Modifiche
1.0.0	2025-09-02	Antonio Musarra	Prima release
1.0.1	2025-09-16	Antonio Musarra	Correzioni minori

- Cronologia delle revisioni
- Prerequisiti
- Uso Base
- Esempi
  - Elencare i repository disponibili (sorgente)
  - Migrazione con filtro glob
  - Migrazione con filtro regex
  - Dry-run (simulazione, nessuna modifica)
  - Migrazione di una lista di repository
  - Forzare il push su repo già esistente
  - Abilitare il trace (debug)
- Note
- Supporto

Questo script consente di migrare repository Git tra progetti/organizzazioni Azure DevOps, con supporto a filtri, lista repo, dry-run e mirror push. Inoltre include un comando per elencare rapidamente tutti i repository disponibili nella sorgente (--list-repos), utile per costruire filtri o file lista e per validare PAT/permessi prima della migrazione.

Un classico caso d'uso è la migrazione di tutte le repo di un progetto sorgente in un progetto destinazione, ad esempio per consolidare progetti o spostare repo tra organizzazioni o meglio servizi ICT che sono mappati su team project.

## Prerequisiti

- Bash >= 4
- jq >= 1.6
- curl >= 8.x
- git
- Variabili d'ambiente:
  - `SRC_PAT` (Personal Access Token sorgente, scope: Code Read)
  - `DST_PAT` (Personal Access Token destinazione, scope: Code Read, Write & Manage)

## Uso Base

```
1 ./migrazione_git_repo.sh --src-org myorg --src-project MyProject \
2   --dst-org targetorg --dst-project TargetProject \
3   --filter 'ansc-*
```

### Console 1 - Uso base con filtro glob

A seguire l'esempio di output per il comando sopra che esegue la migrazione dei repository che matchano il filtro `ansc-*` dal progetto `ansc-tool-test-coop-services` dell'organizzazione `amusarra` al progetto `ansc-tool-test-coop-services-mirror` dell'organizzazione `amusarra`. In questo caso l'organizzazione è la stessa (sia sorgente che destinazione), ma potrebbe essere diversa.

```
1 >>> Sorgente:      amusarra/ansc-tool-test-coop-services
2 >>> Destinazione:  amusarra/ansc-tool-test-coop-services-mirror
3 >>> Filtro (regex): ansc-*
4
5 Repository da migrare (1):
6   - ansc-tool-test-coop-services
7
8 Usando directory temporanea:
9   /Users/amusarra/dev/tools/ansc/git/migrazione/tmp_migrazione_git_13809
10 === Repo: ansc-tool-test-coop-services ===
11 Clono (mirror) dal sorgente...
12 Creo la repo in destinazione: ansc-tool-test-coop-services
13 Push --mirror verso destinazione...
14 OK.
```

```

14
15 >>> Completato.
16
17 ===== RIEPILOGO MIGRAZIONE =====
18 Repository                                | Esito                                | Azure URL
19 -----+-----+-----
20 ansc-tool-test-coop-services              | OK                                  |
    https://user:***@dev.azure.com/amusarra/ansc-tool-test-coop-services-
    mirror/_git/ansc-tool-test-coop-services |
    https://user:***@dev.azure.com/amusarra/ansc-tool-test-coop-services-
    mirror/_git/ansc-tool-test-coop-services
21 =====

```

Output 1 - Uso base con filtro glob

## Esempi

### Elencare i repository disponibili (sorgente)

Questo comando elenca i repository disponibili nel progetto sorgente e termina senza effettuare alcuna migrazione. È utile per:

- costruire velocemente il filtro (--filter) a partire dai nomi reali,
- generare un file lista (--repo-list) con i repository da migrare,
- validare PAT e permessi (errori 401/403 vengono mostrati esplicitamente),
- fare troubleshooting con --trace.

```

1 ./migrazione_git_repo.sh --src-org myorg --src-project MyProject --list-repos
2 # oppure con abbreviazioni e trace
3 ./migrazione_git_repo.sh -so myorg -sp MyProject --list-repos --trace

```

Console 2 - Elencare i repository disponibili (sorgente)

Esempio di output sintetico:

```

1 Repository disponibili in myorg/MyProject:
2
3 - ansc-core
4   cloneUrl: https://dev.azure.com/myorg/MyProject/_git/ansc-core
5   webUrl:   https://dev.azure.com/myorg/MyProject/_git/ansc-core
6 - ansc-tool-test-coop-services
7   cloneUrl: https://dev.azure.com/myorg/MyProject/_git/ansc-tool-test-coop-services
8   webUrl:   https://dev.azure.com/myorg/MyProject/_git/ansc-tool-test-coop-services

```

Output 2 - Lista dei repository disponibili (sorgente)

Nota:

- Richiede la variabile d'ambiente SRC\_PAT impostata (scope: Code Read).
- È possibile usare --trace anche con --list-repos per diagnosticare problemi di autenticazione o rete.

## Migrazione con filtro glob

```
1 | ./migrazione_git_repo.sh -so srcorg -sp SrcProj -do dstorg -dp DstProj -f 'ansc-*
```

Console 3 - Uso base con filtro glob

## Migrazione con filtro regex

```
1 | ./migrazione_git_repo.sh -so srcorg -sp SrcProj -do dstorg -dp DstProj -f '^ansc-
(core|svc)-.*$'
```

Console 4 - Uso base con filtro regex

## Dry-run (simulazione, nessuna modifica)

Questo comando mostra le repo che verrebbero migrate senza eseguire alcuna operazione. Usando questa modalità possiamo verificare quali repo verrebbero migrate prima di eseguire la migrazione e anche il dettaglio delle operazioni che verrebbero eseguite.

```
1 | ./migrazione_git_repo.sh -so srcorg -sp SrcProj -do dstorg -dp DstProj -f '^ansc-.*$'
--dry-run
```

Console 5 - Dry-run (simulazione, nessuna modifica)

A seguire l'esempio di parte dell'output:

```
1 | ...
2 | >>> Sorgente:      amusarra/ansc-tool-test-coop-services
3 | >>> Destinazione: amusarra/ansc-tool-test-coop-services-mirror
4 | >>> Filtro (regex): ansc-*
5 | >>> Modalità: DRY-RUN (nessuna creazione/push in destinazione)
6 |
7 | Repository da migrare (1):
8 |   - ansc-tool-test-coop-services
9 |
10 | Usando directory temporanea:
    /Users/amusarra/dev/tools/ansc/git/migrazione/tmp_migrazione_git_19737
11 | === Repo: ansc-tool-test-coop-services ===
12 | [DRY] Clonerei in mirror dal sorgente:
13 |     git clone --mirror 'https://user:***@dev.azure.com/amusarra/ansc-tool-test-
    coop-services/_git/ansc-tool-test-coop-services'
    '/Users/amusarra/dev/tools/ansc/git/migrazione/tmp_migrazione_git_19737/ansc-tool-
    test-coop-services.git'
14 | [DRY] Creerei la repo in destinazione: ansc-tool-test-coop-services
15 | [DRY] Eseguirei push --mirror verso destinazione:
```

```

16      (cd
    '/Users/amusarra/dev/tools/ansc/git/migrazione/tmp_migrazione_git_19737/ansc-tool-
test-coop-services.git' && git remote add dest
'https://user:***@dev.azure.com/amusarra/ansc-tool-test-coop-services-
mirror/_git/ansc-tool-test-coop-services' && git push --mirror dest)
17  ...

```

Output 3 - Dry-run (simulazione, nessuna modifica)

## Migrazione di una lista di repository

Puoi specificare una lista di repository da migrare usando l'opzione `--repo-list`.

Crea un file `repo_list.txt` con i nomi delle repo (una per riga):

```

1  ansc-core
2  ansc-svc
3  ansc-tool-test-coop-services

```

File 1 - Contenuto di `repo_list.txt` che contiene i nomi delle repo da migrare

Esegui lo script:

```

1  ./migrazione_git_repo.sh -so srcorg -sp SrcProj -do dstorg -dp DstProj --repo-list
repo_list.txt

```

Console 6 - Migrazione di una lista di repository

## Forzare il push su repo già esistente

Può essere utile se la repo esiste già in destinazione e si vuole sovrascrivere il contenuto (ad esempio per retry di una migrazione fallita).

**Importante:** Il processo aziendale definito in SOGEI, impone che la creazione dei repository deve avvenire tramite il [Portale ALM](#) usando l'apposita funzionalità "Nuovo Repository". Occorre assicurarsi che i repository da migrare esistano già in destinazione prima di eseguire lo script con l'opzione `--force-push`. Lo script è capace di creare il repository sulla destinazione solo se non esiste già e se si hanno i permessi necessari ma nel caso di Team Project gestiti tramite il Portale ALM, la creazione deve avvenire tramite il Portale stesso.

Senza `--force-push`, se la repo esiste già in destinazione, il push non viene eseguito e la migrazione di quella repo viene saltata.

```

1  Repository da migrare (da file) (3):
2    - ansc-tool-non-esistente
3    - ansc-tool-test-coop-services
4    - ansc-tool-non-esistente-1
5
6  Usando directory temporanea:
  /Users/amusarra/dev/tools/ansc/git/migrazione/tmp_migrazione_git_3553
7  === Repo: ansc-tool-non-esistente ===

```

```

8      Clono (mirror) dal sorgente...
9  remote: TF401019: The Git repository with name or identifier ansc-tool-non-esistente
    does not exist or you do not have permissions for the operation you are attempting.
10 fatal: repository 'https://dev.azure.com/amusarra/ansc-tool-test-coop-
    services/_git/ansc-tool-non-esistente/' not found
11  Errore: repository sorgente non trovato o accesso negato: ansc-tool-non-esistente
12  Non creo la repo in destinazione. Continuo con il prossimo repository.
13
14  === Repo: ansc-tool-test-coop-services ===
15  Repo già presente in destinazione.
16  Push NON eseguito (repo già presente). Usa --force-push per forzare.

```

Output 4 - Output nel caso di repo già esistente

Con `--force-push`, il push viene eseguito anche se la repo esiste già in destinazione, sovrascrivendo il contenuto.

```

1  ./migrazione_git_repo.sh -so srcorg -sp SrcProj -do dstorg -dp DstProj -f 'ansc-*' --
    force-push

```

Console 7 - Forzare il push su repo già esistente

## Abilitare il trace (debug)

Per abilitare il trace (debug) e vedere i comandi eseguiti, usare l'opzione `--trace`.

```

1  ./migrazione_git_repo.sh ...altri parametri... --trace

```

Console 8 - Abilitare il trace (debug)

## Note

- Il filtro accetta sia glob (`ansc-*`) che regex (`^ansc-.*$`).
- In dry-run non vengono create repo né eseguiti push.
- Il mirror replica tutti i branch/tag e rimuove quelli cancellati.
- Se la repo esiste già in destinazione, il push NON viene eseguito a meno di usare `--force-push`.
- La directory temporanea usata per il clone viene cancellata automaticamente.

## Supporto

Per problemi o richieste, contattare: Antonio Musarra ([amusarr@sogei.it](mailto:amusarr@sogei.it))