

# Gfal2 HTTP Copy

## A look into the fallback mechanism

Mihai Patrascoiu  
*DOMA BDT (18th May 2022)*



# Gfal2 HTTP Control flags

# Enable “3rd pull” and “3rd push”

**ENABLE\_REMOTE\_COPY = <true|false>**

# Enable HTTP → HTTP “streamed”

**ENABLE\_STREAM\_COPY = <true|false>**

# Enable fallback (3rd pull → 3rd push → streamed)

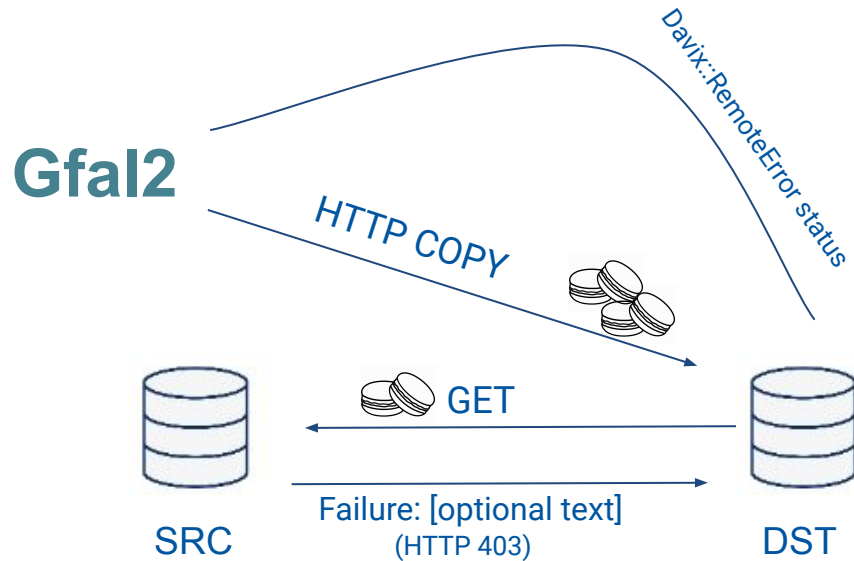
**ENABLE\_FALLBACK\_TPC\_COPY = <true|false>**

# Where to start in the fallback mechanism (fallback order is fixed)

**DEFAULT\_COPY\_MODE = <3rd pull | 3rd push | streamed>**

# Gfal2 Fallback mechanism

```
{ 3rd pull, 3rd push, streamed }
```



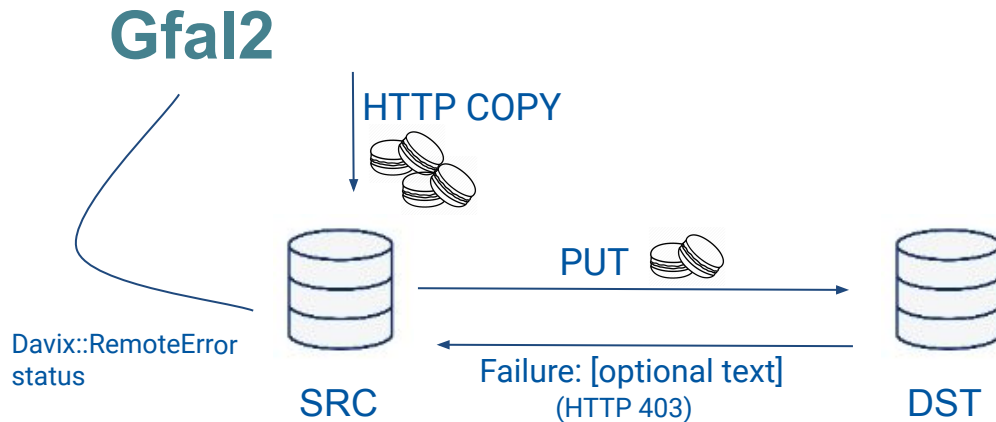
Gfal2 gets back **Davix::RemoteError** status

Is this retry-able? **Can't tell**

Move to the next mode!

# Gfal2 Fallback mechanism

```
{ 3rd pull, 3rd push, streamed }
```



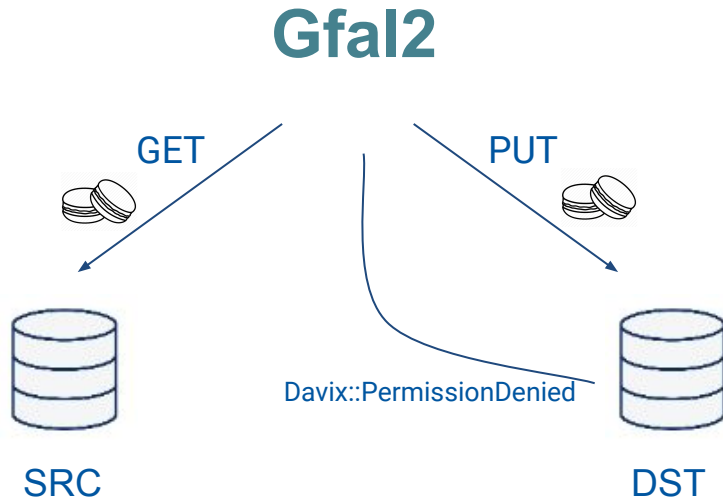
Gfal2 gets back **Davix::RemoteError** status

Is this retry-able? **Can't tell**

Move to the next mode!

# Gfal2 Fallback mechanism

```
{ 3rd pull, 3rd push, streamed }
```



Gfal2 gets back **Davix::PermissionDenied**

Is this retry-able? **No**

Do not continue fallback  
(technical detail: this is the last one anyway)

# Proposal

*Extend HTTP-TPC Protocol to also report the “HTTP Code” when reporting a TPC failure*

## Benefits:

- Prevents doomed-to-fail fallback in Gfal2
- Improves FTS & MONIT error reporting  
(failed FTS transfer reported as streamed because of fallback)
- Removes additional work from FTS/Gfal2 + Storages
- Reduces logging/reporting clutter

## *Why not parse in Gfal2?*

1. Having Gfal2 parse the error code is too ad-hoc:

### EOS error reporting:

```
Remote side failed with status code 403
```

### dCache error reporting:

```
rejected GET: 401 token authorization failed
```

2. Opinion: Good addition to the protocol (HTTP-TPC is here to stay)