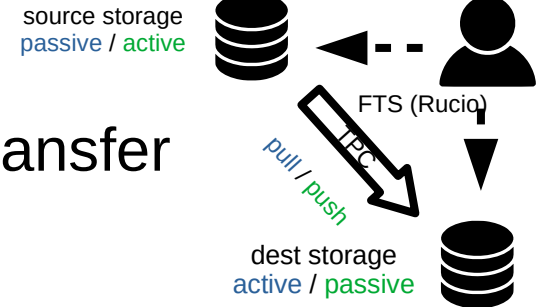


TAPE and SRM+HTTP TPC

- dCache & StoRM rely on SRM for “bringonline”
 - Data asynchronously staged in buffer + normal transfer
 - srmPrepareToGet + srmPrepareToPut
 - bulk operation, **support protocol negotiation**
 - client sends list of protocols → first supported by server used
 - server translate SURL to TURL using selected/preferred protocol
 - gfal-copy works – client waits for staging (timeout)
 - TURL_PROTOCOLS – not clear where it is used
 - TURL_3RD_PARTY_PROTOCOLS – used for 2+3 party copy
 - **dCache** (FZK) provides **gsiftp, http and root TURL** (TPC works)
 - **StoRM** (INFN-T1) provides **only** with **gsiftp TURL**
- **(CERN) CTA** use multihop with local EOS instance
 - tape → (small EOS) buffer → normal EOS instance → remote site
 - **two FTS transfers** with CERN ATLASDATADISK in the middle
 - completely transparent – **any EOS TPC protocol** can be used



TAPE and SRM+HTTP TPC

- Gfal HTTP transfers with (macaroon) token
 - not clear how to pass macaroon token to unknown TURL
 - only SRM SURL is known before we start TPC
 - credentials must be stored before TPC starts
 - remote → dCache tape (pull) or dCache tape → remote (push)
 - X.509 proxy to the dCache could be delegated
 - ask all TAPE dCache sites to support gridsite delegation service(?)
 - dCache tape → remote (pull) or remote → dCache tape (push)
 - if remote site doesn't support gridsite this can't work by design
 - only option – mixture of push and pull and gridsite service on dCache
 - could be solved by OIDC tokens ? FTS SRM support and globus EOL ?
 - depends how we restrict scope (path) – TURL necessary before TPC?
 - does dCache SRM+HTTP works with OIDC tokens?
- FTS – fine once we resolve gfal issues fts-transfer-submit –bring-online 3600 ...
 - testbed with different TURL_3RD_PARTY_PROTOCOLS (?)

