**Repo types/use cases/etc.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Repo type acronym | Repo type | Update in place – con -current write | Marfs Stripes across multiple Repo Stripes | MarFS Access method | Exists | Use | Example Deploy |
| CW.D | Direct (File on File) | Yes | No, backing store has to be a PFS | Direct posix so posix security works | In code, not tested well | Staging | GPFS MD, Same GPFS Data |
| CW.S | Semidirect (File on remote File) | Yes | No, backing store has to be a PFS | Remote posix, so need remote secure posix access method | No requires remote fs access method | Staging | GPFS MD, Diff GPFS Data |
| NCW.NS.OO | Non Striped Object on Object | No | No, object server must be global and scalable (global means all storage servers can see all objects in this repo) | Standard like CDMI, Sproxyd etc., so uses web security | Yes | Long term | GPFS MD, Scality single ring Data |
| NCW.NS.GFS | Non Striped Object on File | No | No, back end must be scalable and global (but not parallel (within a single file) | Remote object (could be just web service, or WebHDFS or something similar, uses web security or similar | No, requires remote object access method only | Long Term | GPFS MD, GPFS as object Data |
| NCW.S.OO | Striped Object on Object | No | Can stripe across multiple back object systems | Standard like CDMI, Sproxyd etc., so uses web security | No, requires striping | Long Term | GPFS MD, multi Scality Rings Data |
| NCW.S.MFS | Striped Object on File | No | Can stripe across multiple remote file systems | Remote object (could be just web service, or WebHDFS or something similar, uses web security or similar | No, requires striping and remote object access method | Long Term | GPFS MD, multi ZFS2 Data |

**Deployment methods in order of appeal (INITIAL CUT) (risk mitigations**)

|  |  |  |
| --- | --- | --- |
| Deploy | Pro | Con |
| NCW.NS.OO (GPFS MD, Scality single ring Data) | Code most complete  Supports MarFS Effort  Easy to Migrate (under MarFS) | Performance efficiency  Other? |
| NCW.S.OO (GPFS MD, Multiple Scality Ring Data and MarFS stripes across these) | Supports MarFS Effort  Easy to Migrate (under MarFS) | Requires Striping  Performance efficiency?  Other? |
| NCW.NS.GFS (GPFS MD, GPFS as scalable global object server), The data GPFS sits on ZFS2 | Supports MarFS Effort  Easy to Migrate (under MarFS)  Presumably performance efficient | Need remote Object over FS access method  Unsupported GPFS data volume |
| NCW.S.MFS (GPFS MD, multiple individual ZFS2 for Data and MarFS stripes across these) | Supports MarFS Effort  Easy to Migrate (under MarFS)  Presumably performance efficient  It is very likely this will be a long term feasible deployment str | Need remote Object over FS access method  Requires striping |
| CW.S (GPFS MD, GPFS remote file on file for data, Data GPFS on ZFS2 | Presumably performance efficient  Easy to migrate (under MarFS)  Supports MarFS effort  Semidirect would be nice for staging at some point | Need remote File over FS access method  Unsupported GPFS data volume  Requires semidirect code |
| CW.D (GPFS MD and Data, GPFS on ZFS2 | Direct code is in place  Presumably performance efficient  Relatively easy to migrate (under MarFS but lacks indirection)  Supports MarFS effort | Direct code not tested  Unsupported GPFS data volume  No separation of MD and Data |
| Direct GPFS use on top of ZFS2 luns (using indirection top level directory to make merging with MarFS later feasible. | No coding/tested at some scale  Presumably performance efficient  Hard to migrate (may be able to “merge under MarFS later” | Unsupported GPFS data volume  No data/metadata separation  Does not support MarFS effort |

Ways to provide remote access

Remote Object over FS( NCW.NS.GFS and NCW.S.MFS) WebHDFS or just a web server set up to get/put files, hacked IOFSL to add security (web style security) would be way more efficient but requires some coding probably, NFS or other remote mount has security issues that would need to be dealt with

Remote File over FS (semidirect CW.S): NFS or other remote mount has security issues that would need to be dealt with, IOFSL hacked for security (web style security) would be very efficient