

Database Cloning

DFSR database cloning is an optional alternative to so-called classic initial replication. By providing each downstream server with an exported copy of the upstream server's database and preseeded files, DFSR reduces or eliminates the need for over-the-wire metadata exchange. DFSR database cloning also provides multiple file validation levels to ensure reconciliation of files added, modified, or deleted after the database export but before the database import. After file validation, initial sync is now instantaneous if there are no differences. If there are differences, DFSR only has to synchronize the delta of changes as part of a shortened initial sync process.

DFSR Database Cloning is an optional alternative to the classic initial sync process introduced in Windows Server 2003 R2. DFSR spends most of its time in initial sync—even when administrators preseed files on the peer servers—examining metadata, staging files, and exchanging version vectors. This can make setup, disaster recovery, and hardware replacement very slow. Multi-terabyte data sets are typically infeasible due to the extended setup times; the estimate for a 100TB dataset is 159 days to complete initial sync on a LAN, if performance is linear (spoiler alert: it's not).

DB cloning bypasses this process. At a high level, you:

1. Build a primary server with no partners (or use an existing server with partners)
2. Clone its database
3. Preseed the data

on N servers

4. Build N servers using that database clone

The existing initial sync portion of DFSR is now instantaneous if there are no differences. If there are differences, DFSR only has to catch up the real delta of changes as part of a shortened initial sync process.

Cloning provides three levels of file validation during the export and import processing. These ensure that if you are allowing users to alter data on the upstream server while cloning is occurring, files are later reconciled on the downstream.

- **None** - No validation of files on source or destination server. Fastest and most optimistic. Requires that you preseed data perfectly and do not allow any modification of data during the clone processing on either server.
- **Basic** - (Default behavior). Hash of ACL stored in the database record for each file. File size and last modified date-time stored in the database record for each file. Good mix of fidelity and performance.
- **Full** - Same hashing mechanism used by DFSR during normal operations. Hash stored in database record for each file. Slowest but highest fidelity (and still faster than initial sync)