

Introducing NEO

NEO is a distributed, NoSQL, database based on the Zope Object Database, that allows concurrent access from multiple processes (cores/CPUs/machines). Key objectives of NEO are scalability and resiliency to hardware failures. Both are achieved through parallelism by scattering multiple instances of NEOs internal nodes on different machines. A master node (backed up with spares) orchestrates data distribution through administration of a partition table, which is accessed by all nodes of a cluster.

NEO is the default database for [ERP5](#) and a number of other solutions [Nexedi](#) has created and is currently creating.

Key Changes

The new version 1.4 includes a change in the SQL tables format to fix potential crashes of storage nodes when storing values that only differ by the compression flag. Details can be found in the UPGRADE notes in case you think your application might be affected by this bug.

Performance improvements and new features

The following new features and improvements have been added:

- The "Importer" storage backend speed has been improved significantly
- The MySQL storage backend now supports [TokuDB](#). The engine can be selected via the new *neostorage* option. By default, InnoDB is still used as engine.
- Another new option added is called *neomaster*. It will automatically start a new cluster if the number of pending storage nodes is greater than or equal to the value specified for this option.

Bugfixes

The following bugs have been fixed:

- Storage crashed on reading empty transactions. This has been fixed but we still need to decide whether NEO should continue storing such instructions, ignore them on commit (like in other ZODB implementations) or fail on commit.
- Storage crashed when a client tries to "steal" the UUID of another client.
- Client could get stuck forever on unreadable cells when not connected to master.
- Client could only instantiate NEOStorage from the main thread and the RTMIN+2 signal only displayed logs for one NEOStorage. With the new release RTMIN+2 & RTMIN+3 are setup when neo.client module is imported.
- Various fixes and improvements to logging and debugging.

On using NEO

You can find more information on NEO on the pypy ["neoppod"](#) website with commercial support being provided by [Nexedi](#).