Terracotta 10.3 Release Notes

Terracotta is a distributed in-memory data management solution for both operational storage and analytical processing. Terracotta has powerful query and computation capabilities, leveraging native JDK features such as Java Streams, collections, and functions. It supports the following sub-systems:

- A storage sub-system fronted by TCStore API
- A caching sub-system fronted by the next generation Ehcache API which includes and extends JSR 107

Both sub-systems are backed by the distributed Terracotta Server, which provides a common platform for distributed in-memory data storage with scale-out, scale-up and high availability features.

Feature Highlights

Terracotta 10.3 release introduces key enterprise readiness features and analytical capabilities in Terracotta. With this release, Terracotta took a big leap forward to support multi cloud deployments, security, and operational usability. Some of the notable features of Terracotta 10.3 include:

- Azure certification as a supported platform for Terracotta deployments.
- Terracotta is now available on docker store.
- Security enhancements including Authentication, Authorization and Auditing.
- Support for local transactions in Terracotta through Transaction API extension.
- Write-behind support for clustered Ehcache.
- Hybrid storage support.
- Configuration helpers for simple to use XML based configuration import and export.
- Numerous operational usability enhancements including latency metering, refactoring and role based access in Terracotta Management Console.

Summary of Changes 10.3.1.2

This is a fix release, fully backward compatible with earlier 10.3.x releases.

Resolved

- 4724 - Possible error and failure to complete making a backup when datasets have been deleted and recreated without a server restart
- 4720 - Cell data can be lost when there are tens-of-thousands of unique cell definitions in a dataset
- 4719 - ArrayIndexOutOfBoundsException possible in server/client when there are tens-of-thousands of unique cell definitions in a dataset
- 4687 - TMC does not refresh home page after connection creation, leaving user on blank home page until refresh
- 4685 - TMC display issue with data query explorer view - when switching between datasets being queried, display can contain superfluous cells
- 4591 - Sharded streams terminated with forEach(...) do not self-close, rely upon explicit closure
- 4568 - Possible errors if client disconnects while deleting and re-creating a dataset (with same definition) on a multi-stripe cluster

Known Issues

- When using tens-of-thousands of unique cell definitions within a dataset, performance of write operations degrades

Summary of Changes 10.3.1

New in Terracotta 10.3.1

- Additional deployment and configuration options for High Availability and Consistency features: more flexibility to best meet use case requirements
- it is now also a requirement to explicitly specify a fail-over tuning priority (to favor availability or data consistency) in Terracotta server configurations
- Improvements and new options for Terracotta server start/stop commands to facilitate safer automation of operations
- Significant performance improvements in most CRUD and search operations, as well as in data replication to mirror servers

Resolved

- 4210 - Fixed possible BufferUnderflowException after repeated Cache.put() operations without any interleaved reads
- 3912 - Fixed IllegalStateException "slot starvation" when running many concurrent streams with filter operations with portable predicates
- 3904 - Improve error messaging when client attempts to connect insecurely to a security-enabled server
- 3903 - Fix possible NullPointerException on server when client process crashes
- 3883 - Improve display of data in TMC when there is a very large number of connected clients

Summary of Changes 10.3.0.3

Resolved

- 4249 – Fixed bug related to sending dataset description during replication and cleanup
- 4149 – Implemented a more efficient backoff waiting algorithm for IO during secure handshake in order to remove a performance degradation when secure network communications are enabled.

Summary of Changes 10.3.0.2

Resolved

- 4098 – The external voter now tracks server partition events and takes care not to vote for multiple servers in the same partition event.
- 3989 – Resolved client not reconnecting to newly promoted active after failover. This happened when cluster was configured for consistency mode with external voters.
- 3912 – Resolved an "IllegalStateException: Tx slot starvation!" occurring in TCStore for read-heavy loads. User will most likely encounter this when the server has a limited number of processors available and many RecordStreams are processed. (The number of concurrent reads was limited to 16 times the number of processors.) The correction for this removes that limit for read loads without concurrent updates.

Summary of Changes 10.3.0.1

Resolved

- 3122 - Fixed error to correctly depict handshake error when server is not reachable(due to partition) and TCStore attempts to reconnect
- 3591 - Fixed the display of server status in SPM plugin from 'UNKNOWN' to 'NOT READY' when server is syncing, stuck or suspended.
- 3732 - Added enhancement to emit configuration to log after cluster tool reconfiguration
- 3805 - Resolved issue to make cluster-tool.bat, bcrypt.bat or start-tc-voter.bat executable on windows
- 3829 - Resolved issue when secured server would be allowed to start in case keys in certificates are not generated with RSA algorithm.
- 3834 - Enabled secure connections for Voter
- 3841 - Added warning for users for mismatched port/hosts
- 3873 - Enhanced Cluster tool server-level status command to provide more details during failures
- 3875 - TMC now shows security status for connection between TMS and the cluster
- 3879 - TMS Server at the startup should migrate the tmc.properties from older version by introducing the newly added properties in 10.3 with default values and update the renamed properties
- 3882 - TMC resource panel now displays server status
- 1699 - Ehcache EE now automatically registers its management registry if absent

Notes:

- Terracotta BigMemory 4.x and Terracotta 10.x clients may be used simultaneously in the same application by ensuring ClassLoader isolation when initializing at least one of the clients.
- Beginning with 10.3, users should move to the new 10.3 security properties (tms.security.root.directory, tms.security.https.enabled and tms.security.root.directory.connection.default).