BigMemory Max Upgrade and Migration Guide

Version 4.3

April 2015

This document applies to BigMemory Max Version 4.3 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2010-2015 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at http://softwareag.com/licenses.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at http://softwareag.com/licenses and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: BMM-UPG-43-20150415

Table of Contents

Upgrading from 4.1/4.2 to 4.3.....5
Summary of Changes from 4.1/4.2 to 4.3.....6
Migration Procedure from 4.1/4.2 to 4.3.....7

Upgrading from 4.1 to 4.2.....9
Migration Procedure (from 4.1 to 4.2).....10

Installing Maintenance Version Upgrades for 4.2..... 11
Overview of Maintenance Version Upgrades for 4.2.....12
Preparing for Maintenance Version Upgrades for 4.2.....12
Installing Maintenance Version Upgrades for 4.2.....13
Reverting TSAs to a Previous Version.....15

Upgrading from 3.7 or 4.0.....17
Summary of Changes from 4.0 to 4.1.....18
Summary of Changes from 3.7 to 4.0.....18
Procedure for Migrating from 3.7 or 4.0.....21

1 Upgrading from 4.1/4.2 to 4.3

- Summary of Changes from 4.1/4.2 to 4.3 6
- Migration Procedure from 4.1/4.2 to 4.3 7

Summary of Changes from 4.1/4.2 to 4.3

Changes from 4.2 to 4.3

Area of Change	Change from 4.2 to 4.3
Terracotta Management Console (TMC)	The TMC now provides a "WAN" tab. This enables you to monitor information about the following aspects of the WAN Replication Service: ■ Your WAN's topology and configuration. ■ Each WAN-enabled cache, including performance statistics and details of their deployment, such as the orchestrator topology, configuration, and status.

Changes from 4.1 to 4.2

Area of Change	Change from 4.1 to 4.2
BigMemory Hybrid	Ability to auto tune BigMemory Hybrid to maximize performance.
BigMemory Max	Zero down-time maintenance version upgrade capability.
WAN Replication	Ability to support even higher data limits, providing greater fault tolerance.
WAN Replication	Improved efficiency of WAN Replication management and monitoring to support large number of clients connected to the cluster. REST APIs are provided to monitor deployment.
Terracotta Server Array	New port dedicated to TMC - A new section, , has been added to the Terracotta configuration file. This section specifies the port used by the Terracotta Management Console (TMC) (port 9540, by default). Prior to 4.2, the TMC used the ports specified in /tc:tc-config/servers/server/tsa-port (port 9510 by default) and /tc:tc-config/servers/server/tsa-group-port (port 9530 by default). Now the TMC uses only the port specified in /tc:tc-config/servers/server/management-port.
Terracotta Server Array	JMX port disabled - In the Terracotta configuration file, the section is now disabled by default. To enable it, change the value of `jmx-enabled=` from false to true in the Terracotta

Area of Change	Change from 4.1 to 4.2
	configuration file, tc-config.xml. This section is used to specify the port that the Terracotta server's JMX Connector listens to. The default value of jmx-port is still 9520. If tsa-port is set, this port defaults to the value of the tsa-port plus 10.
Terracotta Server Array	Property deprecated - The `l1.max.connect.retries` property is deprecated. Now, a client is given an unlimited number of attempts to connect.

Migration Procedure from 4.1/4.2 to 4.3

This section describes how to migrate from 4.1 or 4.2 to 4.3. When migrating from 4.1 or 4.2 to 4.3, your existing 4.1/4.2 data is preserved.

Every installation is unique, with many factors that affect migration. This procedure can be used as a checklist of relevant topics and basic migration tasks.

1. Download the BigMemory kit, and verify that you have the correct license keys for the version you are installing.
2. Prepare the new configuration files. Sample configuration files are located in the BigMemory kit in the config-samples/ directory. You can modify the sample files with specific information for your installation.
3. (Optional) If you have persistence enabled, you should make a backup of your data, as described in the *Database Backup Utility* section of your current 4.1/4.2 BigMemory Max documentation.
4. Shut down the server array. A safe shutdown procedure is as follows.
   a. Shut down the mirror servers using the `stop-tc-server` script.
   b. Shut down the clients. A Terracotta client will shut down when you shut down your application.
   c. Shut down the active servers using the `stop-tc-server` script.
5. Clear the servers by removing the old version of BigMemory from each installation directory.
6. Install the new BigMemory kit. You must upgrade all Terracotta clients and servers in the cluster to the same version before restarting the cluster.
7. Start the servers with their new configuration files.

2 Upgrading from 4.1 to 4.2

- Migration Procedure (from 4.1 to 4.2) 10

Migration Procedure (from 4.1 to 4.2)

This section describes how to migrate from 4.1 to 4.2. When migrating from 4.1 to 4.2, your existing 4.1 data is preserved.

Every installation is unique, with many factors that affect migration. This procedure can be used as a checklist of relevant topics and basic migration tasks.

1. Download the BigMemory kit, and verify that you have the correct license keys for the version you are installing.
2. Prepare the new configuration files. Sample configuration files are located in the BigMemory kit in the config-samples/ directory. You can modify the sample files with specific information for your installation.
3. (Optional) If you have persistence enabled, you should make a backup of your data, as described in the *Database Backup Utility* section of the 4.1 BigMemory Max documentation.
4. Shut down the server array. A safe shutdown procedure is as follows.
   a. Shut down the mirror servers using the `stop-tc-server` script.
   b. Shut down the clients. A Terracotta client will shut down when you shut down your application.
   c. Shut down the active servers using the `stop-tc-server` script.
5. Clear the servers by removing the old version of BigMemory from each installation directory.
6. Install the new BigMemory kit. You must upgrade all Terracotta clients and servers in the cluster to the same version before restarting the cluster.
7. Start the servers with their new configuration files.

3 Installing Maintenance Version Upgrades for 4.2

- Overview of Maintenance Version Upgrades for 4.2 12
- Preparing for Maintenance Version Upgrades for 4.2 12
- Installing Maintenance Version Upgrades for 4.2 13
- Reverting TSAs to a Previous Version 15

Overview of Maintenance Version Upgrades for 4.2

This section describes how to upgrade from version 4.2.x to 4.2.y.

This procedure assumes you are currently running a version 4.2 cluster in a high availability deployment configuration. In a cluster without any mirror servers, downtime is unavoidable during an upgrade installation. Additionally, if a cluster without mirrors is a non-persistent cluster, servers cannot be upgraded without losing all the data.

Starting with 4.2, you can now upgrade the versions of all servers and clients in your cluster one node at a time, without shutting down the cluster and with no downtime.

In a cluster that includes mirror servers (which are in the passive-standby state), you can avoid cluster downtime by installing upgrades using a rolling upgrade. A rolling upgrade means that one by one, you shut down a node, install the upgrade on the node, and then restart the node. Your cluster will not be affected by a rolling upgrade because when an active server instance is being upgraded, the mirror server instance will take over the operations.

Note: In a cluster without any mirror servers, downtime is unavoidable during an upgrade installation. Additionally, if a cluster without mirrors is a non-persistent cluster, servers cannot be upgraded without losing all the data.

When you perform a rolling upgrade, a version of one client or server can continue running with a client or server of a different version, as long as they are both of the same major/minor version.

That is, the third digit of the version may be different, but the first two digits (major/minor) must be the same. For example, 4.2.1 servers can run with 4.2.0 clients. However, rolling upgrade support is not provided for a mix of different minor versions (e.g., 4.1.0 clients with 4.2.0 servers).

Note: When you perform a rolling upgrade, it is expected that you upgrade the entire cluster. A partially upgraded cluster is intended to be only a temporary state. You should continue to implement a rolling upgrade until the entire cluster is running on the same version.

Preparing for Maintenance Version Upgrades for 4.2

Before attempting to perform a rolling upgrade for 4.2, confirm the following:

- Ensure that all of the Terracotta servers and clients in the target cluster are running the same version of Terracotta.
- Ensure that the upgrade is intended for the version of Terracotta being run in your cluster.

- If persistence is enabled in your cluster, we recommend that you back up your data folder before upgrading, as described in "Database Backup Utility (backup-data) in the *BigMemory Max Administrator Guide*.

Note: A non-persistent cluster without mirror servers cannot be upgraded without losing all the data.

Installing Maintenance Version Upgrades for 4.2

Use the following procedure to upgrade BigMemory Max from 4.2.x to 4.2.y.

Note: This procedure assumes you are currently running a version 4.2 cluster in a high availability deployment configuration. In a cluster without any mirror servers, downtime is unavoidable during an upgrade installation. Additionally, if a cluster without mirrors is a non-persistent cluster, servers cannot be upgraded without losing all the data.

Step 1: Download the Kit

Download the BigMemory kit and verify that you have the correct license keys for the version you are installing.

Step 2: Upgrade All Terracotta Server Arrays (TSAs)

You must upgrade all TSAs in all regions before proceeding to Step 3, as follows.

1. For each mirror server instance in the cluster, do the following:
   a. Shut down the mirror server using the `stop-tc-server` script.
   b. Remove the old version of BigMemory Max from the installation directory of the mirror server.
   c. Optionally delete the objectdb data that is saved in the server's data directory under /dirty-objectdb-backup/dirty-objectdb-<timestamp>. Alternatively, you can restore from the old dirty objectdb's at a later time.
   d. Unzip the upgrade file in the mirror Terracotta server's installation (root) directory.

      On a UNIX/Linux machine, use the tar command. For example:

      ```
      tar zxf terracotta-4.2.1.tar.gz
      ```

      On a Microsoft Windows machine, double-click the zip file.
   e. Startup the mirror server using the `start-tc-server` script and wait for the synchronization to complete.
   f. Confirm that the mirror server has synced with the active server. You can use the Terracotta Monitoring Console to monitor server status.

g. Confirm that the upgrade was successful by running the version check shell script. This script is available in<kit>/server/bin/version.sh (or version.bat for Windows) and prints the current kit's version in standard output. For example, this command:

```
$ ./version.sh
```

Might produce this output:

```
Terracotta Enterprise 4.3.0-SNAPSHOT
```

2. For each active server instance in the cluster, do the following:

   a. Shut down the active server using the `stop-tc-server` script.

      The cluster fails over to the mirror server, which now becomes the active server.

   b. Install the upgrade on the shut-down server, the same way you installed it on the mirror server.

   c. Startup the server using the start-tc-server script. The formerly active server now becomes the mirror server.

   d. Optionally perform a failback to make the original active server (which is now a mirror) become the active server again. Alternatively, you can just leave it the way it is (i.e., the original active is now a mirror, and the original mirror is now an active.

      To perform a failback, stop the currently active server (i.e., the original mirror server). This will cause the original active server to become the active server again. Then restart the original mirror server, which will act as a mirror server again. Lastly, confirm that the original mirror server has synced with the newly active server. You can use the Terracotta Monitoring Console to monitor server status.

3. For each client in the cluster, do the following:

   a. Shut down the client. A Terracotta client will shut down when you shut down your application.

   b. Upgrade the software on the client.

   c. Startup the client.

To check the current versions of your cluster's nodes, the Terracotta Management Console (TMC) provides a Check Version button. For details, see the *Terracotta Management Console User's Guide*.

Step 3: Stop All Orchestrators

Perform this step if your cluster includes the BigMemory WAN Replication service. Stop all Orchestrators by running the `stop-wan.sh` or `stop-wan.bat` script from the kit. For example:

```
<kit>/server/bin/stop-wan.sh -f path/to/wan.xml
```

Note: If your cluster contains standby Orchestrators that are configured to reconnect the replica caches if they become disconnected from their master caches

(i.e., if the Orchestrator's `replicaDisconnectBehavior` parameter is set to `reconnectResync`), then we recommend stopping the standby Orchestrators first, and then stopping the master Orchestrators.

Step 4: Upgrade All Orchestrators to the Same Version

Perform this step if your cluster includes the BigMemory WAN Replication service. Install the new Orchestrator version from the latest BigMemory Max kit onto your Orchestrator machines. All Orchestrators must run the same version.

Step 5: Restart All Orchestrators

Perform this step if your cluster includes the BigMemory WAN Replication service. Restart all Orchestrators with the new version by running the `start-wan.sh` or `start-wan.bat` script from the kit. For example:

```
<kit>/server/bin/start-wan.sh -f path/to/wan.xml
```

The Master region will become immediately available, and then it will resynchronize all Replica regions.

Note: The resynchronization process of existing WAN-enabled caches will depend on the value of the replicaDisconnectBehavior configuration parameter. For more information, see "Orchestrator Configuration Parameters in the *BigMemory WAN Replication User Guide*.

Reverting TSAs to a Previous Version

To revert a TSA from 4.2.x to a previous 4.2 version, use the same procedure used for upgrading TSAs (as described in "Installing Maintenance Version Upgrades for 4.2" on page 13), but simply substitute the newer version with the older version.

4 Upgrading from 3.7 or 4.0

- Summary of Changes from 4.0 to 4.1 18
- Summary of Changes from 3.7 to 4.0 18
- Procedure for Migrating from 3.7 or 4.0 21

Summary of Changes from 4.0 to 4.1

Area of Change	Change from 4.0 to 4.1
TSA Operation	■ Changes to "Automatic Resource Management" on page for BigMemory Hybrid. ■ Changes to "Recovery from Near-Memory-Full Conditions" on page for BigMemory Hybrid.
TSA Disk Usage	"BigMemory Hybrid" on page .
TSA Configuration	■ "offheap" on page configuration change. ■ `dataStorage` and `hybrid` elements added. ■ `maxDataSize` element deprecated.
TSA Monitoring	■ "Data Lifecycle" on page support. ■ BigMemory Hybrid "statistics" on page .
WAN Replication	"WAN Replication Service" on page added.
Interoperability	"Cross-language Clients" on page added.
Search	■ Query using "BigMemory SQL" on page . ■ Support for "nulls" on page . ■ "Pagination" on page .
Tier Configuration	`MaxElementsInMemory` deprecated (use `MaxEntriesLocalHeap` instead).

Summary of Changes from 3.7 to 4.0

Area of Change	Change from 3.7 to 4.0
TSA Operation	■ Added "Automatic Resource Management" on page . ■ Added "Recovery from Near-Memory-Full Conditions" on page .

Area of Change	Change from 3.7 to 4.0
	■ Added "continuous uptime for 4.0 and higher" on page 19.
TSA Disk Usage	■ Data held in memory only. ■ "Fast restartability" on page .
TSA Configuration	■ "Simplified configuration of tc-config.xml for 4.0 and higher" on page 20. ■ No more DSO. ■ "offheap" on page required.
TSA Monitoring	"Terracotta Management Console (TMC)" on page replaces the Dev Console.
TSA Security	"Security changes for 4.0 and higher" on page 20.
Interoperability	BigMemory Hadoop Connector.
API	■ "Refresh ahead" on page . ■ EhcacheSizeofEngine.
Tier Configuration	`MaxEntriesInCache` replaces `MaxElementsOnDisk`.
Pinning	■ Key-level pinning removed. ■ `PinningConfiguration.Store#LOCALHEAP` removed.
Statistics	"Statistics changes for 4.0 and higher" on page 21.

Continuous Uptime for 4.0 and Higher

Improvements to provide continuous availability of data include:

- Support for multiple mirrors in mirror groups, with better utilization of extra mirrors.
- Flexibility in server startup sequencing. A mirror server can start up first during a cluster restart, and it will wait for the active server.
- Multi-stripe backup capability.
- Optimizations to bulk loading.

- Performance improvements for data access on rejoin.
- No more Oracle Berkeley DB, enabling in-memory data to be ready for use much more quickly after any planned or unplanned restart.

Simplified Configuration of tc-config.xml for 4.0 and Higher

The tc-config.xml has extensive changes from version 3.7:

- The `<mirror-group>` element can now wrap `<server>` configuration blocks, eliminating the need to repeat the setup of `<server>` information.
- `<dso-port>` is renamed to `<tsa-port>`, and `<l2-group-port>` is renamed to `<tsa-group-port>`.
- `<client-reconnect-window>`, `<garbage-collection>` (DGC), and `<restartable>` settings are now global for all servers.
- The `<data>` element specifies where to store data files used for the Fast Restartability feature.
- The `<clients>` section now has one setting, `<logs>`.
- `<security>` is a new configuration block for setting up authorization/authentication and SSL.
- `<dso>`, `<system>`, `<statistics>`, `<persistence>`, and `<ha>` (and, unless otherwise noted, any sub-elements) have been removed.

See the *BigMemory Max Administrator Guide* for a complete listing and definitions of configuration elements.

Security Changes for 4.0 and Higher

- HTTP authentication no longer supported.
- Security is enabled at the Terracotta Management Server (TMS) level, and so the server scripts only need the following when security is enabled:
  - https TMS URL
  - username and password
- Keychain entries are required for Terracotta clients, servers, and the Terracotta Management Console (TMC). By default, the keychain script that creates Terracotta keychain files uses an obfuscation scheme to protect passwords.
- Active Directory (AD) and Lightweight Directory Access Protocol (LDAP) support is available for Terracotta servers (see "Using LDAP or Active Directory for Authentication in the document *BigMemory Max Security Guide*).
- Custom SecretProvider is available for Terracotta clients (see document *BigMemory Max Security Guide*).

Statistics Changes for 4.0 and Higher

Beginning in version 4.0, statistics are no longer explicitly enabled or disabled via the ehcache.xml configuration file. The core counters for statistics are always enabled, but they have no impact on performance.

If you call for a statistic, via the TMC or programmatically, the various components of the statistic (latency, avg, min, max, history) will start being kept. As long as the statistic continues to be sampled within a defined amount of time, that statistic will be kept active.

Note that this is done on a per-statistic basis, so the overhead can vary depending on how many statistics are being sampled.

Procedure for Migrating from 3.7 or 4.0

Use this procedure to migrate from 3.7 or 4.0 to another major/minor pair version (e.g., from 3.7 to 4.0 or from 4.0 to 4.1). The procedure is the same for both scenarios. Every installation is unique, with many factors that affect migration. This procedure can be used as a checklist of relevant topics and basic migration tasks.

1. Download the BigMemory kit, and verify that you have the correct license keys for the version you are installing.
2. Prepare the new configuration files. Sample configuration files are located in the BigMemory kit in the config-samples/ directory. You can modify the sample files with specific information for your installation.
3. (Optional) If you have persistence enabled, you should make a backup of your data.
   - For version 3.7, see the *Database Backup Utility* section of the 3.7 BigMemory Max documentation.
   - For version 4.0, see the *Live Backup of Distributed In-memory Data* section of the 4.0 BigMemory Max documentation.
4. Shut down the server array. A safe shutdown procedure is as follows.
   a. Shut down the mirror servers using the `stop-tc-server` script.
   b. Shut down the clients. A Terracotta client will shut down when you shut down your application.
   c. Shut down the active servers using the `stop-tc-server` script.
5. Clear the servers by removing the old version of BigMemory, and by deleting the objectdb data that is saved in the server's data directory.
6. Install the new BigMemory kit. You must upgrade all Terracotta clients and servers in the cluster to the same version before restarting the cluster
7. Start the servers with their new configuration files.