



# **Red Hat JBoss Data Grid 6.4**

## **6.4.0 Release Notes**

Known and resolved issues for Red Hat JBoss Data Grid 6.4.0



# Red Hat JBoss Data Grid 6.4 6.4.0 Release Notes

---

Known and resolved issues for Red Hat JBoss Data Grid 6.4.0

Misha Husnain Ali  
Red Hat Engineering Content Services  
[mhusnain@redhat.com](mailto:mhusnain@redhat.com)

## Legal Notice

Copyright © 2015 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

The Red Hat JBoss Data Grid 6.4 Release Notes list and provide descriptions for a series of bugzilla bugs. The bugs highlights issues that are known problems and resolved issues for the release.

---

## Table of Contents

<b>CHAPTER 1. INTRODUCTION TO RED HAT JBOSS DATA GRID 6.4</b>	<b>3</b>
1.1. ABOUT RED HAT JBOSS DATA GRID	3
1.2. OVERVIEW	3
1.3. PACKAGING REVISIONS	3
1.4. UPGRADING FROM JBOSS DATA GRID 6.3.X TO 6.4	4
<b>CHAPTER 2. NEW FEATURES AND ENHANCEMENTS</b>	<b>5</b>
2.1. QUERYING FROM HOT ROD JAVA CLIENT (REMOTE QUERYING)	5
2.2. APACHE CAMEL COMPONENT FOR JBOSS FUSE	5
2.3. CLUSTERED LISTENERS	5
2.4. REMOTE EVENTS AND LISTENERS	5
2.5. HANDLING NETWORK PARTITIONS	5
2.6. CROSS-SITE STATE TRANSFER	6
2.7. SPRING INTEGRATION	6
<b>CHAPTER 3. SUPPORTED CONFIGURATIONS</b>	<b>7</b>
<b>CHAPTER 4. COMPONENT VERSIONS</b>	<b>8</b>
<b>CHAPTER 5. KNOWN AND RESOLVED ISSUES</b>	<b>9</b>
5.1. KNOWN ISSUES	9
5.2. RESOLVED ISSUES	12
<b>APPENDIX A. REVISION HISTORY</b>	<b>15</b>



# CHAPTER 1. INTRODUCTION TO RED HAT JBOSS DATA GRID

## 6.4

Welcome to the Red Hat JBoss Data Grid 6.4. As you become familiar with the newest version of JBoss Data Grid, these Release Notes provide you with information about new features, as well as known and resolved issues. Use this document in conjunction with the entire JBoss Data Grid documentation suite, available at the Red Hat Customer Service Portal's [JBoss Data Grid documentation page](#).

[Report a bug](#)

### 1.1. ABOUT RED HAT JBOSS DATA GRID

Red Hat's JBoss Data Grid is an open source, distributed, in-memory key/value data store built from the Infinispan open source software project. Whether deployed in client/server mode or embedded in a Java Virtual Machine, it is built to be elastic, high performance, highly available and to scale linearly.

JBoss Data Grid is accessible for both Java and Non-Java clients. Using JBoss Data Grid, data is distributed and replicated across a manageable cluster of nodes, optionally written to disk and easily accessible using the REST, Memcached and Hot Rod protocols, or directly in process through a traditional Java Map API.

[Report a bug](#)

### 1.2. OVERVIEW

This document contains information about the new features and known issues of Red Hat JBoss Data Grid version 6.4. Customers are requested to read this documentation prior to installing this version.

[Report a bug](#)

### 1.3. PACKAGING REVISIONS

To simplify embedding Red Hat JBoss Data Grid directly in your application, the Library distribution of JBoss Data Grid 6.4 contains fewer, consolidated jars. Specifically:

- **infinispan-embedded-`{VERSION}`.jar** contains the core Infinispan components, including caches stores.
- **infinispan-embedded-query-`{VERSION}`.jar** contains the Querying components
- **infinispan-cli-`{VERSION}`.jar** contains the command line interface
- Transactions and Spring integration modules are packaged separately

The Java Hot Rod client resides in a single consolidated jar file **infinispan-remote-`{VERSION}`.jar** under **jboss-datagrid-`{VERSION}`-remote-java-client.zip**.



#### NOTE

The packaging of Red Hat JBoss EAP modules is unchanged in JBoss Data Grid 6.4. Additionally, the Maven repository retains the unconsolidated jar files, similar to JBoss Data Grid 6.3, for backward compatibility.

[Report a bug](#)

## 1.4. UPGRADING FROM JBOSS DATA GRID 6.3.X TO 6.4

The following information is useful for users upgrading from Red Hat JBoss Data Grid 6.3.x to 6.4.

### Hot Rod Protocol

The Hot Rod **size()** method in JBoss Data Grid 6.3 is used to obtain the size by looking at the **numberOfEntries** statistic on the server. This required statistics to be enabled and the **ADMIN** permission if security was in use (instead of the more appropriate **BULK\_READ**). It also only returned the number of entries in the node that responded to the request. This included potentially expired entries and did not take stores/loaders into account. In JBoss Data Grid 6.4, the **size** method's behavior can be modified to include the size of the entire cluster (not just the local node). If the **infinispan.accurate.bulk.ops system** property is set to **true** then the **size** method will use the new behavior by utilizing the entry iterator to count all entries.

### Changes to the size() Method

The **size()** method no longer returns the value of the **currentNumberOfEntries** statistic. Instead it will retrieve all the keys using **keySet()** and will then return the number of entries in this set.

The previous implementation was less costly but it did not return accurate information when the cache contained expired entries.

**keySet()** is a costly operation and because of this **size()** is now a costly operation as well. Distributed caches it will retrieve all the keys from all the nodes in the cluster.

[Report a bug](#)



## CHAPTER 2. NEW FEATURES AND ENHANCEMENTS

### 2.1. QUERYING FROM HOT ROD JAVA CLIENT (REMOTE QUERYING)

Red Hat JBoss Data Grid 6.2 introduced a technology preview of a new domain-specific language (DSL) to search for data in the grid using values instead of keys, in Remote Client-Server mode from a Hot Rod Java client. This Remote Querying feature has been enhanced further and is fully supported in JBoss Data Grid 6.4.



#### NOTE

The Infinispan Query DSL, which is used for querying in Remote Client-Server mode, can also be used in Library mode.

[Report a bug](#)

### 2.2. APACHE CAMEL COMPONENT FOR JBOSS FUSE

Red Hat JBoss Data Grid 6.4 adds support for a new camel-jbosssdatagrid component, which is tested with Red Hat JBoss Fuse 6.1 and Red Hat JBoss Fuse Service Works 6.0 (using Apache Camel 2.12). This enables use of JBoss Data Grid for distributed caching, in both Library and Remote Client-Server modes, on Camel routes with JBoss Fuse or JBoss Fuse Service Works.

[Report a bug](#)

### 2.3. CLUSTERED LISTENERS

In previous Red Hat JBoss Data Grid versions, a write event (creation, update, or deletion of an entry) in a distributed cache was visible only to the listeners that were registered on the same node on which the event occurred. That is, the listeners were local. In JBoss Data Grid 6.4, these write events are notified to a listener registered on any node of the distributed cache.

Clustered listeners are a Library mode feature.

[Report a bug](#)

### 2.4. REMOTE EVENTS AND LISTENERS

The Red Hat JBoss Data Grid 6.4 features a technology preview of Remote Events and Listeners, based on the Hot Rod protocol. Hot Rod Java clients can set listeners on the JBoss Data Grid Server and receive relevant notifications when entries are created, modified or deleted in the remote cache. Server-side filtering of events is also available. This feature uses clustered listeners on the server side.

The remote events/listeners feature is applicable to Remote Client-Server mode.

[Report a bug](#)

### 2.5. HANDLING NETWORK PARTITIONS

Red Hat JBoss Data Grid (JDG) 6.4 introduces the ability to handle network partitions (split brains), as a configurable option. When Partition Handling is enabled and JDG suspects that one or more nodes of a clustered cache are no longer accessible, each partition does not start a rebalance immediately, but first

checks whether it should enter Degraded mode instead. When a partition has entered Degraded mode, only those entries for which this partition has all owners (copies) are available for reads and writes in this partition. Attempts to read or write entries that are not fully owned by this partition will result in an Availability Exception. When the partitions merge, JDG also determines whether the combined cache should be made available or remain Degraded.

This feature is available in both Library and Remote Client-Server modes.

[Report a bug](#)

## 2.6. CROSS-SITE STATE TRANSFER

Red Hat JBoss Data Grid supports cross datacenter replication since version 6.1. However, in previous versions, when one of the replica sites goes down and subsequently becomes available, the newly-available site does not synchronize with the live site(s). JBoss Data Grid 6.4 adds the ability to synchronize the newly-available site with its replica sites(s).

This feature is available in both Library and Remote Client-Server modes.

[Report a bug](#)

## 2.7. SPRING INTEGRATION

Red Hat JBoss Data Grid 6.4 includes infinispan-spring modules for integration with Spring Framework versions 3.2 and 4.1. This allows you to seamlessly use JBoss Data Grid for caching in your Spring application.

This feature is available in both Library and Remote Client-Server modes.



### NOTE

Red Hat will test and support the infinispan-spring modules that are included in the JBoss Data Grid distribution with the specified versions of Spring Framework. However, Red Hat will not fix bugs or create patches for issues discovered in the Spring project.

[Report a bug](#)

## CHAPTER 3. SUPPORTED CONFIGURATIONS

For supported configurations, see the Red Hat JBoss Data Grid Supported Configurations reference on the Customer Portal at <https://access.redhat.com/site/articles/115883>.

[Report a bug](#)

## CHAPTER 4. COMPONENT VERSIONS

For component versions, see the Red Hat JBoss Data Grid Supported Configurations reference on the Customer Portal at <https://access.redhat.com/articles/488833>.

[Report a bug](#)

## CHAPTER 5. KNOWN AND RESOLVED ISSUES

### 5.1. KNOWN ISSUES

#### BZ-1178965 - Fail fast when using two phase commit with ASYNC backup strategy

In Red Hat JBoss Data Grid, using the two phase commit with the ASYNC backup strategy results in one phase commit unexpectedly being used instead.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### BZ-1163665 - Node can temporarily read removed data when another node joins the cluster, leaves or crashes

In Red Hat JBoss Data Grid, the distribution of entries in the cluster changes when a node joins, leaves or crashes during a brain split. During this brief period, a read on the previous node owner can return stale data. When the rebalance process is completed, further reads return up-to-date data.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### BZ-1175272 - CDI fails when both remote and embedded uber-jar are present

In Red Hat JBoss Data Grid, when both *infinispan-remote* and *infinispan-embedded* dependencies are on the classpath, the Infinispan CDI extension does not work as expected. This is due to bundling the Infinispan CDI extension in both jar files. As a result, CDI fails with ambiguous dependencies exception.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### BZ-1012036 - RELAY2 logs error when site unreachable

In Red Hat JBoss Data Grid, when a site is unreachable, JGroups's **RELAY2** logs an error for each dropped message. Infinispan has configurable fail policies (**ignore/warn/abort**), but the log is filled with errors despite the **ignore** policy.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### BZ-1024373 - Default optimistic locking configuration leads to inconsistency

In Red Hat JBoss Data Grid, transactional caches are configured with optimistic locking by default. Concurrent **replace()** calls can return true under contention and transactions might unexpectedly commit.

Two concurrent commands, **replace(key, A, B)** and **replace(key, A, C)** may both overwrite the entry. The command which is finalized later wins, overwriting an unexpected value with new value.

This is a known issue in JBoss Data Grid 6.4. As a workaround, enable write skew check and the **REPEATABLE\_READ** isolation level. This results in concurrent replace operations working as expected.

#### BZ-1107613 - SASL GSSAPI auth doesn't use principal configured login\_module

In Red Hat JBoss Data Grid, the server principal is always constructed as **jgroups/server\_name** and is not loaded from the Kerberos login module. Using a different principal results in an authentication failure.

This is a known issue in JBoss Data Grid 6.4 and the workaround for this issue is to use **jgroups/server\_name** as the server principal.

#### **BZ-1114080 - HR client SASL MD5 against LDAP fails**

In Red Hat JBoss Data Grid, the server does not support pass-through MD5 authentication against LDAP. As a result, the Hot Rod client is unable to authenticate to the JBoss Data Grid server via MD5 as the authentication is backed by the LDAP server.

This is a known issue in JBoss Data Grid 6.4 and a workaround is to use the PLAIN authentication over end-to-end SSL encryption.

#### **BZ-881791 - Special characters in file path to JDG server are causing problems**

In Red Hat JBoss Data Grid, when special characters are used in the directory path, the JBoss Data Grid server either fails to start or a configuration file used for logging cannot be loaded properly. Special characters that cause problems include spaces, # (hash sign), ! (exclamation mark), % (percentage sign), and \$ (dollar sign).

This is a known issue in JBoss Data Grid 6.4. A workaround for this issue is to avoid using special characters in the directory path.

#### **BZ-1092403 - JPA cachestore fails to guess dialect for Oracle12c and PostgresPlus 9**

In Red Hat JBoss Data Grid, JPA Cache Store does not work with Oracle12c and Postgres Plus 9 as Hibernate, an internal dependency of JPA Cache Store, is not able to determine which dialect to use for communication with the database.

This is a known issue in JBoss Data Grid 6.4. As a workaround for around this issue, specify the Hibernate dialect directly by adding the following element in the **persistence.xml** file:

```
<property name="hibernate.dialect" value="${hibernate.dialect}" />
```

Set `${hibernate.dialect}` to **org.hibernate.dialect.Oracle10gDialect** or **org.hibernate.dialect.PostgresPlusDialect** for Oracle12c or Postgres Plus 9 respectively.

#### **BZ-1101512 - CLI UPGRADE command fails when testing data stored via CLI with REST encoding**

In Red Hat JBoss Data Grid, the CLI **upgrade** command fails to migrate data from the old cluster to the new cluster if the data being migrated was stored in the old cluster via CLI with REST encoding (e.g by issuing a command such as **put --codec=rest key1 val1**). This issue does not occur if data is stored via REST clients directly.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### **BZ-1158559 - C++ HotRod Client, RemoteCache.clear() will throw out exception when data is more than 1M**

In Red Hat JBoss Data Grid, when a cache contains a large number of entries, the `clear()` operation can take an unexpectedly long time and possibly result in communication timeouts. In this case, the exception is reported to the Hot Rod client.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### BZ-# - Title

In Red Hat JBoss Data Grid, for a cache store where *sharing* is set to **false**, the *fetchInMemory* and *fetchPersistence* parameters must also be set to **true** or different nodes may contain different copies of the same data.

For a cache store where the *shared* parameter is set to **true**, set *fetchPersistence* must be set to **false** because the persistence is shared and enabling it results in unnecessary state transfers.

However, the *fetchInMemory* parameter can be set to either **true** or **false**. Setting this to **true** loads the in-memory state via the network and resulted in a faster start up. Setting the value to **false** loads the data from persistence without transferring the data remotely from other nodes.

This is a known issue in JBoss Data Grid 6.4 and the only workaround for this issue is to follow the stated guidelines above and to ensure that each node in the cluster uses the same configuration to prevent unexpected results based on the start up order.

#### BZ-881080 - Silence SuspectExceptions

In Red Hat JBoss Data Grid, **SuspectExceptions** are routinely raised when nodes shut down because they are unresponsive as they shut down. As a result, a **SuspectException** error is added to the logs. The **SuspectExceptions** do not affect data integrity.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### BZ-807674 - JDBC Cache Stores using a JTA Data Source do not participate in cache transactions

In Red Hat JBoss Data Grid's library mode, JDBC cache stores can be configured to use a JTA-aware datasource. However, operations performed on a cache backed by such a store during a JTA transaction will be persisted to the store outside of the transaction's scope. This issue is not applicable to JBoss Data Grid's Remote Client-Server mode because all cache operations are non-transactional.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

#### BZ-1088073 - Move rebalancing settings in JON from cache level to global cluster level

In Red Hat JBoss Data Grid, it is possible to change rebalancing settings using the JBoss Operations Network UI when navigating to the JBoss Data Grid Server/Cache Container/Cache/Configuration (current)/Distributed Cache Attributes/Rebalancing.

This operation is currently misrepresented as a cache-level operation when the changed rebalancing settings automatically apply for all the parent cache manager's caches and for all nodes in a particular cluster.

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

**BZ-1158839 - Clustered cache with FileStore (shared=false) is inconsistent after restarting one node if entries are deleted during restart**

In Red Hat JBoss Data Grid, when a node restarts, it does not automatically purge entries from its local cache store. As a result, the Administrator starting the node must change the node configuration manually to set the cache store to be purged when the node is starting. If the configuration is not changed, the cache may be inconsistent (removed entries can appear to be present).

This is a known issue in JBoss Data Grid 6.4 and no workaround is currently available for this issue.

[Report a bug](#)

## 5.2. RESOLVED ISSUES

**BZ-1156397 - File System Deployments are not enabled in standalone/clustered.xml by default**

Previously in Red Hat JBoss Data Grid, the server did not contain the deployment scanner which is responsible for handling deployments. As a result, it was not possible to deploy custom user filters, converters and marshallers for enhanced remote listeners by copying jar files with these pieces into the **standalone/deployments** folder in the server.

This issue is now resolved in JBoss Data Grid 6.4 and the server now contains the deployment scanner and works as expected.

**BZ-115555 - JGroups configuration files might be shadowed if the JDG infinispn-core library is provided as EAP module**

Previously in Red Hat JBoss Data Grid, the default configuration files for JGroups were named **jgroups-udp.xml** and **jgroups-tcp.xml**. When a user provided their own configuration file with the same name, it was unexpectedly shadowed with the default one.

This issue is now resolved in JBoss Data Grid 6.4. The default configuration files are now moved to the **default-configs** directory, renamed to **default-jgroups-udp.xml** and **default-jgroups-tcp.xml**. Additionally, if a user chooses the same name for their customized configuration files, a warning is logged.

**BZ-1156071 - Missing partition handling configuration in schema**

Previously in Red Hat JBoss Data Grid, the configuration schema for embedded (library) mode did not contain configuration elements for partition handling.

This issue is now resolved in JBoss Data Grid 6.4 and the configurations for partition handling are included in the configuration schema.

**BZ-1172021 - Cross site state transfer - NPE on consumer site when backup cache is not present**

Previously in Red Hat JBoss Data Grid, when the backup cache was not defined in the consumer site (with either the same name as the main cache or specified via the `<backup-for>` option), the customer site would create a new local cache without the RPC manager. This resulted in a `NullPointerException` being thrown in the consumer site with the following error message also displayed in the producer site:

```
Unable to pushState to 'XYZ'. org.infinispan.commons.CacheException:
Problems invoking command
```



This issue is now resolved in JBoss Data Grid 6.4 so that if the cache does not exist, it is created with an RPC manager as expected.

#### **BZ-1161479 - HR size operation requires ADMIN permission**

Previously in Red Hat JBoss Data Grid, the Map/reduce task missed security actions. As a result, users could not use the Hot Rod size() operation via the map/reduce approach unless they had ADMIN permissions.

This issue is now resolved in JBoss Data Grid 6.4 by adding the required map/reduce security actions. As a result, users with EXEC permissions can now execute map/reduce operations as expected.

#### **BZ-1161590 - Remove features file for JPACacheStore from JDG maven repository as it is unsupported**

Previously in Red Hat JBoss Data Grid, the JPACacheStore was not supported in OSGi/Karaf but the JBoss Data Grid Maven distribution unexpectedly contained a features file for JPACacheStore for deployment in Karaf.

This issue is now resolved in JBoss Data Grid 6.4 and the features file is removed.

#### **BZ-1168329 - Discovery protocols have trouble finding coord in large clusters**

Previously in Red Hat JBoss Data Grid, for clusters with more than 16 nodes, node discovery would prevent a new node from joining an existing cluster.

This issue is now resolved in JGroups 3.6.1, which is included in JBoss Data Grid 6.4 so that node discovery allows new nodes to join existing clusters as expected.

#### **BZ-1172038 - JGroups subsystem doesn't support Vault**

Previously in Red Hat JBoss Data Grid, the JGroups subsystem integration with Vault was missing. As a result, sensitive information contained in the JGroups part of configuration files, such as passwords, could not be stored securely using Vault encryption.

This issue is now resolved in JBoss Data Grid 6.4 by adding the JGroups subsystem with Vault. As a result, all passwords, including JGroups passwords, can now be stored securely using Vault.

#### **BZ-1161529 - Possible false positive suspect of FD\_HOST when the number of hosts is large**

Previously in Red Hat JBoss Data Grid, the JGroups FD\_HOST protocol received an unexpectedly high number of false positive suspects within large clusters.

This issue is now resolved in JBoss Data Grid 6.4 by improving the algorithm used to check the cluster for dead member nodes.

#### **BZ-1163573 - Coordinator promotion is keep failing successively, with massive GCs in a long loop**

Previously in Red Hat JBoss Data Grid, for a large cluster with many caches, the CacheStatusResponse map on the new coordinator consumed an unexpectedly large amount of memory. As a result, the JVM garbage collection has extra load when selecting a new coordinator for the cluster.

This issue is now resolved in JBoss Data Grid 6.4 by the allowing the caching of various instances in the serializaton context. Reuse results in consistent hashes and repeated cache join information.

[Report a bug](#)

## APPENDIX A. REVISION HISTORY

**Revision 6.4.0-2**

First draft concluded.

**Mon Jan 19 2015****Misha Husnain Ali****Revision 6.4.0-1**

Created initial version.

**Wed Jul 23 2014****Gemma Sheldon**