



Red Hat JBoss Data Grid 7.2

7.2 リリースノート

Red Hat JBoss Data Grid 7.2 の既知の問題および解決済みの問題

Red Hat JBoss Data Grid 7.2 7.2 リリースノート

Red Hat JBoss Data Grid 7.2 の既知の問題および解決済みの問題

法律上の通知

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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.

概要

Red Hat JBoss Data Grid 7.2.0 リリースノートでは、Bugzilla のバグをリストし、説明します。これらのバグは、本リリースの既知の問題と解決済みの問題になります。

目次

第1章 JBOSS DATA GRID 7.2 の紹介	3
1.1. RED HAT JBOSS DATA GRID 7.2 の紹介	3
1.2. HAT JBOSS DATA GRID について	3
1.3. 概要	3
第2章 新機能および改良された機能	4
2.1. データコンテナの改良	4
2.2. パーティション処理の改良	4
2.3. HOT ROD C# クライアントの SASL 認証	4
2.4. HOT ROD C++ クライアントの SASL 認証	4
2.5. PROTOSTREAM 4.2.0	4
2.6. HIBERNATE SEARCH 5.7.3.FINAL および APACHE LUCENE 5.5.5	4
2.7. EAP の統合	4
2.8. SPARK コネクタの改良	5
2.9. OPENSIFT イメージの同時リリース	5
2.10. クラスター化カウンター	5
2.11. クラスター化ロック	5
2.12. マルチマップキャッシュ	5
2.13. クラスター全体の動的キャッシュ作成	5
2.14. ブロードキャストクエリー	5
2.15. 設定可能なデータ形式	6
第3章 サポートされる構成	7
3.1. サポートされる構成	7
第4章 コンポーネントのバージョン	8
4.1. コンポーネントのバージョン	8
第5章 既知の問題	9
5.1. 既知の問題	9
第6章 解決済みの問題	11
6.1. 解決済みの問題	11

第1章 JBOSS DATA GRID 7.2 の紹介

1.1. RED HAT JBOSS DATA GRID 7.2 の紹介

Red Hat JBoss Data Grid 7.2 へようこそ。最新バージョンの JBoss Data Grid を理解していただくため、本リリースノートでは新しい機能、既知の問題、および解決された問題について説明します。本書は、Red Hat カスタマーサービスポータル[の JBoss Data Grid のドキュメントページ](#)で利用できる JBoss Data Grid の全ドキュメントとともに使用してください。

1.2. HAT JBOSS DATA GRID について

Red Hat JBoss Data Grid は、Infinispan オープンソースソフトウェアプロジェクトから構築されたオープンソースでインメモリー型の分散キーバリューストアです。クライアントサーバーモードでのデプロイや、Java 仮想マシンへの埋め込みに関係なく、弾力性、高パフォーマンス、高可用性、および線形スケールを実現します。

JBoss Data Grid は Java のクライアントと Java でないクライアントの両方からアクセスできます。JBoss Data Grid を使用すると、データがノードの管理可能なクラスター全体で分散およびレプリケートされ、任意でディスクに書き込まれます。また、REST、Memcached、および Hot Rod プロトコルを使用すると簡単にデータへアクセスできますが、従来の Java Map API を用いてプロセスで直接アクセスすることも可能です。

1.3. 概要

本書には Red Hat JBoss Data Grid 7.2 に関する重要な情報が含まれています。本書を読んでから本バージョンの Red Hat JBoss Data Grid をインストールするようにしてください。

第2章 新機能および改良された機能

2.1. データコンテナの改良

JBoss Data Grid 7.2 のデータコンテナには、**TinyLFU** と呼ばれる新しいエビクションストラテジーや **OffHeap** と呼ばれる新しいストレージタイプなどを含む、多くの変更が追加されました。データコンテナは、新しい **<memory>** 要素を使用して設定され、この要素は非推奨となった従来の **<eviction>** および **<storeAsBinary>** 要素を統合します。現在のデータコンテナに関する詳細は、『Administration and Configuration Guide』の「[Set Up JVM Memory Management](#)」を参照してください。

2.2. パーティション処理の改良

JBoss Data Grid 7.2 のパーティション処理は大幅に更新され、異なるパーティション処理ストラテジーを設定する機能、新しい競合マネージャー、競合を解決する方法のカスタマイズに使用できる複数のマージポリシーなどが含まれます。JBoss Data Grid では、これまでパーティション処理を有効または無効にすることのみが可能でした。この更新により、パーティション処理の動作を細かく調整できるようになりました。更新の詳細については、『Administration and Configuration Guide』の「[Network Partition Recovery](#)」を参照してください。

2.3. HOT ROD C# クライアントの SASL 認証

JBoss Data Grid 7.2 では、C# Hot Rod クライアントとサーバー間の SASL 認証をサポートするようになりました。JDG はこれまで、ネイティブ Java Hot Rod クライアントでの SASL 認証のみをサポートしていました。C# クライアントは、PLAIN、DIGEST-MD5、および EXTERNAL セキュリティーメカニズムをサポートします。詳細は『Developer Guide』の「[Securing Interfaces](#)」を参照してください。

2.4. HOT ROD C++ クライアントの SASL 認証

JBoss Data Grid 7.2 では、C++ Hot Rod クライアントとサーバー間の SASL 認証をサポートするようになりました。JDG はこれまで、ネイティブ Java Hot Rod クライアントでの SASL 認証のみをサポートしていました。C++ クライアントは、PLAIN、DIGEST-MD5、EXTERNAL、および GSSAPI (GSSAPI は Windows ではサポートされません) セキュリティーメカニズムをサポートします。詳細は『Developer Guide』の「[Securing Interfaces](#)」を参照してください。

2.5. PROTOSTREAM 4.2.0

JBoss Data Grid 7.2 には新たに [ProtoStream](#) 4.2.0 が導入されました。ProtoStream は、Google の [Protobuf](#) データ形式をベースにしたシリアライズライブラリーです。

2.6. HIBERNATE SEARCH 5.7.3.FINAL および APACHE LUCENE 5.5.5

JBoss Data Grid 7.2 には新たに Hibernate 5.7.3.Final および Apache Lucene 5.5.5 が導入されました。これらは、分散グリッドに格納されたデータセット全体でクエリーを実行するため、JDG のクエリーモジュールによって使用されます。また、JDG は Hibernate Search のディレクトリープロバイダーの役割も果たします。

2.7. EAP の統合

JBoss Data Grid 7.2 では、新しいサブシステムを利用してキャッシュを **standalone.xml** に設定できるようになりました。この方法を使用すると、キャッシュのライフサイクルはアプリケーションに拘束

されません。EAP の内部 Infinispan モジュールをインジェクトする方法と同様に、キャッシュを `@Resource` アノテーションでインジェクトすることもでき、これはサポートされます。詳細は、『Developer Guide』の「[Integration with EAP](#)」を参照してください。

2.8. SPARK コネクターの改良

JBoss Data Grid 7.2 には、Spark 1.6.x および 2.x をサポートし、それにより Scala 2.10.x および 2.11.x をサポートする、新バージョンの Apache Spark コネクターが導入されました。このコネクターは、改善された設定、クエリー String による簡単な絞り込み、改善されたプロトコルバッファのサポート、Dataset API の新規サポートなど、多くの改良が加えられています。詳細は『Developer Guide』の「[Integration with Apache Spark](#)」を参照してください。

2.9. OPENSIFT イメージの同時リリース

JBoss Data Grid 7.2 は、コンテナ化されていないプラットフォームで使用される標準の zip アーカイブがリリースされると同時にバージョン 7.2 の OpenShift イメージをリリースします。JBoss Data Grid 7.2 の OpenShift イメージは、[Red Hat Container Catalog](#) からダウンロードできます。

2.10. クラスター化カウンター

JBoss Data Grid 7.2 には、クラスター化カウンターのサポートが追加されました。クラスター化カウンターは、Red Hat JBoss Data Grid クラスターのノードすべてで分散および共有され、オブジェクトの数を記録できます。クラスター化カウンターは、`counter` API より Red Hat JBoss Data Grid のライブラリーモードおよびリモートクライアントサーバーモードで利用できます。詳細は、『開発ガイド』の「[Clustered Counters](#)」を参照してください。

2.11. クラスター化ロック

JBoss Data Grid 7.2 には、クラスター化ロックのサポートが追加されました。クラスター化ロックは、Red Hat JBoss Data Grid クラスターのノードすべてで分散および共有されるデータ構造です。クラスター化ロックによって、クラスター内のノード間で同期化されるコードを実行できます。クラスター化ロックは、`lock` API より Red Hat JBoss Data Grid のライブラリーモードでのみ利用できます。詳細は、『開発ガイド』の「[Clustered Locks](#)」を参照してください。

2.12. マルチマップキャッシュ

JBoss Data Grid 7.2 には、キーを値にマップするキャッシュで、各キーに複数の値を含めることができる `MultimapCache` が新たに導入されました。これは、現在ライブラリーモードでのみ機能します。詳細は『Developer Guide』の「[The Multimap Cache](#)」を参照してください。

2.13. クラスター全体の動的キャッシュ作成

JBoss Data Grid 7.2 は、動的なキャッシュの作成をサポートするようになりました。ライブラリーモードまたは Hot Rod クライアントから動的にキャッシュを作成するには、`createCache()` メソッドを使用します。たとえば、ライブラリーモードにてクラスターに参加する新しいノード上でキャッシュを動的に作成するには、`manager.administration().createCache("newCacheName", "newTemplate")` を使用します。詳細は、『Developer Guide』の「[Library Mode](#)」または「[Remote Client-Server Mode](#)」を参照してください。

2.14. ブロードキャストクエリー

ブロードキャストクエリー機能は、各ノードが書き込み中に独自のデータをインデックス化し、クエ

リー時に各ノードにクエリーを送信 (ブロードキャスト) できるようにします。各ノードの結果は、呼び出し元に返される前に組み合わせられます。これは、転送されるデータの量がクエリー自体で結果であるため、大きなインデックスを持つ DIST キャッシュに適しています。詳細は、『Developer Guide』の「[Broadcast Queries](#)」を参照してください。

2.15. 設定可能なデータ形式

JBoss Data Grid では、REST API を介してキャッシュしたデータとの相互運用が可能です。7.2 では、データ形式を定義するメディアタイプを設定できるようになりました。一部の例外を除き、デフォルトのメディアタイプは **application/octet-stream** になります。

本リリースの JBoss Data Grid では、JSON ドキュメントを格納、読み出し、および検索できる **application/json** のサポートも追加されました。さらに、JBoss Data Grid は JSON 形式の Ickle クエリーを使用する、REST インターフェースを介したデータのクエリーをテクノロジープレビューとして提供します。

サポートされるメディアタイプに関する詳細は、『Developer Guide』の「[Using the REST Interface](#)」を参照してください。

第3章 サポートされる構成

3.1. サポートされる構成

JDG 7.2 のサポートされるハードウェアおよびソフトウェア構成は、カスタマーポータルの <https://access.redhat.com/ja/articles/3073961> を参照してください。

第4章 コンポーネントのバージョン

4.1. コンポーネントのバージョン

Red Hat JBoss Data Grid 7.2 で使用されるコンポーネントバージョンの完全なリストは、カスタマーポータル[の <https://access.redhat.com/ja/articles/757083>](https://access.redhat.com/ja/articles/757083) を参照してください。

第5章 既知の問題

5.1. 既知の問題

Red Hat JBoss Data Grid 7.2 での既知の問題および回避方法を確認してください。

JBoss Data Grid が起動せず、永続的な場所が一意でないと `NullPointerException` が発生する

問題: [JDG-1504](#)

説明: `persistent-location` の値が各設定で一意でないと、同じディレクトリーから複数の JBoss Data Grid インスタンスを起動できません。

回避方法: 以下の 1 つを行い、`persistent-location` の値が JBoss Data Grid 設定で一意になるようにします。

- 各インスタンスに一意の `path` 属性を設定します。

```
<persistent-location path="mydir${uniquesuffix}" />
```

- 共通のパスを使用している場合は、各インスタンスに一意の `relative-to` 属性を設定します。

```
<paths>
  <path name="mypath" path="/home/user/jboss-datagrid-7.2.0-
server/standalone/mypath" />
</paths>
```

```
<persistent-location relative-to="mypath" path="mydir" />
```

- 共通のパスを使用している場合は、`-Djboss.server.data.dir` システムプロパティーを指定して、各インスタンスに一意で相対的な場所を設定します。

```
./standalone.sh -Djboss.server.data.dir=/home/user/jboss-datagrid-
7.2.0-server/standalone/mypath
```

認証が有効な場合、`SKIP_CACHE_LOAD` フラグが無効になる

問題: [JDG-1424](#)

説明: リモートクライアントサーバーモードで、`SKIP_CACHE_LOAD` フラグをキャッシュストア設定に指定し、Hot Rod クライアントで認証を有効にすると、エビクトされたエントリーを含むすべてのエントリーがキャッシュから取得されます。

回避方法: この問題を回避する方法はありません。

クラスター間のローリングアップグレード中にデータ損失が断続的に発生する

問題: [JDG-991](#)

説明: JBoss Data Grid のローリングアップグレードを実行する際、ソースクラスターのノードがオフラインになった後に移行したデータがすべてターゲットクラスターから削除される可能性があります。

回避方法: この問題を回避する方法はありません。

コンテナインストールでは HotRod C++ クライアントの JBoss Data Grid API ドキュメントがない

問題 JDG-1879

説明: JBoss Data Grid HotRod C++ クライアントを Red Hat Container Catalog からのコンテナとしてインストールした場合、`/usr/share/doc/infinispan-hotrod-cpp/api/` ディレクトリーに API ドキュメントがありません。

この問題は、yum 設定でコンテナイメージに **nodocs** トランザクションフラグが設定されているため発生します。

回避方法: yum 設定の **nodocs** フラグを消去し、コンテナイメージを次のように再インストールします。

```
yum --setopt=tsflags='' reinstall jdgc-cpp-client
```

リロードが必要な状態で JBoss Data Grid の管理コンソール上のクラスターアクションが無効になる**問題: JDG-1843**

説明: 設定の変更後にクラスターの再起動を選択すると、JBoss Data Grid クラスターで使用できるアクションが管理コンソールで使用できません。この場合、クラスターは **Reload-Required** 状態になります。

Reload および **Stop** アクションはクラスターの各ノードに使用できます。

回避方法: クラスターのノードを 1 つ以上リロードし、クラスターレベルでアクションをリストアします。

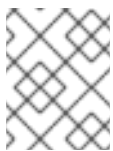
JBoss Data Grid の管理コンソールからエビクションストラテジーを変更するとエラーが発生する**問題: JDG-1804**

説明: JBoss Data Grid をドメインモードで実行している場合、管理コンソールで設定のエビクションストラテジーを変更しても再起動して変更を適用しないと、エラーが発生します。

回避方法: エビクションストラテジーの変更後にサーバーを再起動します。

JBoss Data Grid 7.0 およびそれ以前のバージョンに保存されたキャッシュからデータを読み取ると `NullPointerException` が発生する**問題: JDG-968**

説明: JBoss Data Grid 7.0 およびそれ以前のバージョンでデータをキャッシュに保存し、JBoss Data Grid 7.1 およびそれ以降のバージョンでそのデータの読み取りを試みると、エラーが発生し、データを読み取ることができません。

**注記**

この問題は、JBoss Data Grid 7.1 を 7.2 にアップグレードするときには発生しません。

回避方法: この問題を回避する方法はありません。

第6章 解決済みの問題

6.1. 解決済みの問題

JBoss Data Grid 7.2 で解決した問題は次のとおりです。

JDG-1768 - JBoss Data Grid のエクスパレーションの値が REST API からのリクエストに適用されない

リクエストにヘッダーが含まれていないと、エクスパレーションの値が JBoss Data Grid の REST インターフェースへのリクエストに適用されませんでした。

JDG-1143 - シンプルな Cache インターフェースに対してマージ機能が想定どおりに動作しない

シンプルな Cache インターフェースで **Merge** メソッドが想定どおりに動作しませんでした。

JDG-1127 - 高負荷における空間クエリーリクエストでエラーが発生する

異なるクエリーパラメーターを持つ複数のクエリーが同時に実行されると、**org.hibernate.search.exception.SearchException** エラーが発生しました。

JDG-1039 - 接続障害後に Hot Rod クライアントがトポロジーアドレスを解決できない

JBoss Data Grid ノードが異なる IP アドレスを割り当てられて再起動する動的な環境では、Hot Rod クライアントが再接続に失敗し、ノードの IP アドレスを解決できませんでした。

JDG-1026 - CompatibilityProtoStreamMarshaller クラスを設定できない

互換性マーシャラーを **CompatibilityProtoStreamMarshaller** に設定できませんでした。

JDG-1122 - Memcached サーバーを介した書き込みで UnknownOperationException が発生する

場合によっては、JBoss Data Grid をアプリケーションで memcache サービスとして使用すると、**UnknownOperationException** 例外が発生し、異なるサイズのキーバリューペアを書き込むときにデータが保存されないことがありました。

JDG-954 - Ickle の数字を範囲とするフルテキスト範囲 (range) クエリーが失敗する

QueryRenderer のバグが原因で、数字を範囲とするフルテキスト範囲 (range) クエリーで **NullPointerException** が発生しました。

JDG-989 - インデクシングが有効であると Type のあるキャッシュテンプレートが失敗する

場合によっては、キャッシュテンプレートでインデクシングを使用すると **NullPointerException** が発生することがありました。

JDG-952 - 集計のリモートクエリーで Null 値がカウントされる

Hot Rod クライアントの使用時にフィールドに **Store.YES** アノテーションが付けられると、フィールドを使用する null 値がカウントされました。

JDG-893 - protobuf に @Analyzer および @Field.analyzer を実装

@Analyzer および **@Field.analyzer** アノテーションはすでに認識されていましたが、有効ではありませんでした。

JDG-663 - 演算子 '=' がフルテキストのクエリーで動作しない

from Entity t where t.longDescription : ('-really') のようなクエリーを実行する場合、単語 **really** が含まれないすべてのエンティティーが出力されるはずですが、出力されませんでした。

JDG-664 - 用語 (term) クエリーの正しいオペランドが分析されない

デフォルトの設定とデフォルトのアナライザーを使用する場合、クエリーは「小文字」である必要があります。たとえば、**from Entity where longDescription:'rent'** と **from Entity where longDescription:'RENT'** の2つのクエリーは同じ結果を返す必要がありますが、実際にはこれらのクエリーは同じ結果を返しませんでした。

