



Red Hat Data Grid 8.5

Data Grid での Memcached プロトコルエンドポイントの使用

Data Grid Memcached エンドポイントを使用してリモートキャッシュと対話する

Red Hat Data Grid 8.5 Data Grid での Memcached プロトコルエンドポイントの使用

Data Grid Memcached エンドポイントを使用してリモートキャッシュと対話する

法律上の通知

Copyright © 2024 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, the Red Hat 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 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

Memcached プロトコルエンドポイントを使用すると、クライアントは1つ以上の Data Grid でサポートされる Memcached サーバーに接続し、キャッシュ操作を実行できます。

目次

RED HAT DATA GRID	3
DATA GRID のドキュメント	4
DATA GRID のダウンロード	5
多様性を受け入れるオープンソースの強化	6
第1章 MEMCACHED サーバー	7
1.1. クライアントエンコーディング	7
1.2. プロトコル検出	7
1.3. セキュリティー	7
1.4. サポートされない機能	8

RED HAT DATA GRID

Data Grid は、高性能の分散型インメモリーデータストアです。

スキーマレスデータ構造

さまざまなオブジェクトをキーと値のペアとして格納する柔軟性があります。

グリッドベースのデータストレージ

クラスター間でデータを分散および複製するように設計されています。

エラスティックスケールリング

サービスを中断することなく、ノードの数を動的に調整して要件を満たします。

データの相互運用性

さまざまなエンドポイントからグリッド内のデータを保存、取得、およびクエリーします。

DATA GRID のドキュメント

Data Grid のドキュメントは、Red Hat カスタマーポータルで入手できます。

- [Data Grid 8.5 ドキュメント](#)
- [Data Grid 8.5 コンポーネントの詳細](#)
- [Data Grid 8.5 でサポートされる構成](#)
- [Data Grid 8 機能のサポート](#)
- [Data Grid で非推奨の機能](#)

DATA GRID のダウンロード

Red Hat カスタマーポータルで [Data Grid Software Downloads](#) にアクセスします。



注記

Data Grid ソフトウェアにアクセスしてダウンロードするには、Red Hat アカウントが必要です。

多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、用語の置き換えは、今後の複数のリリースにわたって段階的に実施されます。詳細は、[弊社の CTO、Chris Wright のメッセージ](#) を参照してください。

第1章 MEMCACHED サーバー

Data Grid サーバーディストリビューションには、Memcached [テキスト](#) および [バイナリー](#) プロトコルを実装するサーバーモジュールが含まれています。これにより、Memcached クライアントは1つまたは複数の Data Grid がサポートする Memcached サーバーに接続できるようになります。これらのサーバーは Memcached のようにスタンドアロンにすることができ、各サーバーは独立して動作し、他のサーバーと通信しません。また、サーバーをクラスター化して、サーバーがコンテンツを他の Data Grid でサポートされた Memcached サーバーにレプリケートまたはディストリビュートし、クライアントにフェイルオーバー機能を提供することもできます。

Memcached エンドポイントは、シングルポートエンドポイントではデフォルトで有効になっていません。Memcached クライアント接続は自動的に検出され、内部コネクタにルーティングされます。

1.1. クライアントエンコーディング

Memcached テキストプロトコルでは、クライアントによって読み書きされるデータ値が raw バイトであると想定しています。

Memcached クライアントが、データ型をネゴシエートしてサーバーからデータを取得したり、異なる形式でデータを送信したりすることはできません。オプションで、特定のメディアタイプでエンコードされた値を処理するようにサーバーを設定することもできます。**memcached-connector** 要素に **client-encoding** 属性を設定すると、サーバーはこの設定された形式でコンテンツを返し、クライアントもこの形式でデータを送信できるようになります。

client-encoding は、Rest、HotRod、Memcached などの複数のリモートエンドポイントから単一のキャッシュにアクセスする場合に便利です。これにより、ユーザーは Memcached テキストクライアントへの応答と要求をカスタマイズすることもできます。エンドポイント間の相互運用性については、エンドポイントの相互運用性のドキュメントを参照してください。

1.2. プロトコル検出

Data Grid Memcached コネクタは、**memcached-connector** 要素の **protocol** 属性を **TEXT**、**BINARY**、**AUTO** のいずれかに設定することにより、手動と自動の両方のプロトコル設定をサポートします。自動検出は、Memcached コネクタが専用のソケットバインディング上にあるか、単一のポートエンドポイントの一部であるかによって異なります。単一ポートエンドポイントでテキストプロトコルクライアントを検出するには、他のプロトコルからの要求を明確にするために認証が必要です。

1.3. セキュリティー

Data Grid Memcached コネクタは、テキストプロトコルとバイナリープロトコルの両方の認証をサポートします。2つのバリエーションは認証を異なる方法で実装するため、サポートされるメカニズムは使用されているプロトコルによって異なります。

1.3.1. テキストプロトコル

Memcached テキストプロトコルでの認証は、キーが無視され、値がスペースで区切られたユーザー名とパスワードの連結となる最初の **set** コマンドによって実行されます。この認証スキームには、プレーンテキストをサポートするセキュリティーレルムが必要です。

1.3.2. バイナリープロトコル

Memcached バイナリープロトコルでの認証は、SASL チャレンジ/レスポンススキームをラップする専用の操作を介して実行されます。この認証スキームは、すべての Data Grid セキュリティーレルム実装で機能します。

1.3.3. Flush All

クラスター化された環境でも、**flush_all** コマンドは呼び出しが行われる Data Grid Memcached サーバーのみをクリアします。このフラッシュをクラスター内の他のノードに伝播する試みはありません。これは、遅延のある **flush_all** のユースケースを Data Grid Memcached サーバーで再現できるようにするためです。**flush_all** に遅延を渡す目的は、異なる Memcached サーバーを異なる時間にフラッシュできるようにし、すべての Memcached サーバーが空になったことにより、リクエストによってデータベースが過負荷になるのを回避することです。詳細は、[Memcached テキストプロトコルの flush_all に関するセクション](#) を参照してください。

1.4. サポートされない機能

このセクションでは、Data Grid ベースの Memcached 実装で現在サポートされていない Memcached テキストプロトコルの機能について詳しく説明します。

1.4.1. 個々の統計情報

C/C++ ベースのオリジナルの Memcached 実装と、Java ベースの Data Grid 実装には違いがあります。サポートされていない汎用統計情報がいくつかあります。これらのサポートされていない統計情報については、Data Grid Memcached サーバーは常に **0** を返します。

サポートされていない統計情報

- pointer_size
- rusage_user
- rusage_system
- bytes
- curr_connections
- total_connections
- connection_structures
- auth_cmds
- auth_errors
- limit_maxbytes
- threads
- conn_yields
- reclaimed

統計情報の設定

統計情報の設定はサポートされていません。

引数パラメーターによる設定

Data Grid Memcached サーバーは、**stats** コマンドに引数を渡すことをサポートしていません。パラメーターが渡された場合、Data Grid Memcached サーバーは **CLIENT_ERROR** で応答します。

hold time パラメーターの削除

Memcached は、delete コマンドのオプションの hold time パラメーターを使用しなくなったため、Data Grid ベースの Memcached サーバーではこの機能は実装されません。

verbosity コマンド

Data Grid ロギングはログレベルのみの定義に簡略化できないため、verbosity コマンドはサポートされていません。

メタプロトコル

Memcached 1.6 で導入された Memcached メタプロトコルはサポートされていません。