

RDMA for Apache HBase 0.9.1 User Guide

HIGH-PERFORMANCE BIG DATA TEAM

<http://hibd.cse.ohio-state.edu>

NETWORK-BASED COMPUTING LABORATORY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
THE OHIO STATE UNIVERSITY

Copyright (c) 2011-2016
Network-Based Computing Laboratory,
headed by Dr. D. K. Panda.
All rights reserved.

Last revised: November 10, 2016

Contents

1	Overview of the RDMA for Apache HBase Project	1
2	Features	1
3	Installation Instructions	1
3.1	Prerequisites	1
3.2	Download	1
3.3	Installation	2
4	Configuration	2
4.1	Basic Configuration	2
4.2	Advanced Configuration	4
5	Basic Usage Instructions	4
5.1	Startup	4
5.2	HBase Shell	4
5.3	Shutdown	6
6	RDMA-HBase Benchmarks	7
6.1	Microbenchmarks	7
6.2	YCSB	7
7	Troubleshooting with RDMA-HBase	8

1 Overview of the RDMA for Apache HBase Project

RDMA for Apache HBase is a high-performance design of Apache HBase over RDMA-enabled Interconnects. This version of RDMA for Apache HBase 0.9.1 is based on Apache HBase 1.1.2. This file is intended to guide users through the various steps involved in installing, configuring, and running RDMA for Apache HBase over InfiniBand. If there are any questions, comments or feedbacks regarding this software package, please post them to rdma-hadoop-discuss mailing list (rdma-hadoop-discuss@cse.ohio-state.edu).

2 Features

High-level features of RDMA for Apache HBase 0.9.1 are listed below.

- Based on Apache HBase 1.1.2
- High performance design with native InfiniBand and RoCE support at the verbs level for Apache HBase
- Compliant with Apache HBase 1.1.2 APIs and applications
- Easily configurable for native InfiniBand, RoCE, and the traditional sockets based support (Ethernet and InfiniBand with IPoIB)
- On-demand connection setup
- Tested with
 - Native Verbs-level support with Mellanox InfiniBand adapters (DDR, QDR, FDR, and EDR)
 - RoCE support with Mellanox adapters
 - Various multi-core platforms

3 Installation Instructions

3.1 Prerequisites

In order to use the RDMA-based features provided with RDMA for Apache HBase, install the latest version of the OFED distribution that can be obtained from <http://www.openfabrics.org>.

3.2 Download

The latest version of RDMA for Apache HBase package can be downloaded from <http://hibd.cse.ohio-state.edu/download/hibd/rdma-hbase-0.9.1-bin.tar.gz>.

3.3 Installation

The following steps can be used to install the RDMA for Apache HBase package.

1. Extract the RDMA for Apache HBase distribution tarball using the following command:

```
tar -xzf rdma-hbase-0.9.1-bin.tar.gz
```

2. Change directory to rdma-hbase-0.9.1

```
cd rdma-hbase-0.9.1
```

Please email us at rdma-hadoop-discuss@cse.ohio-state.edu if you experience any trouble installing the package on your system.

4 Configuration

4.1 Basic Configuration

Steps to configure RDMA for Apache HBase include:

1. Configure hbase-env.sh file.

```
export JAVA_HOME=/opt/java/1.7.0
```

2. Configure hbase-site.xml file. RDMA for Apache HBase supports three different modes: IB, RoCE, and TCP/IP.

Configuration of IB mode:

```
<configuration>
  <property>
    <name>hbase.rootdir</name>
    <value>hdfs://node001:9000/hbase</value>
    <description>The directory shared by region servers.
    </description>
  </property>
  <property>
    <name>hbase.ib.enabled</name>
    <value>true</value>
    <description>Disable the RDMA feature over IB. Default value of
      hbase.ib.enabled is true.</ description>
  </property>
  <property>
    <name>hbase.roce.enabled</name>
    <value>>false</value>
```

```
<description>Disable the RDMA feature over RoCE. Default value
  of hbase.roce.enabled is false.</description>
</property>
```

Configuration of the RoCE mode:

```
<configuration>
  <property>
    <name>hbase.rootdir</name>
    <value>hdfs://node001:9000/hbase</value>
    <description>The directory shared by region servers.
    </description>
  </property>
  <property>
    <name>hbase.ib.enabled</name>
    <value>>false</value>
    <description>Disable the RDMA feature over IB. Default value of
      hbase.ib.enabled is true.</ description>
  </property>
  <property>
    <name>hbase.roce.enabled</name>
    <value>>true</value>
    <description>Disable the RDMA feature over RoCE. Default value
      of hbase.roce.enabled is false.</description>
  </property>
```

Configuration of the TCP/IP Mode:

```
<configuration>
  <property>
    <name>hbase.rootdir</name>
    <value>hdfs://node001:9000/hbase</value>
    <description>The directory shared by region servers.
    </description>
  </property>
  <property>
    <name>hbase.ib.enabled</name>
    <value>>false</value>
    <description>Disable the RDMA feature over IB. Default value of
      hbase.ib.enabled is true.</ description>
  </property>
  <property>
    <name>hbase.roce.enabled</name>
    <value>>false</value>
    <description>Disable the RDMA feature over RoCE. Default value
      of hbase.roce.enabled is false.</description>
  </property>
```

3. Configure regionservers file. List all regionserver hostnames in this file, one per line.

```
node002
node003
```

We can also configure more specific items according to actual needs. For example, we can configure the item `hbase.zookeeper.property.dataDir` in `hbase-site.xml` to change the zookeeper data directory. To get more detailed information, please visit <http://hbase.apache.org>.

4.2 Advanced Configuration

There is one parameter specific to RDMA-HBase that can be manually configured by users in the `hbase-site.xml` file. This parameter controls the number of RDMA handler threads in the regionserver. We recommend that this parameter be tuned to get best performance.

```
<property>
  <name>hbase.rdma.handler.count</name>
  <value>2</value>
  <description>Number of RDMA handler threads. Default value of this
    parameter is 4.</description>
</property>
```

5 Basic Usage Instructions

RDMA for Apache HBase 0.9.1 has management options similar to default Apache HBase 1.1.2. This sections lists the basic usage instructions.

5.1 Startup

Before starting the HBase cluster, make sure to start the Hadoop HDFS cluster. Use the following command to start the HBase cluster:

```
bin/start-hbase.sh
```

5.2 HBase Shell

The HBase Shell can be started using the following command:

```
bin/hbase shell
```

Type `status` and then `<RETURN>` to see that status of the cluster.

```
hbase(main):002:0> status
1 servers, 0 dead, 2.0000 average load
```

Type `help` and then `<RETURN>` to see a listing of shell commands and options.

```
hbase(main):003:0> help
HBase Shell, version 1.1.2, r273b69f443ebaec3e84597316f0f690a96ab713,
  Fri Jul 1 16:12:46 EDT 2016
Type 'help "COMMAND"', (e.g. 'help "get"' -- the quotes are necessary)
  for help on a specific command.
Commands are grouped. Type 'help "COMMAND_GROUP"', (e.g. 'help
  "general"') for help on a command group.

COMMAND GROUPS:
Group name: general
Commands: status, table_help, version, whoami

Group name: ddl
Commands: alter, alter_async, alter_status, create, describe,
  disable, disable_all, drop, drop_all, enable, enable_all, exists,
  get_table, is_disabled, is_enabled, list, show_filters

Group name: namespace
Commands: alter_namespace, create_namespace, describe_namespace,
  drop_namespace, list_namespace, list_namespace_tables

Group name: dml
Commands: append, count, delete, deleteall, get, get_counter,
  get_splits, incr, put, scan, truncate, truncate_preserve

Group name: tools
Commands: assign, balance_switch, balancer, balancer_enabled,
  catalogjanitor_enabled, catalogjanitor_run, catalogjanitor_switch,
  close_region, compact, compact_rs, flush, major_compact,
  merge_region, move, split, trace, unassign, wal_roll, zk_dump

Group name: replication
Commands: add_peer, append_peer_tableCFs, disable_peer,
  disable_table_replication, enable_peer, enable_table_replication,
  list_peers, list_replicated_tables, remove_peer,
  remove_peer_tableCFs, set_peer_tableCFs, show_peer_tableCFs

Group name: snapshots
Commands: clone_snapshot, delete_all_snapshot, delete_snapshot,
  list_snapshots, restore_snapshot, snapshot

Group name: configuration
Commands: update_all_config, update_config
```

```
Group name: quotas
Commands: list_quotas, set_quota

Group name: security
Commands: grant, revoke, user_permission

Group name: visibility labels
Commands: add_labels, clear_auths, get_auths, list_labels, set_auths,
         set_visibility
```

SHELL USAGE:

Quote all names in HBase Shell such as table and column names. Commas delimit

command parameters. Type <RETURN> after entering a command to run it.

Dictionaries of configuration used in the creation and alteration of tables are

Ruby Hashes. They look like this:

```
{'key1' => 'value1', 'key2' => 'value2', ...}
```

and are opened and closed with curly-braces. Key/values are delimited by the

'=>' character combination. Usually keys are predefined constants such as

NAME, VERSIONS, COMPRESSION, etc. Constants do not need to be quoted.

Type

'Object.constants' to see a (messy) list of all constants in the environment.

If you are using binary keys or values and need to enter them in the shell, use

double-quote'd hexadecimal representation. For example:

```
hbase> get 't1', "key\x03\x3f\xcd"
hbase> get 't1', "key\003\023\011"
hbase> put 't1', "test\xef\xff", 'f1:', "\x01\x33\x40"
```

5.3 Shutdown

Stop the cluster with the following command:

```
bin/stop-hbase.sh
```


6 RDMA-HBase Benchmarks

6.1 Microbenchmarks

OSU HiBD-Benchmarks (OHB-0.9.2) provides benchmarks for HBase, Hadoop, Spark, and Memcached. For HBase, Put and Get latency microbenchmarks are provided. The source code can be downloaded from <http://hibd.cse.ohio-state.edu/download/hibd/osu-hibd-benchmarks-0.9.2.tar.gz>. The source can be compiled with the help of the Maven. More details on building and running the OHB Micro-benchmark are provided in the README.

A brief description of the OSU HiBD-Benchmarks for HBase is provided below:

Get Latency Benchmark:

This micro-benchmark measures latency of a Get operation for different data sizes.

Put Latency Benchmark:

This micro-benchmark measures latency of a Put operation for different data sizes.

There is one parameter (`--autoflush`) which can be set to true or false. This will enable or disable the autoflush feature provided by HBase. The benchmarks can be run after starting HBase using the following command:

```
java -Djava.ext.dirs=<HBASE_HOME_DIR>/lib
  -Djava.library.path=<HBASE_HOME_DIR>/lib/native/Linux-amd64-64/ -cp
  <HBASE_HOME_DIR>/conf:<HBASE_HOME_DIR>/lib:<OHB_INSTALL_PATH>/hbase/
  target/ohb-hbase-0.9.2.jar
  edu.osu.hibd.ohb.hbase.HBaseSingleClient<Put|Get> --auto-flush
  <true|false> 2> stderr.log
```

6.2 YCSB

YCSB is a popular benchmark for NoSQL database management systems. YCSB provides with different workloads, each with a different combination and proportion of primitive database operations (like Get, Put, Update, Scan, etc.).

Follow these instructions to run YCSB against RDMA for Apache HBase.

1. Download and extract the latest YCSB package from [here](#)
2. Start the HBase cluster
3. Create a HBase table for testing

```
bin/hbase shell
hbase(main):001:0> n_splits = 200 # HBase recommends (10 * number
  of regionservers)
hbase(main):002:0> create 'usertable', 'family', {SPLITS =>
  (1..n_splits).map {|i| "user#{1000+i*(9999-1000)/n_splits}"} }
```

4. Copy HBase lib files

```
cp HBASE-HOME-DIR/lib/* YCSB-HBASE-DRIVER-HOME/lib
```

5. Set LD_LIBRARY_PATH

```
export LD_LIBRARY_PATH=HBASE-HOME-DIR/lib/native/Linux-amd64-64
```

6. Load Data in the table

```
YCSB-HOME/bin/ycsb load hbase10 -P workloads/workloada -cp  
HBASE-HOME-DIR/conf:HBASE-HOME-DIR/lib/native/Linux-amd64-64 -p  
table=usertable -p columnfamily=family
```

7. Run the workload

```
bin/ycsb run hbase10 -P workloads/workloada -cp  
HBASE-HOME-DIR/conf:HBASE-HOME-DIR/lib/native/Linux-amd64-64 -p  
table=usertable -p columnfamily=family
```

Repeat steps 5-7 for each client.

7 Troubleshooting with RDMA-HBase

If you are experiencing any problems with the RDMA-HBase package, please feel free to contact us by sending an email to rdma-hadoop-discuss@cse.ohio-state.edu.