NOTE:

THIS HARDWARE, SOFTWARE OR TEST SUITE PRODUCT (“PRODUCT(S)”) AND ITS RELATED DOCUMENTATION ARE PROVIDED BY MELLANOX TECHNOLOGIES “AS-IS” WITH ALL FAULTS OF ANY KIND AND SOLELY FOR THE PURPOSE OF AIDING THE CUSTOMER IN TESTING APPLICATIONS THAT USE THE PRODUCTS IN DESIGNATED SOLUTIONS. THE CUSTOMER’S MANUFACTURING TEST ENVIRONMENT HAS NOT MET THE STANDARDS SET BY MELLANOX TECHNOLOGIES TO FULLY QUALIFY THE PRODUCT(S) AND/OR THE SYSTEM USING IT. THEREFORE, MELLANOX TECHNOLOGIES CANNOT AND DOES NOT GUARANTEE OR WARRANT THAT THE PRODUCTS WILL OPERATE WITH THE HIGHEST QUALITY. ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT ARE DISCLAIMED. IN NO EVENT SHALL MELLANOX BE LIABLE TO CUSTOMER OR ANY THIRD PARTIES FOR ANY DIRECT, INDIRECT, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES OF ANY KIND (INCLUDING, BUT NOT LIMITED TO, PAYMENT FOR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY FROM THE USE OF THE PRODUCT(S) AND RELATED DOCUMENTATION EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
# Table of Contents

Table of Contents .......................................................... 3  
List of Tables ................................................................. 4  
Document Revision History .................................................. 5  
About This Manual ............................................................ 7  
Intended Audience ............................................................ 7  
Typographical Conventions .................................................. 7  
Common Abbreviations and Acronyms ..................................... 8  
Glossary ............................................................................ 9  
Related Documentation ....................................................... 11  
Support and Updates Webpage ................................................ 12  
Chapter 1 Overview ............................................................. 13  
1.1 Mellanox UDA Solution .................................................... 13  
1.2 Mellanox OFED for Linux .................................................. 13  
Chapter 2 Hardware Setup .................................................... 14  
2.1 Setting up the Adapter Cards .............................................. 14  
2.2 Setting up the Switch System ............................................. 14  
Chapter 3 Installing, Configuring and Running UDA Software ........ 15  
3.1 Supported Operating Systems ............................................ 15  
3.2 Installation Prerequisites .................................................. 15  
3.3 Installing UDA ............................................................... 16  
3.4 UDA Configuration ......................................................... 18  
3.4.1 RDMA Plug-in Parameters Basic Tuning Guidelines ............ 21  
3.5 UDA Log Setting ............................................................ 22  
3.6 Running UDA ............................................................... 22  
Appendix A Patching and Building Hadoop .............................. 24  
Appendix B Configuring UDA on a CDH5+ Cluster via Cloudera Manager .25  
Appendix C Configuring UDA on a HDP Cluster via Ambari ............ 27
List of Tables

Table 1: Document Revision History ............................................. 5
Table 2: Typographical Conventions ........................................... 7
Table 3: Abbreviations and Acronyms ......................................... 8
Table 4: Glossary ........................................................................ 9
Table 5: Reference Documents .................................................... 11
## Document Revision History

**Table 1 - Document Revision History**

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.4.1</td>
<td>November 2015</td>
<td>Added the following section:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Appendix C: “Configuring UDA on a HDP Cluster via Ambari,” on page 27</td>
</tr>
<tr>
<td></td>
<td>December 2014</td>
<td>Updated the following sections:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.2, “Installation Prerequisites”, on page 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3, “Installing UDA”, on page 16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added the following section:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Appendix B: “Configuring UDA on a CDH5+ Cluster via Cloudera Manager,” on page 25</td>
</tr>
<tr>
<td>3.4.0</td>
<td>September 2014</td>
<td>Updated the following sections:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.2, “Installation Prerequisites”, on page 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3, “Installing UDA”, on page 16</td>
</tr>
<tr>
<td>3.3.3</td>
<td>December 2013</td>
<td>Updated the following sections:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3, “Installing UDA”, on page 16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.4, “UDA Configuration”, on page 18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Appendix A: “Patching and Building Hadoop,” on page 24</td>
</tr>
<tr>
<td>3.2.2</td>
<td>September 2013</td>
<td>Updated the following sections:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3, “Installing UDA”, on page 16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.4, “UDA Configuration”, on page 18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Appendix A: “Patching and Building Hadoop,” on page 24</td>
</tr>
<tr>
<td>3.1.11</td>
<td>June 2013</td>
<td>Updated the following sections:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.2, “Installation Prerequisites”, on page 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.3, “Installing UDA”, on page 16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.4, “UDA Configuration”, on page 18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added the following sections:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Section 3.5, “UDA Log Setting”, on page 22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Appendix A: “Patching and Building Hadoop,” on page 24</td>
</tr>
<tr>
<td>3.0</td>
<td>August 2012</td>
<td>Major updates to the following chapters:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Chapter 1, “Overview”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Chapter 2, “Hardware Setup”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Chapter 3, “Installing, Configuring and Running UDA Software”</td>
</tr>
</tbody>
</table>
Table 1 - Document Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
</table>
| 2.1      | April 2012 | • Renamed the document title (was Mellanox Web 2.0 Acceleration Kit Quick Start Guide)  
• Reorganized the sections in Chapter 1, “Overview” and updated links to the software  
• Consolidated all adapter cards HW and SW installation into Section 2.1, “Setting up the Adapter Cards”, on page 14  
• Consolidated all switch system HW and management SW installation into Section 2.2, “Setting up the Switch System”, on page 14 (the details of the HW installation have been removed; the reader is referred to the switch installation guide for the installation details)  
• Added a prerequisite to increase the maximum number of memory translation table segments per HCA in Section 3.2, “Installation Prerequisites”, on page 15  
• Updated EULA path in Section 3.3, “Installing UDA”, on page 16  
• Updated Section 3.4, “UDA Configuration”, on page 18  
• Added Section 3.4.1, “RDMA Plug-in Parameters Basic Tuning Guidelines”, on page 21  
• Updated Section 3.6, “Killing Previous Hadoop Runs,” on page 39 |
| 1.1      | October 2011 | Updated section 3.3, “Mellanox UDA Installation” for UDA 2.0 |
| 1.0      | June 2011   | Initial release |

1.1 October 2011 Updated section 3.3, “Mellanox UDA Installation” for UDA 2.0
1.0 June 2011 Initial release
About This Manual

This document describes the setup and configuration of Mellanox Unstructured Data Acceleration (UDA) software package for Hadoop Map Reduce frameworks.

Intended Audience

This manual is intended for system administrators responsible for the installation, configuration, management and maintenance of Mellanox UDA software. It is also intended for application developers.

Typographical Conventions

<table>
<thead>
<tr>
<th>Description</th>
<th>Convention</th>
</tr>
</thead>
<tbody>
<tr>
<td>File names</td>
<td>file.extension</td>
</tr>
<tr>
<td>Directory names</td>
<td>directory</td>
</tr>
<tr>
<td>Commands and their parameters</td>
<td>command param1</td>
</tr>
<tr>
<td>Optional items</td>
<td>[ ]</td>
</tr>
<tr>
<td>Mutually exclusive parameters</td>
<td>{ p1</td>
</tr>
<tr>
<td>Optional mutually exclusive parameters</td>
<td>[ p1</td>
</tr>
<tr>
<td>Prompt of a user command under bash shell</td>
<td>hostname$</td>
</tr>
<tr>
<td>Prompt of a root command under bash shell</td>
<td>hostname#</td>
</tr>
<tr>
<td>Prompt of a user command under tcsh shell</td>
<td>tcsh$</td>
</tr>
<tr>
<td>Environment variables</td>
<td>VARIABLE</td>
</tr>
<tr>
<td>Code example</td>
<td>if (a==b){};</td>
</tr>
<tr>
<td>Comment at the beginning of a code line</td>
<td>!, #</td>
</tr>
<tr>
<td>Characters to be typed by users as-is</td>
<td>bold font</td>
</tr>
<tr>
<td>Keywords</td>
<td>bold font</td>
</tr>
<tr>
<td>Variables for which users supply specific values</td>
<td>Italic font</td>
</tr>
<tr>
<td>Emphasized words</td>
<td>Italic font</td>
</tr>
<tr>
<td>Pop-up menu sequences</td>
<td>menu1 --&gt; menu2 --&gt;... --&gt; item</td>
</tr>
<tr>
<td>Note</td>
<td>&lt;text&gt;</td>
</tr>
</tbody>
</table>
### Common Abbreviations and Acronyms

#### Table 3 - Abbreviations and Acronyms (Sheet 1 of 2)

<table>
<thead>
<tr>
<th>Abbreviation / Acronym</th>
<th>Whole Word / Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>(Capital) ‘B’ is used to indicate size in bytes or multiples of bytes (e.g., 1KB = 1024 bytes, and 1MB = 1048576 bytes)</td>
</tr>
<tr>
<td>b</td>
<td>(Small) ‘b’ is used to indicate size in bits or multiples of bits (e.g., 1Kb = 1024 bits)</td>
</tr>
<tr>
<td>FCoE</td>
<td>Fibre Channel over Ethernet</td>
</tr>
<tr>
<td>FW</td>
<td>Firmware</td>
</tr>
<tr>
<td>HCA</td>
<td>Host Channel Adapter</td>
</tr>
<tr>
<td>HW</td>
<td>Hardware</td>
</tr>
<tr>
<td>IB</td>
<td>InfiniBand</td>
</tr>
<tr>
<td>LSB</td>
<td>Least significant byte</td>
</tr>
<tr>
<td>lsb</td>
<td>Least significant bit</td>
</tr>
<tr>
<td>MSB</td>
<td>Most significant byte</td>
</tr>
<tr>
<td>msb</td>
<td>Most significant bit</td>
</tr>
<tr>
<td>NIC</td>
<td>Network Interface Card</td>
</tr>
<tr>
<td>SW</td>
<td>Software</td>
</tr>
<tr>
<td>VPI</td>
<td>Virtual Protocol Interconnect</td>
</tr>
<tr>
<td>IPoIB</td>
<td>IP over InfiniBand</td>
</tr>
<tr>
<td>PFC</td>
<td>Priority Flow Control</td>
</tr>
<tr>
<td>PR</td>
<td>Path Record</td>
</tr>
<tr>
<td>RDS</td>
<td>Reliable Datagram Sockets</td>
</tr>
<tr>
<td>RoCE</td>
<td>RDMA over Converged Ethernet</td>
</tr>
<tr>
<td>SDP</td>
<td>Sockets Direct Protocol</td>
</tr>
<tr>
<td>SL</td>
<td>Service Level</td>
</tr>
</tbody>
</table>
Glossary

The following is a list of concepts and terms related to InfiniBand in general and to Subnet Managers in particular. It is included here for ease of reference, but the main reference remains the InfiniBand Architecture Specification.

Table 4 - Glossary

<table>
<thead>
<tr>
<th>Abbreviation / Acronym</th>
<th>Whole Word / Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA (Channel Adapter)</td>
<td>A device which terminates an InfiniBand link, and executes transport level functions</td>
</tr>
<tr>
<td>CLI</td>
<td>Command Line Interface. A user interface in which you type commands at the prompt</td>
</tr>
<tr>
<td>DMA (Direct Memory Access)</td>
<td>Allows hardware to move data blocks directly to the memory, bypassing the CPU</td>
</tr>
<tr>
<td>DNS</td>
<td>Domain Name System. A hierarchical naming system for devices in a computer network</td>
</tr>
<tr>
<td>Fabric Management</td>
<td>The use of a set of tools (APIs) to configure, discover, and manage and a group of devices organized as a connected fabric.</td>
</tr>
<tr>
<td>Gateway</td>
<td>A network node that interfaces with another network using a different network protocol</td>
</tr>
<tr>
<td>GUID (Globally Unique IDentifier)</td>
<td>A 64-bit number that uniquely identifies a device or component in a subnet</td>
</tr>
<tr>
<td>GID (Global IDentifier)</td>
<td>A 128-bit number used to identify a Port on a network adapter (see below), a port on a Router, or a Multicast Group.</td>
</tr>
<tr>
<td>HA (High Availability)</td>
<td>A system design protocol that provides redundancy of system components, thus enables overcoming single or multiple failures and minimal downtime</td>
</tr>
</tbody>
</table>
### Table 4 - Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>A computer platform executing an Operating System which may control one or more network adapters</td>
</tr>
<tr>
<td>Hadoop</td>
<td>Open source, distributed, big data processing application. (an Apache project)</td>
</tr>
<tr>
<td>IB</td>
<td>InfiniBand</td>
</tr>
<tr>
<td>LID (Local IDentifier)</td>
<td>A 16 bit address assigned to end nodes by the subnet manager Each LID is unique within its subnet.</td>
</tr>
<tr>
<td>MTU (Maximum Transfer Unit)</td>
<td>The maximum size of a packet payload (not including headers) that can be sent/received from a port</td>
</tr>
<tr>
<td>Network Adapter</td>
<td>A hardware device that allows for communication between computers in a network</td>
</tr>
<tr>
<td>QoS or Quality of Service</td>
<td>Quality of service is the ability to manage different applications or users by priority such that a required bit rate, delay, packet dropping probability, and/or other measures may be guaranteed.</td>
</tr>
<tr>
<td>RDMA (Remote Direct Memory Access)</td>
<td>Allows accessing memory on a remote side without involvement of the remote CPU</td>
</tr>
<tr>
<td>SA (Subnet Administrator)</td>
<td>The interface for querying and manipulating subnet management data</td>
</tr>
<tr>
<td>SSH</td>
<td>Secure Shell. A protocol (program) for securely logging into and running programs on remote machines across a network. The program authenticates access to the remote machine and encrypts the transferred information through the connection.</td>
</tr>
<tr>
<td>Subnet Manager (SM)</td>
<td>An entity that configures and manages the subnet, discovers the network topology, assigns LIDs, determines the routing schemes and sets the routing tables. There is only one master SM and possibly several slaves (Standby mode) at a given time. The SM administers switch routing tables thereby establishing paths through the fabric</td>
</tr>
<tr>
<td>TCA (Target Channel Adapter)</td>
<td>A Channel Adapter that is not required to support verbs, usually used in I/O devices</td>
</tr>
<tr>
<td>UDA</td>
<td>Unstructured Data Acceleration</td>
</tr>
<tr>
<td>UDA Plugin</td>
<td>A software plugin that plugs into the Hadoop application</td>
</tr>
<tr>
<td>WebUI</td>
<td>Web User Interface. A user interface in which you select commands from drop down menus or by clicking on icons</td>
</tr>
</tbody>
</table>
## Related Documentation

### Table 5 - Reference Documents

<table>
<thead>
<tr>
<th>Document Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InfiniBand Architecture Specification, Vol. 1, Release 1.2.1</td>
<td>The InfiniBand Architecture Specification that is provided by IBTA</td>
</tr>
<tr>
<td>Mellanox OFED for Linux</td>
<td>Software and documentation can be found at <a href="http://www.mellanox.com/content/pages.php?pg=products_dyn&amp;product_family=26&amp;menu_section=34">http://www.mellanox.com/content/pages.php?pg=products_dyn&amp;product_family=26&amp;menu_section=34</a></td>
</tr>
<tr>
<td>Firmware Release Notes for Mellanox adapter devices</td>
<td>See the Release Notes PDF file relevant to your adapter device under docs/ folder of installed package.</td>
</tr>
<tr>
<td>SX6036 SwitchX® 1U 36 Port FDR 56Gb/s InfiniBand Switch Installation Guide Document No. 3489</td>
<td>This manual provides installation and set-up instructions for the SX6036 FDR top of rack InfiniBand Switch platforms. See <a href="http://www.mellanox.com/related-docs/user_manuals/SX60XX_Installation_Guide.pdf">http://www.mellanox.com/related-docs/user_manuals/SX60XX_Installation_Guide.pdf</a></td>
</tr>
</tbody>
</table>
Support and Updates Webpage

Please visit the following Web site for downloads, FAQ, troubleshooting, future updates to this manual, etc: http://support.mellanox.com/SupportWeb/software_products/hostacceler_products/UDA.
1 Overview

1.1 Mellanox UDA Solution

Mellanox UDA (Unstructured Data Accelerator) is a software plugin that accelerates Hadoop and improves the scaling of Hadoop clusters executing data-analytics intensive applications. A novel data shuffling protocol is provided for Hadoop to take advantage of RDMA in the network technologies InfiniBand and RoCE (RDMA over Converged Ethernet). Mellanox UDA is an RDMA based software plugin which combined with MLNX Linux (MLNX OFED) inbox driver and ConnectX® based adapter cards will accelerate tasks associated with Map/Reduce file transfer. UDA more than doubles the data processing throughput and reduces CPU utilization by half of Hadoop nodes. Mellanox UDA is developed in collaboration with Auburn University, Alabama.

1.2 Mellanox OFED for Linux

Mellanox OFED for Linux (MLNX_OFED_LINUX) is provided as ISO images, one per a supported Linux distribution, that includes source code and binary RPMs, firmware, utilities, and documentation. The ISO image contains an installation script (called mlnxofedinstall) that performs the necessary steps to accomplish the following:

- Discover the currently installed kernel
- Uninstall any InfiniBand stacks that are part of the standard operating system distribution or another vendor's commercial stack
- Install the MLNX_OFED_LINUX binary RPMs (if they are available for the current kernel)
- Identify the currently installed InfiniBand HCAs and perform the required firmware updates
2 Hardware Setup

2.1 Setting up the Adapter Cards

This manual assumes one or more of the Mellanox ConnectX® family adapter cards is installed in your host machine. Mellanox UDA package takes advantage of the silicon architectures of ConnectX®-3, ConnectX®-2 and ConnectX® based InfiniBand and Ethernet adapter cards. For details, please refer to the relevant adapter card user manual available under www.mellanox.com -> Products -> Adapters.

When using an OEM pre-installed card please refer to the OEM server user manual.

Mellanox UDA requires the installation of Mellanox OFED for Linux driver, version 2.2 or later. Mellanox UDA is currently supported on Linux based machines only. Visit the driver Web page below to access software and documents. The supported Linux distributions and kernels are listed in the release notes file; the installation instructions are provided in the user manual.

See www.mellanox.com -> Products -> Software > InfiniBand/VPI Drivers -> Linux SW/Drivers

2.2 Setting up the Switch System

Mellanox UDA benefits from lossless fabric characteristics and requires an RDMA based network. The RDMA capability is available on InfiniBand and RoCE (RDMA over Converged Ethernet) based networks. For the best performance of Mellanox UDA, it is recommended to use Mellanox Ethernet and InfiniBand switches as the software utilizes their architectures.

Visit www.mellanox.com -> Products -> Switches for the state-of-the-art switch portfolio Mellanox offers for Big Data clusters.
3 Installing, Configuring and Running UDA Software

3.1 Supported Operating Systems

Please refer to the product release notes.

3.2 Installation Prerequisites

Prior to installing UDA on a cluster node:
1. Make sure you have a Hadoop environment installed and running on the node.
2. Make sure `ulimit -l` is set to unlimited in all slaves and master nodes.
   If it is not set:
   a. Add the following line to your `~/.bashrc` file.
   
   ```bash
   ulimit -l unlimited
   ```
   b. Set the parameters below as follow in the `/etc/security/limits.conf` file.

   ```
   * soft memlock unlimited
   * hard memlock unlimited
   ```
   3. Increase the maximum number of memory translation table segments per HCA\(^1\).

   ```
   # echo "options mlx4_core log_num_mtt=24 log_mtts_per_seg=0" > /etc/modprobe.d/mofed.conf
   ```
   a. Reboot the server or restart the openibd.
   To restart the openibd:
   ```bash
   # sudo service openibd restart
   ```
   b. Verify the changes took effect.

   ```
   # cat /sys/module/mlx4_core/parameters/log_num_mtt
   # cat /sys/module/mlx4_core/parameters/log_mtts_per_seg
   ```

4. Disable swap on all the nodes in the cluster.

   The swap option can be disabled as follow:
   - Edit the `/etc/fstab` file. Remove the swap file system and run once the command below.

   ```bash
   # swapoff -a
   ```

5. [Optional] Run Open MPI to verify RDMA connectivity in your cluster:
   a. Set HOSTS variable to a comma delimited list of your cluster's host names.
   b. Set NUM_OF_HOSTS to the number of hosts in your cluster.
   c. Set OPENMPI_VER variable to your Open MPI version.
   d. Set MLX_PORT to the ConnectX port your cluster is using (i.e: for port 1, set it to mlx4_0:1)

---

1. If you need more than 64GB, you can increase the maximum amount of available RDMA memory by increasing the value of `log_mtts_per_seg`.
2. We recommend using this option as it is a one time operation.
3.3 Installing UDA

The following steps describe how to install the UDA distribution.

In case you are using Cloudera Hadoop 5 and above with Cloudera Manager, please refer to Appendix B: “Configuring UDA on a CDH5+ Cluster via Cloudera Manager,” on page 25.

In case you are using HDP2 and above with Ambari, please refer to Appendix C: “Configuring UDA on a HDP Cluster via Ambari,” on page 27.

Step 1. Install Apache Hadoop 1.x.y or Hadoop 2.x.y. The installation guide and configurations of Apache Hadoop are available at hadoop.apache.org.

Step 2. Test your vanilla Hadoop installation to make sure you have a successful and tuned installation. For tuning and configuration details, see http://hadoop.apache.org/common/docs/<Hadoop Version>/cluster_setup.html.

Step 3. [Optional] Patch Hadoop with Mellanox plugin ability patch.

In case you are using one of Hadoop versions below, please patch your Hadoop with Mellanox plugin-ability patch.

- CDH 4.x.y MRv1 prior to CDH 4.4.0
- HDP 1.1
- community hadoop-1.x.y prior to hadoop-1.3.0
- community hadoop-2.x.y prior to hadoop-2.3.0

1. Download the appropriate patch from the UDA's Github repository at http://github.com/Mellanox/UDA/wiki/Downloads.

2. Apply the patch as follows:

```bash
# cd <hadoop extraction directory>
# patch -p0 <patch_name>
# echo $?
```


Step 4. Install the UDA RPM on all cluster nodes.

1. Use the following install command:

```bash
# sudo rpm -Uvh <rpm location>
```
2. Make sure that all the files were successfully installed by running the following query. Expected output is listed as command output below.

```
# rpm -ql libuda
/usr/lib64/uda/LICENSE.txt
/usr/lib64/uda/README
/usr/lib64/uda/journal.txt
/usr/lib64/uda/libuda.so
/usr/lib64/uda/source.tgz
/usr/lib64/uda/uda-hadoop-1.x-old.jar
/usr/lib64/uda/uda-hadoop-1.x.jar
/usr/lib64/uda/uda-hadoop-2.x.jar
/usr/lib64/uda/uda-hadoop-3.x.jar
/usr/lib64/uda/utils.tgz
```

3. Add at the end of your hadoop-env.sh a line containing the jar name matching your hadoop version.
   - For Hadoop 1.x.y, HDP 1.1 add to your hadoop-env.sh add 1:
     ```
     export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/usr/lib64/uda/uda-hadoop-1.x.jar
     ```
   - For Hadoop 2.x.y:
     - Perform the following command on each of your slaves:
       ```
       ln -s /usr/lib64/uda/uda-hadoop-2.x.jar <YOUR-HADOOP-HOME>/share/hadoop/common/lib
       ```
     Note: If you are using CDH, perform:
     ```
     ln -s /usr/lib64/uda/uda-hadoop-2.x.jar <YOUR-HADOOP-HOME>/lib/hadoop/
     ```
   - For CDH4.x.y MRv1:
     a. Run:
     ```
     hadoop version
     ```
     Example output:
     ```
     $ hadoop version
     Hadoop 2.0.0-cdh4.4.0
     Subversion file://data1/jenkins/workspace/generic-package-rhel64-6.0/topdir/BUILD/hadoop-2.0.0-cdh4.4.0/src/hadoop-common-project/hadoop-common -r c0ebac283c984557e96a16cc5756b7de835e79 Compiled by jenkins on Tue Sep  3 19:33:17 PDT 2013 From source with checksum ac7e170aa709b3ace13dc5f775487180 This command was run using /opt/cloudera/parcels/CDH-4.4.0-1.cdh4.4.0.p0.39/lib/hadoop/hadoop-common-2.0.0-cdh4.4.0.jar
     ```
     b. Create a symbolic link to UDA on each slave using the path shown in the last line of the output.
     ```
     sudo ln -s /usr/lib64/uda/uda-hadoop-1.x.jar /opt/cloudera/parcels/CDH-4.4.0-1.cdh4.4.0.p0.39/lib/hadoop/
     ```

   **Step 5.** Restart MapReduce and/or YARN.

---

1. When using the old v1 patch and plugin, add: `export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/usr/lib64/uda/uda-hadoop-1.x-v1.jar`
3.4 UDA Configuration

Assume a cluster with 16 nodes, eagle1 through eagle16, where you wish to set eagle1 as the master of the InfiniBand cluster and the rest as slaves. Similar settings are needed for RoCE based deployments, replacing the InfiniBand host name with the corresponding Ethernet host name.

**Step 1.** For a single homed machines, skip to the next step.
For multi-homed machines, you first need to configure hadoop to use the right interface by setting the “slave.host.name” property. Note, this is a special property and requires each node host to have a unique property value along with the appropriate interface. The host name can be configured as follow:

On all slaves and master, add to the file /etc/hosts the hadoop addresses of all hosts in the cluster (in the format: 40.0.0.1 eagle1.ib.cluster). In this case, configure as follow:

The actions performed below, are supported in Hadoop v1.x.y only.

- hadoop-env.sh:

  ```bash
  export HADOOP_OPTS='-Djava.net.preferIPv4Stack=true -DHADOOPHOSTNAME='hostname'.ib.cluster ${HADOOP_OPTS}"
  ```

- core-site.xml:

  ```xml
  <property>
  <name>slave.host.name</name>
  <value>${HADOOPHOSTNAME}</value>
  </property>
  ```

**Step 2.** XML Configuration:

1. HDFS settings:

   Merge the following lines into your hdfs-site.xml:
   ```xml
   <?xml version="1.0"?>
   <!DOCTYPE configuration SYSTEM "configuration.dtd">
   <configuration>
     <property>
       <name>dfs.datanode.dns.interface</name>
       <value>ib0</value>
       <description>The name of the Network Interface from which a data node should report its IP address.</description>
     </property>
   </configuration>
   ```

2. TaskTracker level settings:

   Merge the following lines into your mapred-site.xml:
   ```xml
   <?xml version="1.0"?>
   <configuration type="text/xsl" href="configuration.xsl"?
   <property>
     <name>mapreduce.framework.name</name>
     <value>local</value>
   </property>
   <property>
     <name>mapreduce.jobtracker.address</name>
     <value>localhost:9000</value>
   </property>
   <property>
     <name>mapreduce.jobtracker.webapp.address</name>
     <value>localhost:9001</value>
   </property>
   </configuration>
   ```

These lines must be in `mapred-site.xml` to be considered during TaskTracker initialization. Therefore, this step cannot be performed per job only.
1. The example below is used for the old v1 version of Mellanox plugin (uda-hadoop-1.x-v1.jar).

```xml
<property>
  <name>mapreduce.shuffle.provider.plugin.classes</name>
  <value>com.mellanox.hadoop.mapred.UdaShuffleProviderPlugin,
  org.apache.hadoop.mapred.TaskTracker$DefaultShuffleProvider</value>
  <description>A comma-separated list of classes that should be loaded as
  ShuffleProviderPlugin(s).
  A ShuffleProviderPlugin can serve shuffle requests from reducers.
  Each class in the list must be an instance of
  org.apache.hadoop.mapred.ShuffleProviderPlugin.
</description>
</property>

<property>
  <name>mapreduce.tasktracker.shuffle.provider.plugin</name>
  <value>com.mellanox.hadoop.mapred.UdaShuffleProviderPlugin</value>
  <description>Represents plugin for shuffle at TaskTracker side.
  Default value is: (empty string)
  You can also try: com.mellanox.hadoop.mapred.UdaShuffleProviderPlugin
</description>
</property>
```
3. Optional settings:

The following are optional default parameter settings for UDA.

```xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
  <property>
    <name>mapred.rdma.compression.buffer.ratio</name>
    <description>The ratio in which memory is divided between RDMA buffer and decompression buffer (used only with intermediate data compression)</description>
    <value>0.20</value>
  </property>
  <property>
    <name>mapred.rdma.cma.port</name>
    <description>Port number to be used for the RDMA connection</description>
    <value>9011</value>
  </property>
  <property>
    <name>mapred.rdma.wqe.per.conn</name>
    <description>Number of allocated Work Queue Elements (WQEs) for Receive Queue per connection.</description>
    <value>256</value>
  </property>
  <property>
    <name>mapred.rdma.buf.size</name>
    <value>1024</value>
    <description>Used by both UdaShuffleProvider and UdaShuffleConsumer:
    - UdaShuffleProvider (TaskTracker): determines the RDMA&AIO Buffers size to satify Map Output's RDMA fetch requests
    - UdaShuffleConsumer (Reducer): user preferred RDMA buffer size for fetching map outputs. Size is in KB and must be aligned to page size.</description>
  </property>
</configuration>
```
4. YARN configuration, for Hadoop-2.x.y. It requires modifying the yarn-site.xml\textsuperscript{1,2}.

\begin{verbatim}
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle,uda_shuffle</value>
</property>

<property>
  <name>yarn.nodemanager.aux-services.uda_shuffle.class</name>
  <value>com.mellanox.hadoop.mapred.UdaShuffleHandler</value>
</property>

<property>
  <name>mapreduce.job.shuffle.provider.services</name>
  <value>uda_shuffle</value>
  <description>A comma-separated list of classes that should be loaded as additional ShuffleProviderPlugin(s). A ShuffleProviderPlugin can serve shuffle requests from reduce tasks.</description>
</property>
\end{verbatim}

5. Verify \texttt{mapreduce.application.classpath} property value in mapred-site.xml contains \texttt{HADOOP_MAPRED_HOME}. If not, add UDA JAR path to \texttt{mapreduce.application.classpath}.

### 3.4.1 RDMA Plug-in Parameters Basic Tuning Guidelines

- UdaShuffleProviderPlugin allocates buffers for reading MOFs from the disk and for writing them using RDMA to satisfy reduce task shuffle requests. Therefore, UdaShuf-

---
\textsuperscript{1} If you are using Cloudera Manager, make sure you insert the above snippet in both Service-Wide and Gateway categories.
\textsuperscript{2} If you already have entry with the name \texttt{"yarn.nodemanager.aux-services.mapreduce_shuffle.class"}, please add this entry in addition to it, do not replace it!
fileProviderPlugin's buffer size determines the max buffer size to be used also by reduce tasks.

- When TaskTracker is spawned and the UdaShuffleProviderPlugin is initialized, it is essential that the mapred.rdma.buf.size parameter is properly configured to satisfy reducers. RDMA buffers from each reducer are allocated from mapred.child.java.opts * mapred.job.shuffle.input.buffer.percent.

When UDA is enabled, each reducer must allocate 2*#MOFs = 2*Dataset/Blocksize. Unless you have memory issues we recommend that each RDMA buffer will be of size 1024 KB for optimum performance.

For example, when running a job with a 100GB input size and a 256MB split size, 600 MOFs are created. This requires configuring at least 1200 buffers. Continuing with the above example, a configuration that runs 4 slots of reducers per node requires the allocation of 4 x1200 = 4800 buffers for the job. By using mapred.rdma.buf.size=1024, a total of 4800MB is allocated per node.

3.5 UDA Log Setting

UDA Consumer and Provider logs are now integrated with the Hadoop log system and their properties are configured via the Hadoop's log4j.properties file (in your <hadoop-conf-dir>).

The Consumer logs are integrated into the ReduceTask whereas the Provider logs are integrated into the TaskTracker.

To configure these modules, add the lines below to the log4j.properties files:

- log4j.logger.org.apache.hadoop.mapred.ShuffleConsumerPlugin=<log_level>
- log4j.logger.org.apache.hadoop.mapred.ShuffleProviderPlugin=<log_level>

The ShuffleProviderPlugin logging level can be changed at runtime. To do so, type:

```bash
#> bin/hadoop daemonlog -setlevel <hostname>:50060 org.apache.hadoop.mapred.ShuffleProvider-Plugin <log level>
```

In the example above, 50060 is the default port value of mapreduce.task-tracker.http.address

If logs are not modified, UDA log level will set to the default setting of the distribution (default is INFO).

3.6 Running UDA

- **To verify UDA was installed properly, run a simple MapReduce job:**

  **Step 1.** Export the following variable in bash.

  ```bash
  ```
To run MapReduce jobs with UDA, the above properties must be passed via the command line or alternatively, set in mapred-site.xml.

**Step 2.** Run Pi.

```bash
hadoop jar <mapreduce-lib-path>/hadoop-mapreduce-examples.jar pi $UDA_ENABLE 8 2000
```

**Step 3.** Verify UDA ran successfully. Make sure the following line exists in the job's reducer log.

```bash
===XXX Successfully closed UdaShuffleConsumerPlugin XXX===
```

If you are using YARN with log aggregation, use the following command to retrieve MapReduce's logs.

```bash
yarn logs -applicationId <applicationId>
```
Appendix A: Patching and Building Hadoop

This section provides the procedure to add to the supported Hadoop distributions the plug-in ability if the ability is lacking, enabling the application to utilize or disable UDA.

If you are using one of Hadoop versions below, please patch your Hadoop with Mellanox plugin-ability patch.

- CDH 4.x.y MRv1 prior to CDH 4.4.0
- HDP 1.1
- community hadoop-1.x.y prior to hadoop-1.3.0
- community hadoop-2.x.y prior to hadoop-2.3.0

To patch and build Hadoop:


Step 2. Extract the tarball on all the nodes and test your installation.


Step 4. Patch hadoop.

1. Extract hadoop-x.y.z.tar.gz in a temp directory.
2. Change directory into the extraction directory.
3. Run the Mellanox patch.

```
$ patch -p0 < HADOOP-1.x.y.patch
```

4. Verify the previous operation was successful.
   The expected result should be 0.

```
$ echo $? 
```

Step 5. Build your patched Hadoop.

Hadoop-1.x.y example:

```
ant -Djava.home=/usr/lib64/java/jdk1.6.0_25 clean tar
```

This will create you a new tar.gz file under the ./build/ dir. (notice that the result will be called hadoop-1.1.3-SNAPSHOT since it is not a default 1.1.2 version).

Hadoop-2.x.y example:

```
mvn package -Pdist -DskipTests -Dtar
```
Appendix B: Configuring UDA on a CDH5+ Cluster via Cloudera Manager

This section was written under the following assumptions:

- Your Cloudera cluster is configured to use an RDMA-supported interface.
- Each of your cluster's nodes has MLNX_OFED 2.2 or later.

If you are using an earlier version, you may have to build UDA manually.

For further information and for the source code, please refer to our Github repository at http://github.com/Mellanox/UDA.

Step 1. Download UDA's RPM from http://github.com/Mellanox/UDA and install it on all nodes:

```
 rpm -i /path/to/libuda-3.4.1-0.1034.el6.x86_64.rpm
```

Step 2. Create a symbolic link to UDA's JAR in ${HADOOP_HOME}/lib/hadoop on all nodes:

```
 ln -s /usr/lib64/uda/uda-hadoop-2.x.jar ${HADOOP_HOME}/lib/hadoop
```

Step 3. Go to the Cloudera Manager:

Step a. Enter Clusters -> Services -> YARN -> Configuration.

Step b. Insert the following property in the advanced code snippet for mapred-site.xml of the Service-Wide category (see Figure 1):

```
<property>
<name>mapreduce.shuffle.provider.plugin.classes</name>
<value>com.mellanox.hadoop.mapred.UdaShuffleProvider-Plugin,org.apache.hadoop.mapred.TaskTracker$DefaultShuffleProvider</value>
</property>
```

Step c. Insert the following properties in the advanced code snippet for yarn-site.xml in both Service-Wide and Gateway categories (see Figure 2). Make sure to Save Changes afterwards:

```
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle,uda_shuffle</value>
</property>
<property>
<name>yarn.nodemanager.aux-services.uda_shuffle.class</name>
<value>com.mellanox.hadoop.mapred.UdaShuffleHandler</value>
</property>
<property>
<name>mapreduce.job.shuffle.provider.services</name>
<value>uda_shuffle</value>
</property>
```

Step d. Verify the mapreduce.application.classpath property value in contains $HADOOP_MAPRED_HOME. If not, add UDA JAR path to mapreduce.application.classpath.

Step 4. Restart YARN.

Go to the Cloudera Manager -> Clusters -> Services -> YARN -> Actions -> Restart
Step 5. [Optional] In BASH, export UDA_ENABLE environment variable.

```
export UDA_ENABLE="-D mapreduce.job.reduce.shuffle.consumer.plugin.class=com.mellanox.hadoop.mapred.UdaShuffleConsumerPlugin"
```

Step 6. Run a small MapReduce job with UDA.

```
hadoop jar /path/to/hadoop-mapreduce-examples.jar pi ${UDA_ENABLE} 8 200
```

Step 7. Verify that UDA was used by checking the reducer’s log. Look for the following:

```
====XXX Successfully closed UdaShuffleConsumerPlugin XXX====
```

In case you have YARN’s log aggregation enabled, the application’s logs can be fetched using:

```
yarn logs -applicationId <applicationId>
```
Appendix C: Configuring UDA on a HDP Cluster via Ambari

This section was written under the following assumptions:

• Your Cloudera cluster is configured to use an RDMA-supported interface.
• Each of your cluster's nodes has MLNX_OFED 2.2 or later.

If you are using an earlier version, you may have to build UDA manually.

For further information and for the source code, please refer to our Github repository at http://github.com/Mellanox/UDA.

Step 1. Download UDA's RPM from http://github.com/Mellanox/UDA and install it on all nodes:

```
rpm -i /path/to/libuda-3.4.1-0.1034.el6.x86_64.rpm
```

Step 2. Create a symbolic link to UDA's JAR in /usr/hdp/current/hadoop-mapreduce-client/lib/ on all nodes:

```
ln -s /usr/lib64/uda/uda-hadoop-2.x.jar /usr/hdp/current/Hadoop-mapreduce-client/lib/
```

Step 3. In the Ambari, enter MapReduce2 -> Configs -> Advanced.

Step a. In the Custom mapred-site insert the following properties (see Figure 3):

```
<property>
  <name>mapreduce.shuffle.provider.plugin.classes</name>
  <value>com.mellanox.hadoop.mapred.UdaShuffleProvider-
  Plugin,org.apache.hadoop.mapred.Task- Tracker$DefaultShuffleProvider</value>
</property>
<property>
  <name>mapreduce.job.shuffle.provider.services</name>
  <value>uda_shuffle</value>
</property>
```

Step b. In the Advanced mapred-site verify that the mapreduce.application.classpath property value contains $HADOOP_MAPRED_HOME. If not, add UDA JAR path to mapreduce.application.classpath.

Step 4. In the Ambari enter Yarn -> Configs -> Advanced:

Step a. In the Custom yarn-site insert the following property (see Figure 4):

```
<property>
  <name>yarn.nodemanager.aux-services.uda_shuffle.class</name>
  <value>com.mellanox.hadoop.mapred.UdaShuffleHandler</value>
</property>
```

Step b. In the Node Manager insert the following property (see Figure 5):

```
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle,uda_shuffle</value>
</property>
```

Step 5. Restart YARN.
In the Ambari -> Yarn- > Service Actions- > Restart all

Step 6. Restart MapReduce2.
In the Ambari -> MapReduce2 -> Service Actions- > Restart all
Step 7. [Optional] In BASH, export UDA_ENABLE environment variable.

   ```bash
   export UDA_ENABLE="-D mapreduce.job.reduce.shuffle.consumer.plugin.class=com.mellanox.hadoop.mapred.UdaShuffleConsumerPlugin"
   ```

Step 8. Run a small MapReduce job with UDA.

   ```bash
   hadoop jar /path/to/hadoop-mapreduce-examples.jar pi $UDA_ENABLE 8 20
   ```

Step 9. Verify that UDA was used by checking the reducer’s log. Look for the following:

   ```none
   ====XXX Successfully closed UdaShuffleConsumerPlugin XXX====
   ```

In case you have YARN’s log aggregation enabled, the application’s logs can be fetched using:

   `yarn logs -applicationId <applicationId>`

---

**Figure 3: Configuring mapred-site.xml via Ambari**

![Configuring mapred-site.xml via Ambari](image)
Figure 4: Configuring yarn-site.xml (Custom yarn-site) via Ambari

Figure 5: Configuring yarn-site.xml (Node Manager) via Ambari