



Pushing Performance

Development Container



People | Power | Partnership

HARTING Hadoop for HAIIC MICA

HARTING IT Software Development
Marienwerder Str. 3, 32339 Espelkamp, Germany
Phone: +49 5572 47-97300, Fax: +49 5772 47-482
mica@HARTING.com



Hadoop v1.1 for HAIC MICA Guide

Content

1.	HADOOP CONTAINER BASICS	3
1.1.	OVERVIEW.....	3
1.2.	INSTALLATION OF THE HADOOP CONTAINER.....	3
2.	DESCRIPTION OF THE HADOOP CONTAINER	4
2.1.	OVERVIEW OF THE USER INTERFACE	4
2.2.	UPLOADING A HADOOP/HDFS/YARN CONFIGURATION FILE.....	5
2.3.	DOWNLOADING THE CURRENT HADOOP/HDFS/YARN CONFIGURATION	6
3.	HDFS USAGE EXAMPLE.....	6
3.1.	LOG INTO THE HADOOP CONTAINER	6
3.2.	FORMAT AND START THE HDFS.....	7
3.3.	STREAM DATA INTO THE HDFS	7
3.4.	RETRIEVE DATA FROM THE HDFS	7
4.	SERVICES PROVIDED BY THE HADOOP CONTAINER	8
4.1.	OVERVIEW.....	8
4.2.	USAGE OF THE SERVICES.....	8

1. Hadoop container basics

1.1. Overview

Apache Hadoop is an open-source software framework written in Java for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common and should be automatically handled by the framework.

For more information about Hadoop visit <https://hadoop.apache.org/>.

1.2. Installation of the Hadoop container

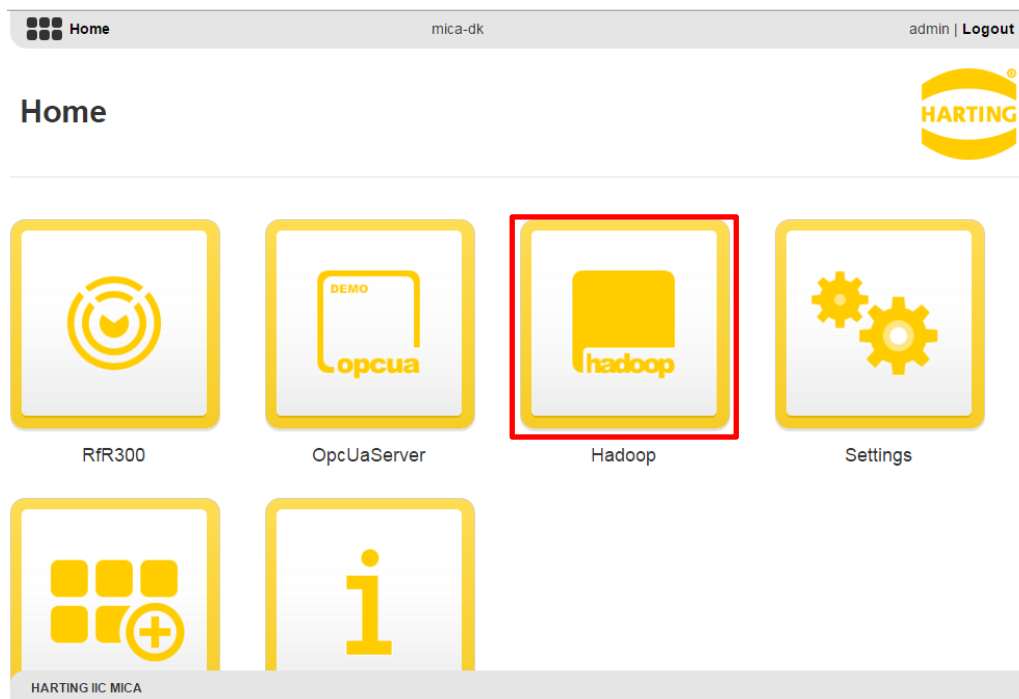


Figure 1: IIC MICA home screen including a (highlighted) Hadoop Container

The installation and configuration routine of the Hadoop Container follows the standard routine as provided by the IIC MICA and can be found in the “MICA Programming Guide”.

2. Description of the Hadoop container

2.1. Overview of the user interface

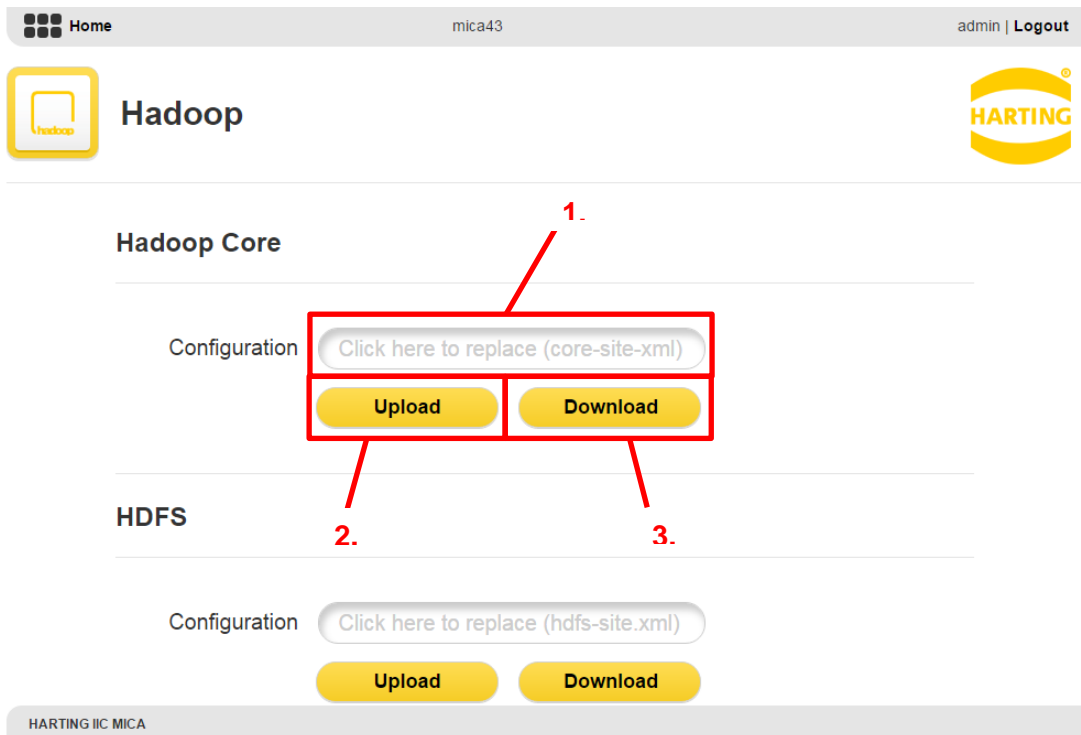


Figure 2: Hadoop Container user-interface

The user-interface of the Hadoop Container consists of three sections “Hadoop Core”, “HDFS” and “Yarn” each including the entry “Configuration”, which can be used to import/export the Hadoop/HDFS/Yarn configuration files to/from the container. The user-interface structure can be described as follows:

- 1.) **Configuration file text field:** The text field can be used to set the Hadoop/HDFS/Yarn configuration file you want to upload to the container.
- 2.) **Upload:** The upload button engages the upload of the configuration file that has been specified in 1.)
- 3.) **Download:** The download button engages the download of the current Hadoop/HDFS/Yarn configuration file to your file system.

2.2. Uploading a Hadoop/HDFS/Yarn configuration file

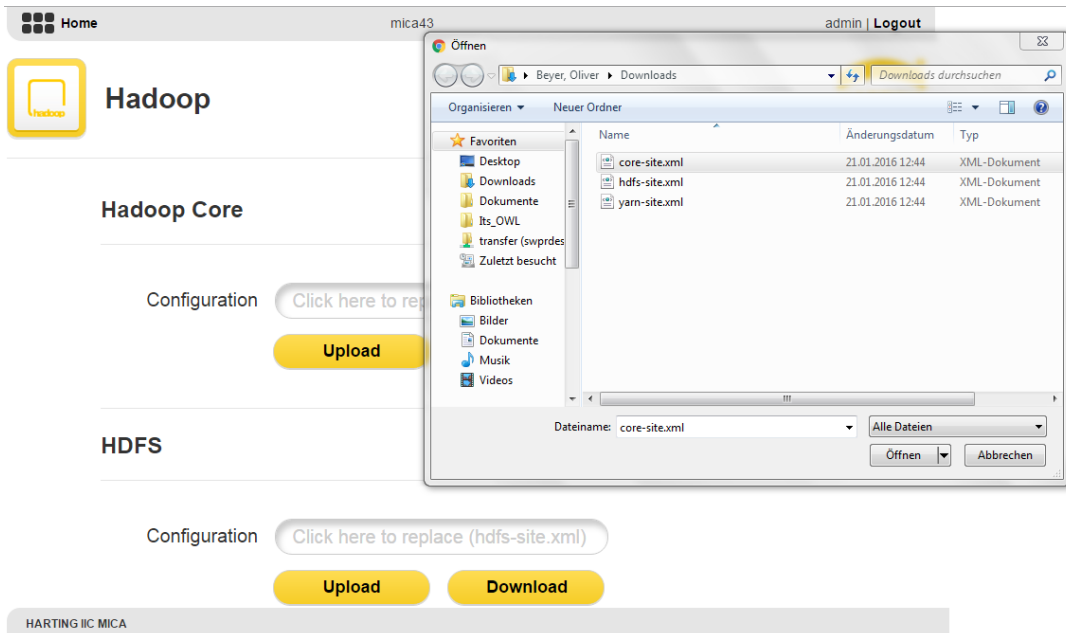


Figure 3: Uploading a configuration to the Hadoop container

In order to upload a new configuration to the container, you have to either click the configuration file text field (see Section 2.1) and select the new Hadoop/HDFS/Yarn configuration file as shown in Figure 3, or you simply drag & drop the file into the text field. After selecting a file the file name is shown in the configuration file text field.

When pressing the button “Upload” the selected file will be sent to the container and stored internally. After the transmission ended a symbol to the right side of the configuration file text file will indicate if the transmission was successful or failed, as shown in Figure 4.

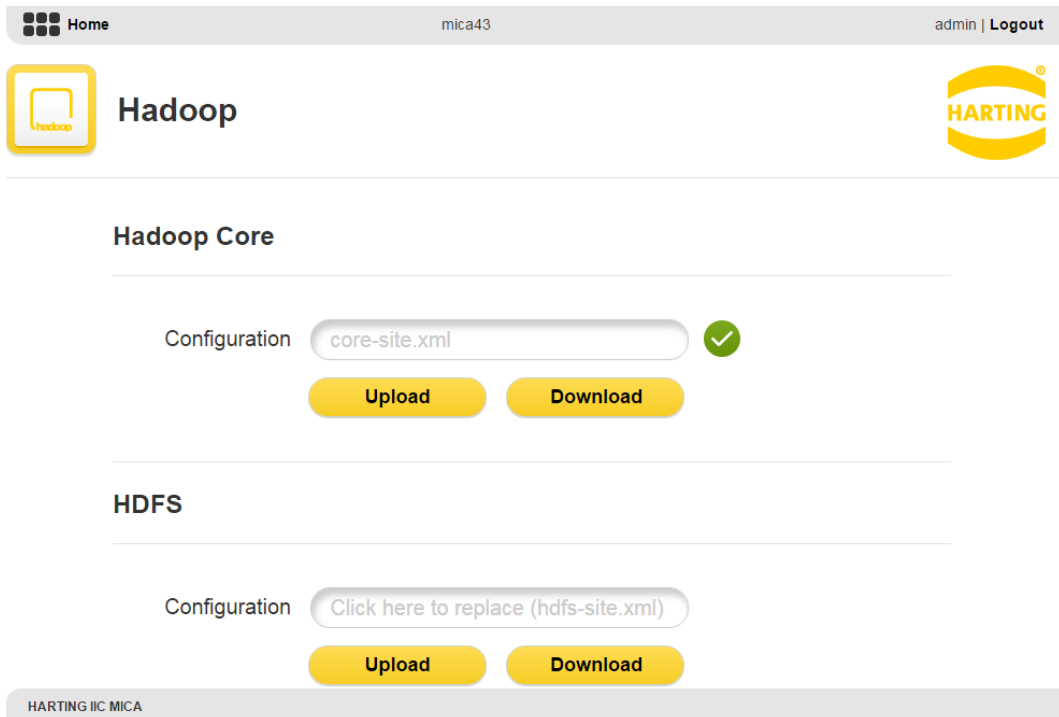


Figure 4: Successful upload of a new configuration to the Hadoop Container

2.3. Downloading the current Hadoop/HDFS/Yarn configuration

You can download the current Hadoop/HDFS/Yarn configuration by simply clicking the download button. After having downloaded the configuration you can open the file in a simple text editor such as “WordPad” (Windows) and “TextEdit” (OS X), in order to see/modify the configuration and re-upload it as described in Section 2.2.

3. HDFS usage Example

3.1. Log into the Hadoop container

At first we will log into the container via SSH, as described in the “MICA Programming Guide”. The Hadoop framework provides a number of command line tools that we want to utilize in this short HDFS example. In the following we will prepare and start the HDFS, stream data into it and retrieve data from it.

3.2. Format and start the HDFS

As that the Hadoop master would usually start the HDFS daemon on the container via SSH automatically, the HDFS and Yarn daemon are not started when the container starts. Thus, we have to format the HDFS and start it manually. In order to format the HDFS we type the following line into our SSH terminal:

```
hdfs namenode -format
```

The default configuration of the HDFS as provided by the container is set to a single-node configuration, which means that we will simultaneously start a `namenode`, `secondary namenode` and `datanode` instance at once. Therefore we type the following line into our SSH terminal:

```
start-dfs.sh
```

Note that this may take a few moments.

Caution: Due to the single-node setup and the three java instances running parallel, each of the following operations take a few moments to be processed.

3.3. Stream data into the HDFS

We will now stream a string from the stdout into the HDFS by using the `hdfs` command line tools. Therefore, we type the following line into the SSH terminal:

```
echo "Hello World!" | hdfs dfs -put - /input
```

After executing this command the string "Hello World!" is written into the new file `input` into the HDFS. When applying `hdfs dfs -ls /` you will see the HDFS root folder and the new file `input`.

3.4. Retrieve data from the HDFS

We will now retrieve the string we just written into the file by again using the `hdfs` command line tools. Therefore, we type the following line into the SSH terminal:

```
hdfs dfs -cat /input
```

After executing this command you should see the string "Hello World!" in your terminal, which was retrieved from the file `input` out of the HDFS.

For more information about setting up a Hadoop cluster visit <https://hadoop.apache.org/>.

4. Services provided by the Hadoop container

4.1. Overview

The Hadoop Container allows to access its functionality as provided by the user-interface remotely over HTTP requests. The container therefore utilizes the token provided by the IIC MICA host to authenticate the request. For more information see the Section “Single sign-on (SSO)” of the “MICA Programming Guide”.

4.2. Usage of the services

The following HTTP-requests are provided by the container:

1. Upload-Core: Uploading a Hadoop core configuration file.
Method: POST
URL : `coreupload?token=XXX`
2. Download-Core: Receiving the content of the current Hadoop core configuration.
Method: POST
URL : `coredownload?token=XXX`
3. Upload-HDFS: Uploading a HDFS configuration file.
Method: POST
URL : `hdfsupload?token=XXX`
4. Download-HDFS: Receiving the content of the current HDFS configuration.
Method: POST
URL : `hdfsdownload?token=XXX`
5. Upload-Yarn: Uploading a Yarn configuration file.
Method: POST
URL : `yarnupload?token=XXX`
6. Download-Yarn: Receiving the content of the current Yarn configuration.
Method: POST
URL : `yarndownload?token=XXX`