

Hadoop Commands

start-all.sh : Starts all Hadoop daemons, the namenode, datanodes, the jobtracker and tasktrackers. Deprecated; use start-dfs.sh then start-yarn.sh

jps : JPS stands for Java Virtual Machine Process Status Tool. It is used to check all the Hadoop daemons like NameNode, DataNode, ResourceManager, NodeManager etc. which are running on the machine.

appendToFile

Usage: `hadoop fs -appendToFile <localsrc> ... <dst>`

Append single src, or multiple srcs from local file system to the destination file system

cat

Usage: `hadoop fs -cat URI [URI ...]`

Copies source paths to stdout.

chgrp

Usage: `hadoop fs -chgrp [-R] GROUP URI [URI ...]`

Change group association of files. The user must be the owner of files, or else a super-user.

chmod

Usage: `hadoop fs -chmod [-R] <MODE[,MODE]... | OCTALMODE> URI [URI ...]`

Change the permissions of files. With -R, make the change recursively through the directory structure. The user must be the owner of the file, or else a super-user.

chown

Usage: `hadoop fs -chown [-R] [OWNER][:[GROUP]] URI [URI]`

Change the owner of files. The user must be a super-user.

copyFromLocal

Usage: `hadoop dfs -copyFromLocal <localsrc> URI`

Similar to put command, except that the source is restricted to a local file reference.

copyToLocal

Usage: `hadoop dfs -copyToLocal [-ignorecrc] [-crc] URI <localdst>`

Similar to get command, except that the destination is restricted to a local file reference.

count

Usage: `hadoop fs -count [-q] <paths>`

Count the number of directories, files and bytes under the paths that match the specified file pattern. The output columns with -count are: DIR_COUNT, FILE_COUNT, CONTENT_SIZE FILE_NAME

Example:

- `hadoop fs -count hdfs://nn1.example.com/file1 hdfs://nn2.example.com/file2`
- `hadoop fs -count -q hdfs://nn1.example.com/file1`

cp

Usage: `hadoop fs -cp [-f] URI [URI ...] <dest>`

Copy files from source to destination. This command allows multiple sources as well in which case the destination must be a directory.

get

Usage: `hadoop fs -get [-ignorecrc] [-crc] <src> <localdst>`

Copy files to the local file system. Files that fail the CRC check may be copied with the `-ignorecrc` option. Files and CRCs may be copied using the `-crc` option.'

getmerge

Usage: `hadoop fs -getmerge <src> <localdst> [addnl]`

Takes a source directory and a destination file as input and concatenates files in `src` into the destination local file.

ls

Usage: `hadoop fs -ls <args>`

For a file returns stat on the file with the following format:

permissions number_of_replicas userid groupid filesize modification_date
modification_time filename

For a directory it returns list of its direct children as in Unix. A directory is listed as:

permissions userid groupid modification_date modification_time dirname

Example:

- `hadoop fs -ls /user/hadoop/file1`

lsr

Usage: `hadoop fs -lsr <args>`

Recursive version of `ls`. Similar to Unix `ls -R`.

mkdir

Usage: `hadoop fs -mkdir [-p] <paths>`

Takes path uri's as argument and creates directories.

mv

Usage: `hadoop fs -mv URI [URI ...] <dest>`

Moves files from source to destination. This command allows multiple sources as well in which case the destination needs to be a directory. Moving files across file systems is not permitted.

put

Usage: `hadoop fs -put <localsrc> ... <dst>`

Copy single `src`, or multiple `srcs` from local file system to the destination file system. Also reads input from stdin and writes to destination file system.

rm

Usage: `hadoop fs -rm [-skipTrash] URI [URI ...]`

Delete files specified as args. Only deletes non empty directory and files.

rmr

Usage: `hadoop fs -rmr [-skipTrash] URI [URI ...]`

Recursive version of delete.

setfacl

Usage: `hadoop fs -setfacl [-R] [-b|-k -m|-x <acl_spec> <path>][--set <acl_spec> <path>]`

Sets Access Control Lists (ACLs) of files and directories.

setrep

Usage: `hadoop fs -setrep [-R] [-w] <numReplicas> <path>`

Changes the replication factor of a file. If path is a directory then the command recursively changes the replication factor of all files under the directory tree rooted at path.

stat

Usage: `hadoop fs -stat URI [URI ...]`

Returns the stat information on the path.

tail

Usage: `hadoop fs -tail [-f] URI`

Displays last kilobyte of the file to stdout.

text

Usage: `hadoop fs -text <src>`

Takes a source file and outputs the file in text format. The allowed formats are zip and TextRecordInputStream.

touchz

Usage: `hadoop fs -touchz URI [URI ...]`

Create a file of zero length.

Example:

- `hadoop -touchz pathname`

Creating a file using nano:







