# POST mapreduce/streaming

**Table of contents**

## 1 Description

Create and queue an [Hadoop streaming MapReduce](#) job.

## 2 URL

`http://www.myserver.com/templeton/v1/mapreduce/streaming`

## 3 Parameters

| Name | Description | Required? | Default |
|------|-------------|-----------|---------|
| **input** | Location of the input data in Hadoop. | Required | None |
| **output** | Location in which to store the output data. If not specified, Templeton will store the output in a location that can be discovered using the [queue](#) resource. | Optional | See description |
| **mapper** | Location of the mapper program in Hadoop. | Required | None |
| **reducer** | Location of the reducer program in Hadoop. | Required | None |
| **file** | Add an HDFS file to the distributed cache. | Optional | None |
| **define** | Set an Hadoop configuration variable using the syntax `define=NAME=VALUE` | Optional | None |
| **cmdenv** | Set an environment variable using the syntax `cmdenv=NAME=VALUE` | Optional | None |
| **arg** | Set a program argument. | Optional | None |
| **statusdir** | A directory where Templeton will write the status of the Map Reduce job. If provided, it is the caller's responsibility to | Optional | None |

| Name | Description | Required? | Default |
|------|-------------|-----------|---------|
| | remove this directory when done. | | |
| **callback** | Define a URL to be called upon job completion. You may embed a specific job ID into this URL using `$jobId`. This tag will be replaced in the callback URL with this job's job ID. | Optional | None |

The [standard parameters](#) are also supported.

## 4 Results

| Name | Description |
|------|-------------|
| **id** | A string containing the job ID similar to "job_201110132141_0001". |
| **info** | A JSON object containing the information returned when the job was queued. See the Hadoop documentation ([Class TaskController](#)) for more information. |

## 5 Example

### Code and Data Setup

```
% cat mydata/file01 mydata/file02
Hello World Bye World
Hello Hadoop Goodbye Hadoop

% hadoop fs -put mydata/ .

% hadoop fs -ls mydata
Found 2 items
-rw-r--r--   1 ctdean supergroup          23 2011-11-11 13:29 /user/ctdean/mydata/file01
-rw-r--r--   1 ctdean supergroup          28 2011-11-11 13:29 /user/ctdean/mydata/file02
```

### Curl Command

```
% curl -s -d user.name=ctdean \
       -d input=mydata \
       -d output=mycounts \
```

```
        -d mapper=/bin/cat \
        -d reducer="/usr/bin/wc -w" \
        'http://localhost:50111/templeton/v1/mapreduce/streaming'
```

## JSON Output

```
{
 "id": "job_201111111311_0008",
 "info": {
         "stdout": "packageJobJar: [] [/Users/ctdean/var/hadoop/hadoop-0.20.205.0/share/
hadoop/contrib/streaming/hadoop-streaming-0.20.205.0.jar...
                    templeton-job-id:job_201111111311_0008
                    ",
         "stderr": "11/11/11 13:26:43 WARN mapred.JobClient: Use GenericOptionsParser for
 parsing the arguments
                    11/11/11 13:26:43 INFO mapred.FileInputFormat: Total input paths to
 process : 2
                    ",
          "exitcode": 0
        }
}
```

## Results

```
% hadoop fs -ls mycounts
Found 3 items
-rw-r--r--   1 ctdean supergroup          0 2011-11-11 13:27 /user/ctdean/mycounts/_SUCCESS
drwxr-xr-x   - ctdean supergroup          0 2011-11-11 13:26 /user/ctdean/mycounts/_logs
-rw-r--r--   1 ctdean supergroup         10 2011-11-11 13:27 /user/ctdean/mycounts/
part-00000

% hadoop fs -cat mycounts/part-00000
      8
```