hpipe Documentation

3.0.0

Menglong TAN

2015 10 28

Contents

Hpipe

- HadoopHive
- HDFSSqlite
- GraphvizGraphviz
-

CHAPTER 1

1.1

1.2

1.2.1 hpipe

:

```
Execute a hpipe workflow
Usage:
    hpipe [options]
Options:
    -h, --help          Print this message
    -v, --verbose       Use verbose output

    -p, --path          Working path
    -f, --flow          Entry filename of workflow
    --max-retry         Max retry times of failed jobs, default: 3
    --start             Start from a specific job
    --end               End at a specific job

    --status-saver      Method to track job status
                        default: sqlite
                        available: hdfs, sqlite

    --namenode          Address of Hadoop NameNode, default: 127.0.0.1:8020
    --sqlite            File path for sqlite database

    --hadoop            Enable Hadoop streaming job
    --jar               Path of Hadoop streaming jar file

    --odps              Enable ODPS job
    --odps-endpoint     Address of ODPS endpoing
    --odps-project      ODPS project name
    --odps-access-id    ODPS access id
    --odps-access-key   ODPS access key

    --hive              Enable Hive job

    --bizdate           Set variable $bizdate in YYYYMMDD format
```

```
    --less-log          Less log output
```

- `-p --path`
- `-f --flow`
- `--max-retry`
- `--start`
- `--end`
- `--bizdate $bizdate $gmtdate`

Hpipe2

1. SQLite
2. HDFS

hpipe`--status-saver`

SQLite`--sqlite /var/run/hpipe/status.db`

HDFSHDFSNamenode`--namenode master:8020`

Hpipe

Hadoop Streaming

- `--hadoop` Hadoop
- `--jar` Hadoop streamingjar

Hive

- `--hive` Hive

Aliyun ODPS

- `--odps` ODPS
- `--odps-endpoing` ODPS endpoint
- `--odps-project` ODPS project
- `--odps-access-id` ODPS access id
- `--odps-access-key` ODPS access key

1.3

1.3.1

:

```
digraph wordcount {
    dummy -> step1
    dummy -> step2
    step1 [
          name="step1"
          type="hadoop"
          vars="$date=${date:YYYYMMDD}"
          input="/tmp/hpipe/examples/wordcount/input/part-*"
          output="/tmp/hpipe/examples/wordcount/output/${bizdate}/step1"
          mapper="cat"
          reducer="wc -l"
          mapred.reduce.tasks=1
          test.custom.val="today is ${date}, yestoday is ${bizdate}, and yestoday is ${bizdate}"
          ]
    step2 [
          name="step2"
          type="hadoop"
          vars="$date=$bizdate"
          input="/tmp/hpipe/examples/wordcount/input/part-*"
          output="/tmp/hpipe/examples/wordcount/output/${bizdate}/step2"
          mapper="cat"
          reducer="wc -l"
          ]
    dummy [
          name="dummy"
          type="dummy"
          vars="$bizdate=${date:YYYYMMDD}+2*$day"
          ]
}
```

1.3.2

name

type

output

`output="/tmp/example/${bizdate}/"`

vars

-
- $
-
  - int
  - `$name='test'`
  - `$$date=$bizdate`
  - `$date=${date:YYYYMMDD}`

`output script`

- `output="/tmp/${bizdate}"`
- `script="test.sh ${bizdate}"`

nonstrict

`nonstrict="true"`

- hadoop
- hive
- script

hadoop

hive

hql HiveQL

script HiveQL

option Hive `hive.exec.dynamic.partition.mode=nonstrict`

script

interpreter `perl bash $PATH`

script Script

gmtdate

bizdate hpipe `--bizdate gmtdate`

job_report

1.3.3

‘‘->‘‘

nonstrict

`examples/email`

```
digraph email {
    dummy -> script
    script -> mailitsh [nonstrict="true"]
    mailitsh -> mailitpl
    ()
}
```

- genindex
- modindex
- search