

Configure MoSKito-Central Storage

Intro

As we mentioned before, you'll come to the stage of configuring MoSKito-Control storage after declaring dependencies and instantiating MoSKito-Central service. This is true for both Embedded and Remote Modes.

After instantiating, MoSKito-Central will run, but basically will do nothing. We will have to add some storages to actually store the data.

To do so, create or modify the *moskito-central.json* file, as in the example below:

moskito-central.json

```
{
  "@storages": [
    {
      "name": "json-file",
      "clazz": "org.moskito.central.storage.fs.FileSystemStorage",
      "configName": "moskito-fs"
    },
    {
      "name": "csv-file",
      "clazz": "org.moskito.central.storage.fs.CSVFileStorage",
      "configName": "moskito-csv"
    },
    {
      "name": "psql-db",
      "clazz": "org.moskito.central.storage.psql.PSQLStorage",
      "configName": "moskito-psql-hibernate"
    },
    {
      "name": "tsdb",
      "clazz": "org.moskito.central.storage.tsdb.OpenTSDBStorage",
      "configName": "moskito-tsdb"
    }
  ]
}
```

This variant will add two storages, a FileSystem storage and a CSFFileStorage. They require own configuration files (referred to in *configName* parameter).

FileSystemStorage

moskito-fs.json

```
{
  "pattern": "/tmp/central/{host}/{component}/{producer}/{interval}/{date}/{date}_{time}_{producer}.json",
  "serializer": "org.moskito.central.storage.serializer.GsonSerializer",
  "includeIntervals": "*",
  "excludeIntervals": "1m",
  "includeProducers": "*",
  "excludeProducers": ""
}
```

In the code block above, logging is enabled in the following **path**:

```
/tmp/central/{host}/{component}/{producer}/{interval}/{date}/{date}_{time}_{producer}.json
```

The example could be:

```
/tmp/central/colin/app/RequestURIFilter/5m/27_03_2013/27_03_2013_16_06_RequestURIFilter.json
```

- The GsonSerializer is used to convert java based snapshot objects to json.
- All Intervals are included
- All producers are included
- 1m Interval is excluded (to reduce amount of data).

CSVFileStorage

moskito-csv.json

```
{
  "pattern": "/tmp/central/csv/{host}/{component}/{producer}/{interval}/{producer}_{stat}.csv",
  "@entries": [
    {
      "includedProducers": "SessionCount",
      "includedStats": "**",
      "includedIntervals": "**"
    },
    {
      "includedProducers": "RequestURIFilter",
      "includedStats": "cumulated",
      "includedIntervals": "5m,1h"
    },
    {
      "includedProducers": "ThreadStates",
      "includedStats": "**",
      "includedIntervals": "1m"
    }
  ]
}
```

The CSVFileStorage is configured slightly differently. It also supports a path, but it writes a file per stat not a file per producer, therefore it makes sense to have *stat* as part of the path.

For example in `/tmp/central/csv/colin/app/ThreadStates/1m/`:

```
ThreadStates_BLOCKED.csv
ThreadStates_NEW.csv
ThreadStates_RUNNABLE.csv
ThreadStates_TERMINATED.csv
ThreadStates_TIMED_WAITING.csv
ThreadStates_WAITING.csv
ThreadStates_cumulated.csv
```

The producers and their stats are configured separately in *entries*.

```

@ConfigureMe
public class CSVFileStorageConfigEntry {
    @Configure
    private String includedIntervals;
    @Configure
    private String excludedIntervals;
    @Configure
    private String includedProducers;
    @Configure
    private String excludedProducers;
    @Configure
    private String includedStats;
    @Configure
    private String excludedStats;
}

```

Some examples on entries:

Configuration entry.	Meaning
<pre> { "includedProducers": "SessionCount", "includedStats": "**", "includedIntervals": "*" }, </pre>	Log all stats and all intervals for SessionCount.
<pre> { "includedProducers": "RequestURIFilter", "includedStats": "cumulated", "includedIntervals": "5m,1h" }, </pre>	Log the RequestURIFilter Producer, stat <i>cumulated</i> only. Log 5m and 1h Interval.
<pre> { "includedProducers": "ThreadStates", "includedStats": "**", "includedIntervals": "1m" } </pre>	Log all ThreadStates Producer data in 1m Interval.

PSQLStorage

moskito-psql-hibernate,json

```

{
  driver:"org.hsqldb.jdbc.JDBCdriver",
  url:"jdbc:hsqldb:/tmp/hsqldb-central",
  userName:"daa",
  password:"daa",
  persistenceUnitName:"hsqldbSnapshotStorage",
  "@mappings" : [
    {
      "producerName": "SessionCount",
      "statEntityClass": "org.moskito.central.storage.psql.HttpSessionStatisticsEntity"
    },
    {
      "producerName": "testProducerId,*API,*Service*",
      "statEntityClass": "org.moskito.central.storage.psql.ServiceStatsEntity"
    }
  ]
}

```

Configures DB connection options and JPA persistenceUnitName that described in META-INF/persistence.xml file.

"mappings" property describes mapping producers statistics on DB table.

"producerName" - you can point as full producer name like "testProducerId", also wildcard values like *API - PersistenceAPI, *Service* - RMICentralServiceImpl.

OpenTSDBStorage

moskito-tldb.json

```
{
  "url": "http://localhost:4242/api/put",
  "@entries": [
    {
      "includedProducers": "SessionCount",
      "includedStats": "*",
      "includedIntervals": "*"
    },
    {
      "includedProducers": "RequestURIFilter",
      "includedStats": "cumulated",
      "includedIntervals": "5m,1h"
    }
  ]
}
```

The OpenTSDBStorage is configured in the code block above, **url** is OpenTSDB HTTP API path for storing data. The producers and their stats are configured separately in *entries* similar to CSVFileStorage.

Each stat are stored as a set of *metrics* with following tags: *hostName*, *intervalName*, *componentName*.

Example:

stat.json

```
{
  "metaData": {
    "producerId": "RequestURIFilter",
    "componentName": "app",
    "hostName": "MacBook-Pro",
    "intervalName": "5m",
    "creationTimestamp": 1385327470774,
    "arrivalTimestamp": 1385327470868,
    "category": "filter",
    "subsystem": "default",
    "statClassName": "net.anotheria.moskito.core.predefined.FilterStats"
  },
  "stats": {
    "cumulated": {
      "Last": "0",
      "CR": "0",
      "Max": "-9223372036854775808",
      "MCR": "0",
      "ERR": "0",
      "TR": "0",
      "TT": "0",
      "Avg": "NaN",
      "Min": "9223372036854775807"
    }
  }
}
```

Stat above stored as a set of metrics:

set-of-metrics.json

```
[
  {
    "metric": "RequestURIFilter.cumulated.Last",
    "timestamp": 1385327470774,
    "value": 0,
    "tags": {
      "hostName": "MacBook-Pro",
      "intervalName": "5m",
      "componentName": "app"
    }
  },
  {
    "metric": "RequestURIFilter.cumulated.CR",
    "timestamp": 1385327470774,
    "value": 0,
    "tags": {
      "hostName": "MacBook-Pro",
      "intervalName": "5m",
      "componentName": "app"
    }
  },
  {
    "metric": "RequestURIFilter.cumulated.Max",
    "timestamp": 1385327470774,
    "value": -9223372036854775808,
    "tags": {
      "hostName": "MacBook-Pro",
      "intervalName": "5m",
      "componentName": "app"
    }
  },
  {
    "metric": "RequestURIFilter.cumulated.MCR",
    "timestamp": 1385327470774,
    "value": 0,
    "tags": {
      "hostName": "MacBook-Pro",
      "intervalName": "5m",
      "componentName": "app"
    }
  },
  {
    "metric": "RequestURIFilter.cumulated.ERR",
    "timestamp": 1385327470774,
    "value": 0,
    "tags": {
      "hostName": "MacBook-Pro",
      "intervalName": "5m",
      "componentName": "app"
    }
  },
  {
    "metric": "RequestURIFilter.cumulated.TR",
    "timestamp": 1385327470774,
    "value": 0,
    "tags": {
      "hostName": "MacBook-Pro",
      "intervalName": "5m",
      "componentName": "app"
    }
  },
  {
    "metric": "RequestURIFilter.cumulated.TT",
    "timestamp": 1385327470774,
    "value": 0,
    "tags": {
      "hostName": "MacBook-Pro",
      "intervalName": "5m",
      "componentName": "app"
    }
  },
],
```

```
{
  "metric": "RequestURIFilter.cumulated.Min",
  "timestamp": 1385327470774,
  "value": 9223372036854775807,
  "tags": {
    "hostName": "MacBook-Pro",
    "intervalName": "5m",
    "componentName": "app"
  }
}
```

Notes:

- Empty and not numeric values skipped.
- Integration done for OpenTSDB running with --auto-metric flag, so please use it too.
- /api/put refers to OpenTSDB 2.x versions.