



## SmartTUBE 5. Description of configuration files

This document describes configuration files which contain meta data and settings of SmartTUBE server components.

SmartLabs LLC is able to change and improve any product described in this guide, to change the use conditions and description, and also to edit the guide itself without preliminary notice. Please, follow the changes on our web site.

SmartTUBE, SmartLabs logo are trademarks of SmartLabs, LLC.

Confidential. No part of this document can be reissued, republished or distributed without the permission of SmartLabs.

## Table of contents

1	Introduction .....	4
2	*.conf files overview .....	4
2.1	Example .....	4
2.2	Parameters description .....	5
3	system.conf .....	6
3.1	Example .....	6
3.2	Parameters description .....	6
4	container.conf.....	7
4.1	Example .....	7
4.2	Parameters description .....	8
5	cache-server.conf.....	9
5.1	Example .....	9
5.2	Parameters description .....	10
6	data-model-core.conf .....	11
6.1	Example .....	11
7	data-server.conf .....	11
7.1	Example .....	11
7.2	Parameters description .....	12
8	report.conf.....	13
8.1	Example .....	13
8.2	Parameters description .....	13

# 1 Introduction

**Configuration file** is a text file in JSON format containing meta data and settings of SmartTUBE server component. A configuration file must have a common part which is the same for all configuration files and a part relevant only for a specific component.

The common part is described in the section [\\*.conf files overview](#).

Other properties are described in the sections on specific configuration files.

An example of setting the properties (a configuration file for SmartSDPDataServer component):

```
properties = {
    sdp-properties-file-path = "/opt/smartlabs/smarttube/master/config/" #
    sdp-properties-file-path property has a string value
    http-request-handlers = 2000 #
    http-request-handlers property has an integer value
    database = { #
    database is a complex property
        server = "postgresql" # the
        real name of the property is going to be database.server
        connection-string = "jdbc:postgresql://pgsql_db_ip:5432/psql_dbname"
        login = "login"
        password = "password"
        max-connections = 100 # the
        real name of the property is going to be database.max-connections
    }
    scheduled-jobs = [ # the
    property is a JSON array of objects
        {
            enabled = false
            runWhen = "0 59 23 * * ?" # every day at 23:59 #
            every day at 23:59
            method = "cacheoperations#reload"
            params = {
                caches = "MetaContentYear, Epg"
            }
        }
    ]
    trace-http-requests = true #
    trace-http-requests property has a logical value
}
```

## 2 \*.conf files overview

All configuration files have some parameters in common.

### 2.1 Example

```
component = {
    description = "SDP Cache Server"
    jar = "sdp-cache-server.jar"
    class = "smartsdp.cacheserver.main.SmartSDPCacheServer"
```

```

singleton = true
load-on-startup = 12
servlet-context-path = "/api/*;/cmds/*"
install-path = "/opt/smartlabs/sdp/modules/cache-server"
properties = {
  ...
}
}

```

## 2.2 Parameters description

Parameter	Description
install-path	Full path to the directory, where a SmartTUBE component is installed. The <code>jar</code> file of the component must be stored in the <code>Component-Install-Directory/bin</code> directory. The <code>Component-Install-Directory/lib</code> directory contains <code>jar</code> files of external libraries that affect the component's work
jar	Name of the component's <code>jar</code> file. For example, <code>SmartBGPTableAggregator.jar</code>
class	Full name (including the Java package name) of the main class of the component. For example, <code>smartlabs.bgptableaggregator.main.SmartBGPTableAggregator</code>
singleton	An attribute indicating whether the component will work as a single instance ( <code>yes</code> value), or there must be several of instances of this component ( <code>no</code> value)
load-on-startup	An integer that defines the priority order of the component start: components with a smaller value are started earlier than the components with a bigger value
servlet-context-path	Only for servlet components: context of HTTP requests
properties	Initial values of the component's properties and links with other components. Properties can be simple properties (e.g., <code>load-on-startup = 12</code> ) or collections (e.g., <code>sdp-dataservers = [ "10.25.190.11:8084", "10.25.190.12:8084" ]</code> ). Each property has a name and a value: <ul style="list-style-type: none"> <li>▶ <b>Property name</b> – property identifier that will be used by the component to request the property value.</li> <li>▶ <b>Property value</b> – the value that will be sent to the component.</li> </ul>

## 3 system.conf

### 3.1 Example

```
system-properties = {
    default-profile-language = "en"
    default-profile-currency = "EUR"
    user-accounts.store-days = 30
    pending-notifications.store-hours = 24
    media-positions.store-days = 3
    upload-path = "/opt/smartlabs/images/"
    sdp-dataservers = [ "10.25.190.11:8084", "10.25.190.12:8084" ]
    cache-servers = [ "10.25.190.13:8084", "10.25.190.14:8084" ]
    http-proxy = {
        host = ""
        port = 80
    }
    formats = {
        date-incoming = "MM/dd/yyyy"
        date-outgoing = "MM/dd/yyyy"
        time-incoming = "HH:mm:ss"
        time-outgoing = "HH:mm:ss"
        datetime-incoming = "MM/dd/yyyy HH:mm:ss"
        datetime-outgoing = "MM/dd/yyyy HH:mm:ss"
    }
    subscription-timeout=14400000
}
```

### 3.2 Parameters description

Parameter	Description
default-profile-language	Code of the profile language by default
default-profile-currency	Currency code by default
user-accounts.store-days	Number of days the data about the user account are kept in cache
pending-notifications.store-hours	Number of hours notifications for STBs are kept
media-positions.store-days	Number of days data on media positions are kept in cache
upload-path	Directory, where images are loaded to
sdp-dataserver	▶ host – a computer, where the SmartTUBE database

	<p>server is installed</p> <ul style="list-style-type: none"> <li>▶ port – port of access to the database server; it must be the same as <code>tomcat-attributes.port</code> in <code>container.conf</code></li> </ul> <p>Host and port of data servers. The port must match <code>tomcat-attributes.port</code> port in <code>container.conf</code> on the servers with installed <code>SmartSDPDataServer</code> component</p>
cache-server	<ul style="list-style-type: none"> <li>▶ host – a computer, where the Cache server is installed</li> <li>▶ port – port of access to the Cache server; it must be the same as <code>tomcat-attributes.port</code> in <code>container.conf</code></li> </ul> <p>Host and port of Cache servers. The port must match <code>tomcat-attributes.port</code> port in <code>container.conf</code> on the servers with installed <code>SmartSDPCacheServer</code> component</p>
http-proxy	<ul style="list-style-type: none"> <li>▶ host – a computer, where the proxy server is installed; leave the field empty if proxy is not used</li> <li>▶ port – proxy server port if proxy is used</li> </ul>
formats	<ul style="list-style-type: none"> <li>▶ date-incoming – format of date for incoming data</li> <li>▶ date-outgoing – format of date for outgoing data</li> <li>▶ time-incoming – format of time for incoming data</li> <li>▶ time-outgoing – format of time for outgoing data</li> <li>▶ datetime-incoming – format of date and time for incoming data</li> <li>▶ datetime-outgoing – format of date and time for outgoing data</li> </ul>
subscription-timeout	Timeout after the device subscribes to Comet notification channel used by the server to close the old channel socket, ms

## 4 container.conf

### 4.1 Example

```

component-container = {
    container-name = "CC on Projects"
    cli-enabled = false # if you install the container
                        as a Linux daemon, it must be set to false

```

```

control-port = 4440
log-file-path = "/var/log/smartlabs/sdp"
log-level = "INFO"
check-dependences = true
web-interface-enabled = true
stop-on-component-error = false
pause-before-notify-tomcat-runned = 5000
tomcat-attributes {
    port = 8084
    maxConnections = 10000
    maxThreads = 1000
    connectionTimeout = 10000
    socket-soKeepAlive = true # socket.soKeepAlive is
needed
}
remote-logging-server = {
    host = "127.0.0.1"
    port = 5556
}
}

```

## 4.2 Parameters description

Parameter	Description
container-name	Name of the components container (CC) that is used to create the logger name for debugging (by default the name of the host is used, where CC is deployed)
cli-enabled	Managing the CC in a command line or by sending <code>ccadmin</code> utility commands over the socket  If you install the container as a Linux daemon, this parameter must be set to <i>false</i>
control-port	The port for managing the CC; commands from the <code>ccadmin</code> utility are accepted at this port. By default the value is 4440
log-file-path	Log file directory <code>SmartSDPCompContainer.log</code>
log-level	Minimum level of detail for the log
check-dependences	Indicator of the requirement to check when loading the component of libraries JAR files versions: <i>true</i>   <i>false</i>
web-interface-enabled	Indicator of the web interface permission through built-in Apache Tomcat servlet container: <i>true</i>   <i>false</i>



	If <code>web-interface-enabled</code> equals <code>true</code> , Apache Tomcat servlet container is going to launch, which is going to create <code>Contaner_Install_Directory/work</code> directory for its operation. This requires that the launcher has the respective rights.
<code>stop-on-component-error</code>	Indicator of the requirement to stop if an error occurs when creating / initializing / launching components. If it equals <code>false</code> , the specified errors are ignored (but recorded in the log)
<code>pause-before-notify-tomcat-runned</code>	Pause in ms, after which the components are notified on the successful launch of Tomcat
<code>tomcat-attributes.port</code>	Port number for the integrated Apache Tomcat
<code>tomcat-attributes.maxConnections</code>	Maximum number of connections for in-built Apache Tomcat
<code>tomcat-attributes.maxThreads</code>	Maximum number of streams served for in-built Apache Tomcat
<code>tomcat-attributes.connectionTimeout</code>	Timeout of Apache Tomcat connection
<code>tomcat-attributes.socket-soKeepAlive</code>	Value of keep-alive socket setting (Apache Tomcat): <code>true/false</code>
<code>remote-logging-server.host</code>	Host, where the remote log server is deployed
<code>remote-logging-server.port</code>	Port of the remote log server  To disable remote logging, delete the <code>remote-logging-server</code> section

## 5 cache-server.conf

### 5.1 Example

```

component = {
  description = "SDP Cache Server"
  jar = "sdp-cache-server.jar"
  class = "smartsdp.cacheserver.main.SmartSDPCacheServer"
  singleton = true
  load-on-startup = 12
  servlet-context-path = "/api/*;/cmds/*"
  install-path = "/opt/smartlabs/sdp/modules/cache-server"
}

```

```

properties = {
  sdp-properties-file-path = "/opt/smartlabs/smarttube/master/config/"
  elasticsearch = {
    cluster-name = "smart_cache_dev"
    nodes = [ "127.0.0.1:9300" ]
  }
  number-of-http-get-handlers = 2000
  number-of-http-post-handlers = 10
  number-of-notification-handlers = 10
  trace-notification = false
  recreate-indices-on-startup = false
  trace-http-requests = true
}
}

```

## 5.2 Parameters description

Parameter	Description
description, jar...	<a href="#">*.conf files overview</a>
servlet-context-path	Mapping to /api and /cmds
sdp-properties-file-path	Path to the file with SmartTUBE server settings (/opt/smartlabs/smarttube/master/config/ by default)
elasticsearch.cluster-name	Name of Elasticsearch server cluster.  Must be unique for the local network
elasticsearch.nodes	Elasticsearch servers connected in a cluster
number-of-http-get-handlers	Number of worker streams processing HTTP GET requests
number-of-http-post-handlers	Number of worker streams processing HTTP POST requests
number-of-notification-handlers	Number of notification processors
trace-notification	Flag of notification tracing
recreate-indices-on-startup	Indicator of the requirement to recreate all caches when reloading the component
trace-http-requests	Flag of tracing HTTP requests in the log

## 6 data-model-core.conf

### 6.1 Example

```
component = {
  description = "Core Data Model"
  jar = "sdp-datamodel-core.jar"
  class = "smartsdp.datamodelcore.main.SmartSDPDataModelCore"
  singleton = true
  load-on-startup = 1
  install-path = "/opt/smartlabs/sdp/modules/data-core"
}
```

For parameters description see [\\*.conf files overview](#).

No special properties for the component.

## 7 data-server.conf

### 7.1 Example

```
component = {
  description = "JSON Data Server"
  jar = "sdp-data-server.jar"
  class = "smartsdp.dataserver.main.SmartSDPDataServer"
  singleton = true
  load-on-startup = 10
  servlet-context-path = "/sdpapi/*;/sdpapibd/*"
  install-path = "/opt/smartlabs/sdp/modules/data-server"
  properties = {
    sdp-properties-file-path = "/opt/smartlabs/smarttube/master/config/"
    http-request-handlers = 2000
    database = {
      server = "postgresql"
      connection-string = "jdbc:postgresql://pgsql_db_ip:5432/psql_dbname"
      login = "login"
      password = "password"
      max-connections = 100
    }
    notifications = {
      database-path = "/var/smartlabs/sdp/data-server/db"
      receptionists = [ "127.0.0.1:5051" ]
      dispatch-jobs-period-sec = 1000
      dispatch-batch-size = 1000
      trace-dispatching = false
    }
    scheduled-jobs = [
      {
        enabled = false
        runWhen = "0 59 23 * * ?" # every day at 23:59
        method = "cacheoperations#reload"
        params = {
          caches = "MetaContentYear, Epg"
        }
      }
    ]
  }
  epg-import.url = "http://111.1.1.1:80/epg.xml"
  epg-import.login = "Admin"
```

```

epg-import.password = "AdminPass"
    }
}
]
trace-http-requests = true
}
}

```

## 7.2 Parameters description

Parameter	Description
description, jar...	<a href="#">*.conf files overview</a>
sdp-properties-file-path	Path to the file with SmartTUBE server settings (/opt/smartlabs/smarttube/master/config/ by default)
http-request-handlers	Number of processors of HTTP-requests
database.server	Type of database server
database.connection-string	Database server connection string
database.login	Database user login
database.password	Database user password
database.max-connections	Maximum number of connections to the database
notifications.database-path	Path to database notifications folder
scheduled-jobs	<p>JSON array of objects describing jobs that are run periodically. Each JSON object of a job has the following format:</p> <ul style="list-style-type: none"> <li>▶ name - job name.</li> <li>▶ enabled - flag enabling to run this job.</li> <li>▶ runWhen - time / period of running the job (string). Please see format description here: <a href="http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/tutorial-lesson-06.html">http://www.quartz-scheduler.org/documentation/quartz-2.x/tutorials/tutorial-lesson-06.html</a>.</li> <li>▶ method - name of the service and method to run in "service#method" format.</li> <li>▶ params - parameters passed to the methods during</li> </ul>

	<p>the launch. For example:  <u>for EPG import from the file</u>  epg-import.url – URL of a file with EPG.  epg-import.login – login to access the EPG file.  epg-import.password – password to access the EPG file.  <u>for importing a weather forecast from an external system</u>  appid – identifier of the application from which the forecast is going to be imported</p>
trace-http-requests	Flag of tracing HTTP requests in the log

## 8 report.conf

### 8.1 Example

```

component = {
  description = "SDP Report"
  jar = "sdp-report.jar"
  class = "smartsdp.report.main.SmartSDPReport"
  singleton = true
  load-on-startup = 14
  servlet-context-path = "/report/*"
  install-path = "/opt/smartlabs/sdp/modules/reports"
  #install-path = "/SmartSDPReport/target"
  properties = {
    sdp-properties-file-path = "/opt/smartlabs/smarttube/master/config/"
    number-of-http-get-handlers = 100
  }
}

```

### 8.2 Parameters description

Parameter	Description
description, jar...	<a href="#">*.conf files overview</a>
sdp-properties-file-path	Path to the file with SmartTUBE server settings (/opt/smartlabs/smarttube/master/config/ by default)
number-of-http-get-handlers	Number of processors of HTTP GET requests